Dissertation
an der

FAKULTÄT FÜR MATHEMATIK,
INFORMATIK UND STATISTIK
DER LUDWIG–MAXIMILIANS–UNIVERSITÄT MÜNCHEN

Architektur und Werkzeuge für dynamisches Identitätsmanagement in Föderationen

eingereicht von

Daniela Pöhn

München, den 05. August 2016

1. Gutachter: **Prof. Dr. Wolfgang Hommel**, Universität der Bundeswehr München
2. Gutachter: **Prof. Dr. Gabi Dreo Rodosek**, Universität der Bundeswehr München

Tag der mündlichen Prüfung: 25. November 2016

Eidesstattliche Versicherung
(Siehe Promotionsordnung vom 12.07.11, §8, Abs. 2 Pkt. 5)

Hiermit erkläre ich an Eides statt, dass die Dissertation von mir selbstständig, ohne unerlaubte Beihilfe angefertigt ist.

München, den 05. August 2016

.....
Daniela Pöhn

Danksagung

Die vorliegende Arbeit entstand im Rahmen meiner Tätigkeiten als wissenschaftliche Mitarbeiterin am Leibniz-Rechenzentrum der Bayerischen Akademie der Wissenschaften.
Mein besonderer Dank gilt an dieser Stelle meinem Doktorvater, Prof. Dr. Wolfgang Hommel, der diese Arbeit von Beginn an unterstützt sowie durch seine Anregungen und fachliche Auseinandersetzung maßgeblich zu ihrem Gelingen beigetragen hat. Im Rahmen des Projekts GÉANT-TrustBroker schuf er ein ideales Umfeld, um die Ideen zu vertiefen und in die Praxis umzusetzen. Die vielen gemeinsam geschriebenen Paper haben Spaß gemacht - Danke! Mein Dank gilt außerdem Frau Prof. Dr. Gabi Dreo Rodosek, die sich dazu bereit erklärt hat, die Aufgabe des zweiten Gutachters zu übernehmen und diese Arbeit zu lesen.
Ferner möchte ich auch allen Kolleginnen und Kollegen am Leibniz-Rechenzentrum danken, die durch Diskussionen diese Arbeit beeinflusst haben. Besonderen Dank gebührt Michael Grabatin, der die Implementierung im Rahmen seiner Masterarbeit umsetzte, und Tanja Hanauer für die Diskussionen im Rahmen der "Kanban"-Runde.
Nicht zuletzt danke ich allen, die hier nicht genannt sind und denen ein Dank gebührt.

Zusammenfassung

Federated Identity Management (FIM) hat die Motivation, Identitätsdaten eines Benutzers von einer Heimatorganisation, d. h. Identity Provider (IdP), einem Dienstbetreiber, Service Provider (SP) genannt, bereitzustellen. Dies ermöglicht zum einen die Vermeidung von Redundanzen und Inkonsistenzen und zum anderen kann der Benutzer viele weitere Dienste nutzen, ohne sich zusätzliche Benutzerkonten merken zu müssen. Mit der Security Assertion Markup Language (SAML) und dem Protokoll OpenID Connect haben sich in Wirtschaft und Research & Education (R&E) zwei Standards etabliert. Durch die vermehrte Vernetzung zeigen sich zunehmend die Grenzen der aktuell eingesetzten Architektur.
In dieser Arbeit wird zunächst eine umfangreiche Anforderungsanalyse anhand verschiedener Szenarien durchgeführt, die unterschiedliche Perspektiven auf die Architektur und ihre Anforderungen ermöglicht. Die Schwerpunkte dieser mehr als 70 strukturierten und gewichteten Anforderungen liegen dabei auf der Automatisierung und der Skalierbarkeit, Vertrauen sowie der Interoperabilität. Zudem sollen organisatorische Randbedingungen wie Sicherheits- und Datenschutzaspekte eingehalten werden.
Im Rahmen eines umfassenden, gesamtheitlichen Architekturkonzepts wird anschließend eine Managementplattform für dynamisches Federated Identity Management erarbeitet. Neben der Spezifikation des orchestrierten, technischen Metadatenaustausches, der den bestehenden Ansätzen fehlt, fokussiert diese Arbeit auf die organisatorische Eingliederung hinsichtlich des IT Service Managements. Hierbei liegt der Fokus auf Security Management und Change Management. Zur Kompensation weiterer Defizite bisheriger Ansätze werden zwei zusätzliche Werkzeuge spezifiziert, die auf eine optimierte Interoperabilität bestehender FIM-Systeme sowie die Automatisierung und Skalierbarkeit existierender Abläufe abzielen.
Eine Beschreibung der prototypischen Implementierung der Managementplattform und der Werkzeugkonzepte mit einer Diskussion ihrer Skalierbarkeit und die methodische Anwendung auf ein realistisches Szenario runden diese Arbeit ab.

Abstract

Federated Identity Management (FIM) has the motivation to provide identity data of users from their home organisation, also called Identity Provider (IdP), to a Service Provider (SP). This facilitates the prevention of redundancy and inconsistency, while users can re-use their home account for other services, without remembering further user accounts and passwords. The Security Assertion Markup Language (SAML) and the protocol OpenID Connect are two well-known standards within the industry sector and research & education (R&E) environment. Due to the ongoing interconnectedness, the limitations of the current architecture are increasingly revealed.

In the first part of the thesis, a profound and comprehensive analysis is presented, in order to illustrate different perspectives on the architecture and the requirements. The focus of the more than seventy structured and weighted requirements in the categories function, non-functional, organizational as well as privacy- and security-specific categories lays in the automation and scalability of the approach as well as trust implications and interoperability. As part of the holistic, integrated architecture conceived in this thesis, a management platform for dynamic FIM has been developed. Besides the precise specification of the orchestrated, technical metadata exchange, special emphasis has been put on the organizational integration concerning the IT service management. Dependencies and effects on the security management and change management have been investigated in detail. To compensate further shortcomings of existing approaches, two new FIM components have been specified, which enhance the interoperability between FIM systems in heterogeneous identity federations, as well as the scalability and automation of existing workflows.

The thesis is concluded with a description of the prototypical implementation of the management platform and the tool concepts as well as a discussion on their scalability characteristics and the application of the architecture to a realistic scenario.

Inhaltsverzeichnis

Kapitel 1

Einleitung

Inhalt dieses Kapitels

In unserer modernen, vernetzten Welt werden Arbeitnehmer häufig aufgerufen, mobil und flexibel auf Veränderungen in der Arbeitswelt zu reagieren. Die Mobilität wurde in Europa innerhalb der Mitgliedstaaten durch das Schengen-Abkommen vereinfacht, was u. a. den Wegfall der Grenzkontrollen und die Vereinheitlichung der Vorschriften für die Einreise und den kurzfristigen Aufenthalt von Ausländern im Schengen-Raum zur Folge hatte. Der Bologna-Prozess intensiviert diese Dynamik im Hochschulumfeld. Während die Hochschul-Abschlüsse vergleichbar gemacht wurden, sollte die Mobilität, beispielsweise von Studenten hinsichtlich Auslandssemester, verbessert werden.

Analog dazu werden sowohl der europäische Hochschulraum und Forschungsraum als auch Unternehmen verstärkt miteinander vernetzt. Das GÉANT-Projekt baut zu diesem Zweck ein pan-europäisches Internet-Verbindungsnetzwerk der europäischen Forschungseinrichtungen, den so genannten *National Research and Education Networks (NRENs)*, auf. Bis jetzt sind über 50 Millionen Benutzer verteilt auf 10.000 Institute innerhalb von Europa miteinander vernetzt [GÉA16a]. In der Wirtschaft konsolidieren Firmen und ihre Zulieferer ihre Kommunikationsnetze, um möglichst reibungslos miteinander zu kollaborieren. Gleichzeitig eröffnen die Rechnerallgegenwart und die Omnipräsenz des Internets neue Wege der Kollaboration. Wissenschaftler aus diversen Ländern forschen zusammen in internationalen Projekten, ohne vor Ort einen Arbeitsplatz zu benötigen. Die Dynamik erfordert Anpassungen in der Informationstechnologie (IT), um den geänderten Anforderungen gerecht zu werden. Als ein wichtiger Aspekt erweist sich das der IT-Sicherheit zugeordnete Gebiet des Identity Managements, welches unter anderem die Identifikation, Authentifizierung und Autorisierung von Benutzern beinhaltet.

Im *Identity Management* wird traditionell für jede Person an sämtlichen Instituten, an denen sie arbeitet, eine digitale Identität erstellt. Diese ist mit persönlichen Informationen wie beispielsweise der E-Mail-Adresse verknüpft, die als Attribute bezeichnet werden. Je nach Rolle, d. h. Aufgabengebiet, Pflichten, Verantwortungen und Privilegien, hat der Benutzer

unterschiedliche Berechtigungen. Zum Beispiel darf ein Projektmitarbeiter im Projektordner auf der Dateiablage neue Dateien ablegen und die Projektdokumentation im Wiki pflegen, während er keinen Zugriff auf die Verwaltungssoftware mit den Daten aller Beschäftigten hat. Die hierfür gespeicherten Informationen können verteilt auf den einzelnen Systemen oder Diensten vorgehalten werden. Jedoch ist der Pflegeaufwand, beispielsweise wenn ein Mitarbeiter eine Organisation verlässt, erhöht, da sein Benutzerkonto auf allen Systemen deaktiviert oder gelöscht werden muss. Wird ein System vergessen, sind die Daten nicht mehr konsistent. Daher werden die entsprechenden Informationen organisationsintern in einem Softwaresystem gespeichert, um an einer zentralen Stelle alle Aspekte der Authentifizierung und Autorisierung einheitlich verwalten zu können. Dieses *Identity & Access Management (I&AM)* besteht aus einer Datenbasis, meist in Form eines Lightweight Directory Access Protocol (LDAP)-basierten Verzeichnisdienstes, einem Management-Interface sowie den als Konnektoren bezeichneten Schnittstellen zu den angeschlossenen Datenquellen und Diensten. Daraus resultiert eine Trennung von Benutzerverwaltung und Diensten. Ein weiterer organisatorischer Vorteil für den Betreiber einzelner Dienste ist die Automatisierung der Abläufe im Lebenszyklus von digitalen Identitäten vom Anlegen von Benutzerkonten bis hin zum automatischen Löschen nicht mehr benötigter Konten. Folglich wird das nicht mehr benötigte Benutzerkonto zentral gelöscht und nicht mehr auf jedem einzelnen System oder von jedem Dienst. Die Nutzer können durch das I&AM mit demselben Login-Verfahren, meist über eine Kombination aus Benutzername und Passwort realisiert, organisationsweit von allen Diensten Gebrauch machen. Häufig wird in den Heimatorganisationen Single Sign On (SSO) eingesetzt, um nach einer einmaligen Authentifizierung auf unterschiedliche Dienste, z. B. Intranet, Wiki und ein Zeitabrechnungstool, zugreifen zu können ohne sich erneut anmelden zu müssen.

Obgleich das I&AM auf die einzelne Organisation begrenzt ist, sollen bestehende Synergien zwischen Organisationen, beispielsweise durch die Forschungskooperation einer Firma mit einem Forschungsinstitut, genutzt werden. Daher bietet sich für einen kleinen Benutzerkreis meist an, Ad-hoc-Lösungen zu implementieren. Damit ist gemeint, dass Nutzer der Organisation A, um beispielsweise das Projekt-Wiki zur Dokumentation der Ergebnisse in Organisation B verwenden zu können, auch jeweils ein lokales Benutzerkonto bei Organisation B benötigen. Solch eine einfache Regelung wird u. a. durch folgende Nachteile erkauft: Betreiber des Services haben einen erhöhten Pflegeaufwand für Benutzerdaten, während sich Anwender neben dem Benutzerkonto ihrer Organisation auch separate Zugangsdaten für die Kooperation merken müssen.

Da diese Lösung schlecht skaliert sowie Redundanzen und Inkonsistenzen bei der Datenhaltung verursacht, wurde im Rahmen des *Federated Identity Managements (FIMs)* die Verwaltung von Benutzern zwischen mehreren Organisationen vereinfacht. Dieser Ansatz ermöglicht eine verteilte Benutzerverwaltung, bei der jeder Benutzer mindestens einer Heimatorganisation, dem so genannten *Identity Provider (IdP)*, zugeordnet ist. Initiiert durch den Benutzer ruft der Anbieter von Diensten und Ressourcen, auch *Service Provider (SP)* genannt, die benötigten Informationen (*Attribute*) vom IdP des Benutzers ab. Dies ist beispielsweise bei hochschulübergreifenden Lernplattformen sinnvoll, damit Studenten spezielle virtuelle Kurse an anderen Hochschulen besuchen können, ohne ein weiteres Benutzerkonto

anlegen zu müssen. Der Anbieter des Kurses kann für den zu belegenden Kurs spezifische Informationen zur Überprüfung der Zugangsvoraussetzungen von der Heimat-Hochschule des Studenten abfragen, z. B. Studiengang und Fachsemester. Dazu müssen technische Grundlagen festgelegt werden, u. a. folgende:

- Die Entitäten, d. h. IdP und SP, müssen die jeweiligen Kommunikationsendpunkte kennen. Zu diesem Zweck werden beispielsweise Serverzertifikate für Transport Layer Security (TLS)-verschlüsselte Verbindungen ausgetauscht. Die Informationen zu den Endpunkten können in sogenannten *Metadaten* gespeichert werden, die geeignet zwischen den Kommunikationspartnern ausgetauscht werden müssen. Dazu ist eine *Metadaten-Verwaltung* notwendig, basierend auf dem Begriff des FIM-Standards Security Assertion Markup Language (SAML) [`SAML2Core`] [CKPM05].

- Die Entitäten benötigen ein gemeinsames Protokoll zur Kommunikation. Im Hochschulumfeld ist dies meist *SAML* in Form dessen Implementierungen *Shibboleth* oder *SimpleSAMLphp*. Alternativ kann *OAuth 2.0* mit der Identitätsschicht von *OpenID Connect* verwendet werden.

- Die Entitäten benötigen ein gemeinsames Verständnis der Sprache. Dies bedeutet, dass die *Syntax* und *Semantik* der ausgetauschten Informationen über Benutzer deckungsgleich sein sollte oder dass eine Abbildung (Mapping) existiert. Das föderationsweit genutzte Datenmodell wird analog LDAP-basierten I&AM-Systemen als *Schema* bezeichnet. Die Authentication and Authorization Infrastructure (AAI) des Verein zur Förderung eines Deutschen Forschungsnetzes e. V. (DFN-Verein) basiert beispielsweise auf dem internationalen `eduPerson`-Schema mit der nationalen Erweiterung `dfnEduPerson` [DFN15b].

- Der Identity Provider muss die für den Service Provider benötigten Attribute kennen. Um die Privatsphäre des Anwenders zu schützen, sendet der IdP nur die benötigten Attribute an den SP. Im Shibboleth-Umfeld ist es notwendig, dass der IdP-Administrator den sogenannten Attribute Filter und bei Bedarf den Attribute Resolver für jeden Service Provider, den mindestens ein Nutzer der Heimatorganisation verwenden will, konfiguriert. Äquivalente Mechanismen sind in anderen Softwarepaketen wie SimpleSAMLphp ebenfalls erforderlich. Zusätzlich muss der Nutzer der Weitergabe der Attribute zustimmen.

Darüber hinaus gibt es einen organisatorischen Rahmen. *Föderationen*, d. h. der Zusammenschluss verschiedener IdPs und SPs, schreiben beispielsweise die Datenqualität und die Einhaltung datenschutzrechtlicher Bestimmungen vor. In einer bilateralen Vereinbarung, bei kommerziellen Anbietern häufig in Form von vertraglichen Vereinbarungen (*Service Level Agreement (SLA)*) zwischen zwei Entitäten, werden Qualität und Verfügbarkeit der zu ermittelnden Daten zwischen den beteiligten Organisationen festgehalten, z. B. wann spätestens veraltete Benutzerdaten geändert bzw. gelöscht werden müssen. Dazu ist ein gewisses Vertrauen der involvierten Parteien, die einer *Föderation* angehören, erforderlich. Folglich ist neben der Verlässlichkeit (*behavioural trust*), u. a. auf die Qualität der gelieferten Da-

ten, ein technisches Vertrauen (*technical trust*) für den Datenaustausch notwendig. Infolge der sublimierten Authentifizierung benötigen Anwender lediglich die Zugangsdaten ihrer Heimatorganisation. Der DFN-Verein bildet ein Beispiel für FIM, welches seit 2007 eine Föderation aus wissenschaftliche Einrichtungen, d. h. für Hochschulen sowie Forschungseinrichtungen, und wissenschaftsnahen Anbietern, mit mehr als 470 IdPs und SPs in Deutschland betreibt [Ter16]. Die AAI des DFN-Verein (DFN Authentication and Authorization Infrastructure (DFN-AAI)) ermöglicht beispielsweise Studenten, E-Learning-Systeme anderer Hochschulen in Anspruch zu nehmen, ohne mit dem organisatorischen Overhead in Form von Selbstregistrierung und Einsendung von Immatrikulationsbescheinigungen belastet zu werden. Gleichzeitig werden die IdPs in die Verlässlichkeitsklassen *Basic* und *Advanced* eingeteilt, je nachdem welches Verfahren sie zur Überprüfung der Identität sowie zur Authentifizierung verwenden und wie schnell sie Daten aktualisieren.

Föderationen existieren nicht nur im wissenschaftlichen Umfeld innerhalb der Entitäten eines Landes, sondern auch in der Wirtschaft, wo beispielsweise ein Automobilhersteller gemeinsam mit den Zulieferern und seinen Vertriebspartnern IT-Dienste im Rahmen von *Odette Federated Identity Management Service Standards for Automotive (SESAM)* nutzt. Wenn nun der Automobilhersteller branchenübergreifend mit anderen Herstellern kooperieren will, muss seine Föderation um die Föderation des Geschäftspartners erweitert werden oder beide Föderationen schließen sich zu einer höheren Organisationsform zusammen.

Durch die technische Heterogenität der verschiedenen Föderationen unterscheiden sich die Organisationen beispielsweise in eingesetzten Schemata und FIM-Protokollen sowie bezüglich divergierender organisatorischer Zielsetzung, wie Datenqualität, rechtlicher Grundlagen und Datenschutz. Auf Grund dessen gleicht eine weltweite Föderation mit einem gemeinsamen, universell gültigen Schema einer Utopie (vgl. [YLJ09]). Daher werden Konzepte für das Identitätsmanagement in *Inter-Föderationen* (*Inter-Federation Identity Management (Inter-FIM)*), die ein Zusammenschluss mehrerer Föderationen vor allem auf organisatorischer Ebene gleichen, erarbeitet und prototypisch implementiert. Es gibt aktuell zwei Formen von Inter-Föderationen:

- Bilaterale Inter-Föderationen, d. h. die Inter-Föderation besteht aus zwei Föderationen, die untereinander einen Vertrag abschließen. Dieser legt die technischen Grundlagen zur Kommunikation fest. Die Föderation der wissenschaftlichen Einrichtungen der USA, InCommon [InC13], hat z. B. je ein Abkommen mit den National Institutes of Health [U.S16] und mit der UK Federation [Jis16].

- Vollvermaschte Inter-Föderation aus mehreren Föderationen, zum Beispiel KALMAR-2 [Kal15] und eduGAIN [GÉA16b]. Mit eduGAIN wurde eine lose Inter-Föderationen aus vielen nationalen Föderationen hauptsächlich innerhalb von Europa im Rahmen des GÉANT-Projektes initiiert, um die Zusammenarbeit im Hochschulumfeld zu vereinfachen.

Die technischen Grundlagen der Föderationen gelten ebenfalls für Inter-Föderationen, jedoch sind dort insbesondere in vollvermaschten Netzen mit vielen Föderationen mehr Enti-

täten beteiligt, was folgende Auswirkungen nach sich zieht:

Metadaten-Verwaltung. Die Metadaten der einzelnen Föderationen, bestehend aus den Metadaten aller teilnehmenden Organisationen, werden meist in einer zentralen Einheit der Inter-Föderation aggregiert und beispielsweise über den *Metadata Distribution Service (MDS)* an die beteiligten Föderationen zurückgegeben. Die Verwaltung der Föderationen filtern teilweise die Metadaten der eigenen Teilnehmer heraus, da diese bereits in den nationalen Metadaten vorhanden sind, bevor die aggregierten Metadaten an die einzelnen Teilnehmer der Föderation über Push- oder Pull-Verfahren verteilt werden. Die Größe der Metadaten steigt mit Anzahl der Entitäten und Föderationen, wodurch sie aktuell knapp 270.000 Zeilen Extensible Markup Language (XML) umfasst [GÉA16c].

Service Level Agreement und Level of Assurance. Analog zu Föderationen muss der IdP wissen, welche Attribute der Service Provider einsetzt. Das benötigte Vertrauen in den SP beim Versand der Benutzerinformationen sowie die manuelle Konfiguration der Attribut-Filter finden in den vollvermaschten Inter-Föderationen auf Grund der Größe und Anzahl von Teilnehmern zumeist erst dann statt, wenn ein Benutzer seinen Identity Provider über die gewünschte Nutzung eines Dienstes informiert. Dazu muss der Nutzer wissen, an wen er sich wenden muss und welche Informationen zur Konfiguration benötigt werden. Gleichzeitig muss der IdP die Anforderungen des SPs erfüllen. Daher beträgt die Zeitspanne zwischen gewünschter Nutzung und Freischaltung des Dienstes häufig mehrere Tage, wenn der SP dem IdP genug vertraut und umgekehrt.

Konvertierung. Wenn SP und IdP nicht dieselbe Sprache sprechen, d. h. wenn sich Semantik und Syntax unterscheiden, ist eine Konvertierung der Attribute notwendig. Beispielsweise benötigt der SP einen `fullname`, wofür der IdP `givenname` und `surname` miteinander verknüpfen muss. Diese Transformationsregeln werden aktuell vom Administrator des IdPs manuell für jeden Service Provider erstellt.

Dies zeigt eine gestiegene Komplexität im Vergleich zu Föderationen, was bei einfacher Erweiterung der Architektur schnell Auswirkungen auf die Skalierbarkeit und Nutzbarkeit hat. Auf Grund der technischen Basis ergeben sich zusätzlich folgende Beschränkungen:

- Nutzer von Identity Providern können ohne manuelle Konfiguration nur Dienste von Service Providern verwenden, deren Metadaten in der Inter-Föderation bzw. Föderation bekannt sind. So ist es beispielsweise im Hochschulumfeld nicht problemlos möglich, das Projekt-Wiki mit einem Partner aus der Wirtschaft zu teilen, dessen Metadaten und somit Informationen über den Kommunikationsendpunkt nicht in dem Metadaten-Satz der Föderation enthalten sind.

- Benutzer können nur Dienste nutzen, wenn ihr IdP den Anforderungen des SPs entspricht. Dies wird aktuell u. a. ebenfalls über die Metadaten der Föderation geregelt.

- Zusatzinformationen, die nicht im Schema vorgesehen sind, können nicht ausgetauscht

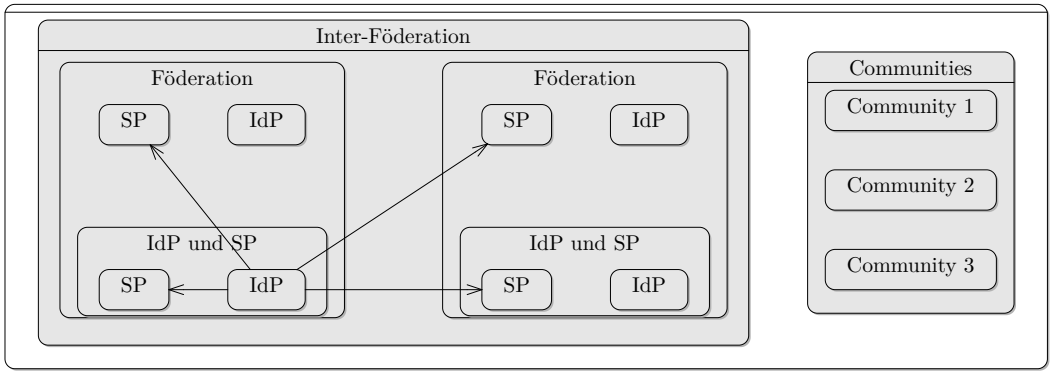

Abbildung 1.1.: Beispiel einer möglichen Inter-Föderation

werden.

Eine pragmatische Lösung ist die Teilnahme von Service Providern und teilweise auch Identity Providern in mehreren Föderationen und Inter-Föderationen. Dies zieht neue Herausforderungen bezüglich der Skalierbarkeit nach sich, da die Entitäten den Anforderungen jeder einzelnen Föderation bzw. Inter-Föderation genügen müssen. Beispielsweise müssen die Attributinformationen für die verschiedenen Schemata angepasst werden.

Der Aufbau einer beispielhaften Inter-Föderation wird in Abbildung 1.1 gezeigt. Sie besteht aus zwei Föderationen, die jeweils unterschiedliche Entitäten, d. h. IdP, SP und eine Heimatorganisation mit IdP und SP, enthalten. Um das vollvermaschte Netz zu verdeutlichen, existieren zwischen einem Identity Provider und allen Service Providern in beiden Föderationen Geschäftsbeziehungen, die vorher ausgehandelt werden mussten. Parallel dazu befinden sich die Föderationen der Communities, auch Virtuelle Organisationen (VOs) oder Forschungsgruppen genannt. Ein Benutzer kann gleichzeitig sowohl einer oder mehreren Föderationen der Communities als auch einem bzw. verschiedenen IdPs in der Inter-Föderation zugehören.

1.1. Motivation und Zielsetzung

I&AM- und FIM-Lösungen existieren seit Jahren und sind dementsprechend technisch ausgereift. Die bisherigen Ansätze des Inter-FIM, die in den Kapiteln 2 und 3 detaillierter skizziert werden, haben verschiedene Vorteile und Nachteile. Eine der gemeinsamen Schwachstellen ist die schlechte Skalierbarkeit bei steigender Anzahl an Teilnehmern. Dies ist insofern von Bedeutung, da immer mehr Föderationen sich den Inter-Föderationen anschließen wollen und eine stärkere Kooperation zwischen Europa mit GÉANT, Nordamerika und Asien angestrebt wird (vgl. [GÉA16b]). Ein weiterer Aspekt ist die Prozessoptimierung für Administratoren bei der Etablierung des Vertrauens, der Freigabe von Attributen sowie bei der Konvertierung

von Attributen, um den Arbeitsaufwand zu minimieren und die Interoperabilität zu steigern.

Skalierbarkeit bezüglich der Vertrauensverhältnisse. In einem vollvermaschten Netz, beispielsweise in aktuellen Inter-Föderationen, verbinden sich theoretisch n IdPs mit m SPs, was insgesamt zu $n*m$ bilateralen Verträgen führt. In der Realität werden nicht alle $n*m$ Verbindungen benötigt, da für einen Identity Provider nur $m - x$ Service Provider in Frage kommen. Dessen ungeachtet verdeutlicht die Anzahl den Aufwand, der betrieben werden muss, damit Nutzer die Dienste ihrer Wahl verwenden können. Sollen viele verschiedene Dienste in Anspruch genommen werden, ergibt sich ein erheblicher organisatorischer Overhead. Mit jeder weiteren teilnehmenden Föderation steigt dieser für die beteiligten Entitäten. Äquivalent steigt die Größe des Metadatensatzes, der die Grundlage für das technische Vertrauen bildet. Die Metadaten aller teilnehmenden Entitäten werden aggregiert, obwohl jede einzelne Entität nur einen kleinen Teil daraus benötigt. Dies wirkt sich insbesondere auf die Performanz aus.

Verlässlichkeit. Jede Einrichtung verwendet unterschiedliche Verfahren zur

- Feststellung der Identität ihrer Nutzer.
- Authentifizierung ihrer Nutzer vor dem Zugriff auf einen Dienst.
- Datenhaltung und Datenpflege der digitalen Identitäten.

Durch diese Merkmale werden die Identity Provider unterschiedlichen Verlässlichkeitsklassen, auf Englisch *Level of Assurance (LoA)*, zugeteilt. Die DFN-AAI gliedert die LoA in die Klassen *Basic* und *Advanced* auf, je nachdem, welche Anforderungen die Identity Provider erfüllen. Zusätzlich existiert eine Klasse zu Testzwecken. Für die Verlässlichkeitsklasse Advanced muss die Organisation u. a. den zukünftigen Nutzer durch ein persönliches Gespräch und mit einem amtlichen Dokument zur Identitätsfeststellung identifizieren. Service Provider können bei der Registrierung ihrer Dienste anhand ihrer Anforderungen festlegen, welche Verlässlichkeitsklassen sie unterstützen wollen. Bei Inter-Föderationen wird bislang nicht nach unterschiedlichen Verlässlichkeitsklassen unterschieden. Erschwerend kommt hinzu, dass jede Föderation unterschiedliche Ansprüche hat und Schwerpunkte setzt bezüglich der einzelnen Aspekte, wie Authentifizierung und Datenhaltung, welche sich mit den bisher entwickelten Verlässlichkeitsklassen in den Normen nur unzureichend abbilden lassen. Weitere Unzulänglichkeiten sind in den Beziehungen zwischen den Entitäten zu finden. Zum einen müssen sich Service Provider auf die IdPs verlassen, dass die Semantik der Attribute korrekt ist. Zum anderen müssen IdPs den SP vertrauen, dass nur benötigte Informationen angefragt werden (*Prinzip der Datensparsamkeit*), SPs die Benutzerinformationen nur zum angegebenen Zweck verwenden, sicher speichern und dass sie diese persönlichen Informationen nicht ungefragt an Dritte weitergeben.

Benutzbarkeit des Attributsschemas. Auf Grund der Heterogenität der eingesetzten Datenhaltungssysteme in den Instituten und Hochschulen, wurde für jede Ebene der

Kollaboration in Inter-Föderationen, d. h. föderativ und inter-föderativ, mindestens ein gemeinsames *Attributsschema* konzipiert. Dieses stellt den kleinsten gemeinsamen Nenner dar, um für einen Großteil der Anwendungen ausreichend Informationen bereit zu stellen. Dem entgegengesetzt kann auch die Gesamtsumme alle Schemata eingesetzt werden, was zu einer Vielzahl an teils mehrfach vorhandenen Attributen führen würde. Bedingt durch die Differenz zwischen den historisch gewachsenen Datenschemata der Identity Provider und dem Attributsschema der Föderation, ergo vom Schema der Inter-Föderation, müssen die Attribute in das jeweils vom Service Provider benötigte Format konvertiert werden. Die Erstellung der benötigten Konvertierungsregeln ist aktuell eine manuelle Aufgabe des IdPs.

Folglich kann die aktuelle Lösung höchstens als suboptimal bezeichnet werden. Sie behindert u. a. Wissenschaftler und Angestellte bei ihrer Arbeit, erzeugt einen bedeutenden Mehraufwand bei der Administration und macht die Nutzung bestimmter Dienste für Benutzer unattraktiv. Daher gibt es zurzeit Bestrebungen, der steigenden Komplexität mit dynamischen Lösungen im inter-föderationsweiten Identitätsmanagement zu entgegnen und die einzelnen Föderationen der Communities mit einzubinden. Als Beispiele können hierfür das Metadata Query Protocol und Public Endpoint Entities Registry (PEER) gelten, die in Kapitel 3 näher erläutert werden. Die Herausforderungen im Identity Management über Föderationsgrenzen beziehen sich zum einen auf die technischen Komponenten, wie Skalierbarkeit, Kompatibilität der Einzelbausteine mit der Gesamtarchitektur und Implementierung. Zum anderen ist die organisatorische Sicht wichtig, beispielsweise Auswirkungen auf das Change-Management, Risiko- und Qualitätsmanagement sowie Anpassung und Optimierung der Geschäftsprozesse.

Das Ziel dieser Arbeit ist dementsprechend die Analyse verschiedener Szenarien, um daraus systematisch Anforderungen an eine geeignete Lösung zu stellen. Bereits existierende Ansätze und Implementierungen werden detailliert hinsichtlich der herausgearbeiteten Kriterien betrachtet, Vorteile und Defizite aufgezeigt.
Darauf basierend wird eine Architektur mit den dafür nötigen Elementen, zum Beispiel Information, Kommunikation, Organisation und Funktion sowie Schnittstellen und Werkzeugen, erstellt. Die bereits genannten Aspekte werden miteinbezogen. So soll die Architektur den Aufbau von Vertrauensbeziehungen optimieren und die notwendigen Konvertierungsregeln bereitstellen. Ein Hauptaugenmerk soll auf die Benutzbarkeit, Optimierung und Automatisierung der bestehenden Prozesse geworfen werden. Die Architektur soll skalierbar sein, um sich dynamisch der Größe der Föderation und der neu erstellten Vertrauensbeziehungen anzupassen. Dabei ist ein wesentlicher Aspekt, dass sich die Lösung nahtlos in die gegebene Infrastruktur einbinden lässt. Zusätzlich soll die Erweiterung auch für existierende Föderationen angewandt werden können.
Weitere Aspekte, die berücksichtigt werden sollen, sind die unterschiedlichen LoAs der Föderationen und Ansprüche an ein Qualitätsmanagement. Die Architektur soll sich auf den Einsatz externer Identity Provider und zusätzlicher Attributsquellen, den so genannten Attribute Authorities (AAs), anpassen lassen. Da sich IT-Infrastrukturen im Laufe der Zeit bedingt durch den technologischen Fortschritt stetig ändern, ist es nötig, Schnittstellen zu etablierten Managementprozessen, wie dem Security und Change Management, zu integrieren.

Auf Basis dieser Konzepte wird eine Referenzarchitektur entworfen, welche die benötigten Schnittstellen zu den bestehenden Lösungen bietet und sich so in die bereits existierende Strukturen einbinden lässt. Während der Arbeit werden neue, benötigte Komponenten konzipiert, die Funktionsweisen und Schnittstellen spezifiziert und ihr Zusammenspiel mit anderen Komponenten definiert.

1.2. Fragestellungen

Zur Verdeutlichung der Komplexität bei der Konzipierung einer geeigneten Architektur skizziert Abbildung 1.2 verschiedene Dimensionen. Diese werden im Folgenden kurz erläutert und die sich daraus ergebenen Fragestellungen abgeleitet. Dabei sind neben den eigentlichen Aspekten der Architektur auch die Rahmenbedingungen zu beachten.

Art und Automatisierung der Konvertierung von Attributen. Die Art und die damit einhergehende Automatisierung der Konvertierung spielen eine wichtige Rolle bei der Skalierbarkeit von Inter-Föderationen. Derzeit ist es die Aufgabe eines IdP-Administrators, *manuell* die Konvertierungsregeln für jeden, von einem Nutzer verwendeten SP zu erstellen. Eine mögliche Verbesserung des Ist-Zustandes stellt die *Übernahme und Anpassung* bereits vorhandener Regeln dar. Die *automatische Übernahme* vertrauenswürdiger Konvertierungsregeln, die einmalig erstellt wurden, bedeutet einen weiteren Progress in Richtung *automatischer Konvertierung*, beispielsweise durch ontologische Ansätze.

Skalierbarkeit. Die Skalierbarkeit der zu wählenden Architektur und der zugrunde liegenden Föderationen ist ein wichtiges Kriterium für die Adaption der noch zu konzipierenden Komponenten. Eine bisherige Schwachstelle stellt die schlechte Skalierbarkeit bei zunehmender Anzahl an Teilnehmern und Geschäftsbeziehungen dar. Deswegen ist es essentiell, die Dynamik der inter-föderativen Strukturen, etwa die Expansion der Inter-Föderation, in die Arbeit mit einfließen zu lassen. Momentan muss *manuell* bilaterales Vertrauen zwischen IdP und SP in einer festen Föderation ausgehandelt werden, bevor Anwender die Möglichkeit haben, einen Dienst einzusetzen. Diesen Prozess kann beispielsweise der Benutzer anstoßen, indem *halbautomatisch* eine angezeigte Nachricht weitergeleitet wird, um die Aushandlung anzustoßen. Benutzerfreundlicher ist eine *voll-automatische* Initiierung in dynamischen virtuellen Föderationen, beispielsweise mit Hilfe der Datenbasis eines Dienstes, der direkt oder indirekt einen Überblick über die bereits vorhandenen Geschäftsbeziehungen hat.

Vertrauen und Zuverlässigkeit. Unerlässlich für den Abschluss von SLAs, folglich für die Nutzung eines Dienstes, ist das Vertrauensverhältnis zwischen SP und IdP. Ein wichtiges Element hierfür ist die Zuverlässigkeit des IdPs bezüglich der Datenqualität. Dies kann mit Klassen, wie bei der DFN-AAI [DFN15a], oder in *diskreten Werten*, beispielsweise zwischen 1 und 4 analog zu National Institute of Standards and Technology (NIST) [BDN+13] wiedergegeben werden. 1 entspricht der niedrigsten Stufe,

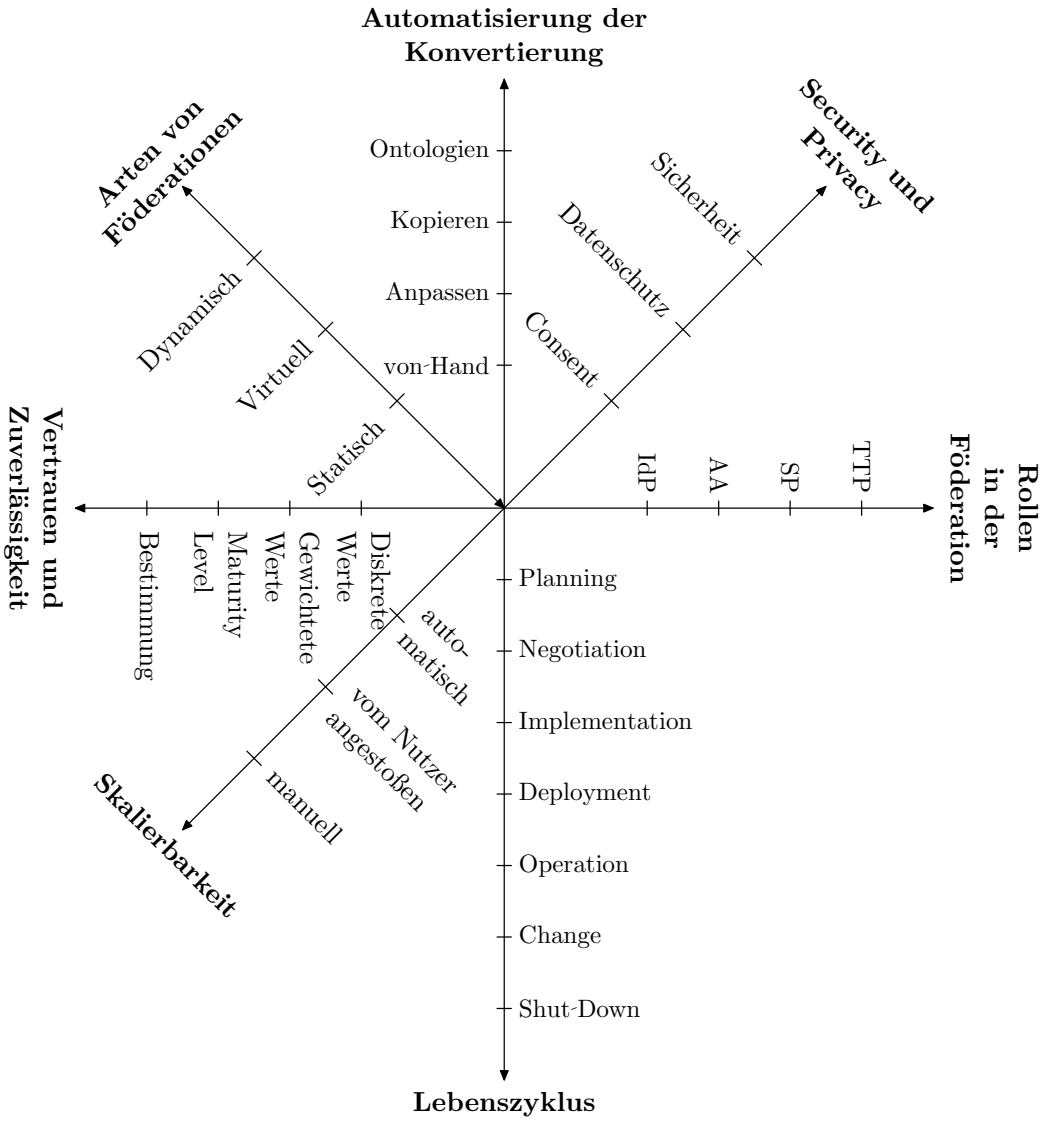

Abbildung 1.2.: Ausgewählte Dimensionen der Problemstellung

während 4 die höchste Verlässlichkeit kennzeichnet. Die niedrigste Stufe bzw. Verlässlichkeitsklasse kann bedeuten, dass die Identität des Nutzers nicht überprüft wird. Folglich ist es möglich, dass die Qualität der Daten und die Art der Authentifizierung von IdPs der niedrigsten Klasse nicht den Anforderungen des SPs entsprechen. Bei einem hohen Vertrauen zwischen IdP und SP beschließt beispielsweise der Identity Provider auch die optionalen Attribute freizugeben. Eine unterschiedliche *Gewichtung* von Kriterien, die zu einer Klasse oder einem diskreten Wert führen, wurde bisher nicht angedacht. Ferner gibt es die Möglichkeit, *Maturity Levels* zur Vergleichbarkeit aufzubauen.

Security und Privacy. Ein interessanter Aspekt aus Anwendersicht ist das *Consent-Management*, d. h. die Freigabe der übertragenen Attribute für einen SP. Bislang geschieht dieser datenschutzrechtlich sensible Vorgang bei der erstmaligen Nutzung eines Dienstes; die Freigabe kennt keine Zwischenstufen zwischen zustimmen und ablehnen. Einhergehend mit dem Consent-Management müssen die gesetzlichen Rahmenbedingungen im Bereich des *Datenschutzes*, die bei länderübergreifenden Geschäftsbeziehungen stark variieren können, eingehalten werden. Ferner ist die *Sicherheit* der Architektur und ihrer Protokolle von Bedeutung. Bereits etablierte Sicherheitsmechanismen dürfen durch die Bildung von zusätzlichen Föderationen nicht beeinträchtigt werden. Die Sicherheit der Neuerungen, d. h. der Architektur an sich sowie der benötigten Schnittstellen und Datenübertragungsprotokolle, muss analysiert und entsprechend konzipiert werden. Durch die verteilte Struktur ist es wichtig, die Sicherheit des gesamten Systems zu betrachten, und nicht nur der einzelnen Entitäten.

Die Rahmenbedingungen der Inter-Föderationen und Föderationen stellen ein wesentliches Grundgerüst für diese Arbeit dar. Da die Lösung auf den etablierten Elementen des FIMs und I&AMs sowie deren Strukturen, Schnittstellen und Prozesse aufbaut, sollen grundlegende Änderungen nicht erforderlich werden.

Rollen. Jede Inter-Föderation besteht aus zumindest zwei Föderationen, die wiederum mindestens einen IdP und einen SP aufweisen. Den einzelnen Organisationen ist es dabei möglich, eine oder mehrere unterschiedliche Rollen in mehreren Föderationen auszuüben. Neben IdP und SP können so genannte Attribute Authorities (AAs) die vom IdP gelieferten Benutzerinformationen ergänzen oder ihre Korrektheit bestätigen. Die zuletzt genannte Eigenschaft ist vor allem dann relevant, wenn mehrere Organisationen benötigt werden, um dem Service Provider die Eigenschaften eines Benutzers glaubhaft zu versichern. Für die Mitglieder der einzelnen Communities ist, bedingt durch die Teilnahme an unterschiedlichen Föderationen, die Aggregation der Attribute von Interesse. Organisationen, auf die keine der bereits beschriebenen Rollen zutrifft, aber zu denen entsprechende Vertrauensbeziehungen bestehen, werden als Trusted Third Party (TTP) bezeichnet. Dazu gehören beispielsweise Broker und Lokalisierungsdienste, über die ein Service Provider den zuständigen Identity Provider ermitteln kann. Ferner kümmert sich die Föderationsverwaltung, z. B. der DFN-Verein in Deutschland, um die organisatorischen Belange einer Föderation, vom Aggregieren der Metadaten über das Aufsetzen von Verträgen bis hin zur Pflege der Richtlinien (Policies). Die Richtlinien bilden das rechtliche Rahmenwerk einer nationalen Föderation, indem sie beispielsweise Aufnahmebedingungen festlegen, Konditionen für die Nutzung von Diensten definieren und die Verlässlichkeitsklassen bestimmen.

Arten von Föderationen. Es gibt verschiedene Arten von Föderationen und Kooperationen, die eine gemeinsame Datenbasis benötigen. Neben den eher *statischen* nationalen Föderationen existieren *virtuelle* Föderationen der Communities, die Organisationen in mehreren Föderationen sowie einzelne externe Forschungseinrichtungen umfassen können. Zudem können *dynamische* Föderationen für Projekte gebildet werden, die nur für eine vordefinierte Projektlaufzeit existieren.

Lebenszyklus. Neben den Komponenten der Architektur an sich, unterliegen sowohl die einzelnen Dienste und die SPs als auch ganze Föderationen und Inter-Föderationen einem Lebenszyklus. Dabei werden die Abschnitte *Planning*, *Negotiation*, *Implementation*, *Deployment*, *Operation*, *Change* und *Shut-Down* durchlaufen.

Basierend auf einer Analyse der Defizite aktueller Lösungsansätze werden folgende Aspekte betrachtet:

- Wie sieht eine skalierbare, sichere Architektur für Föderationen und Inter-Föderationen aus?
- Welche Schnittstellen müssen zwischen den bereits vorhandenen Komponenten und der Lösung existieren?
- Welche Abhängigkeiten müssen zwischen bestehenden Komponenten und Management-Prozessen bei einer Änderung der Architektur beachtet werden?
- Wie können durch eine geeignete Lösung Prozesse vereinfacht, Workflows automatisiert und gleichzeitig die Benutzbarkeit verbessert werden?
- Wie gestaltet sich eine zu entwickelnde Komponente, so dass zwischen Entitäten einer potentiellen Föderation, die einer bzw. unterschiedlichen Trust-Levels angehören, möglichst automatisch Beziehungen aufgebaut werden können?
- Wie können Benutzerattribute automatisch und unabhängig von verwendetem Schema und Software verstanden werden?

Im nächsten Abschnitt wird die zur Analyse und Lösung dieser Fragestellungen gewählte Vorgehensweise näher erläutert.

1.3. Vorgehensmodell

Abbildung 1.3 zeigt die im Rahmen dieser Arbeit gewählte Vorgehensweise. In Kapitel 2 werden die technischen Grundlagen von FIM, Inter-FIM und Föderationen erläutert, um mit Hilfe der einheitlichen *Terminologie* eine grundlegende Basis für die Arbeit zu schaffen. Anhand der darauf aufbauenden *Szenarien* werden systematisch Anforderungen an die Architektur, Schnittstellen, Werkzeuge und die notwendigen Protokolle abgeleitet. Der konkludierende *Anforderungskatalog* an eine skalierbare Lösung zur Etablierung von Vertrauen stellt die Grundlage für die Bewertung existierender und des im Rahmen dieser Arbeit entwickelten Konzepts dar.

Die eben erwähnte Bewertung existierender Ansätze und Implementierungen aus der Wissenschaft wird im Kapitel 3 durchgeführt. Die Analyse dient dazu, die vorliegende Arbeit

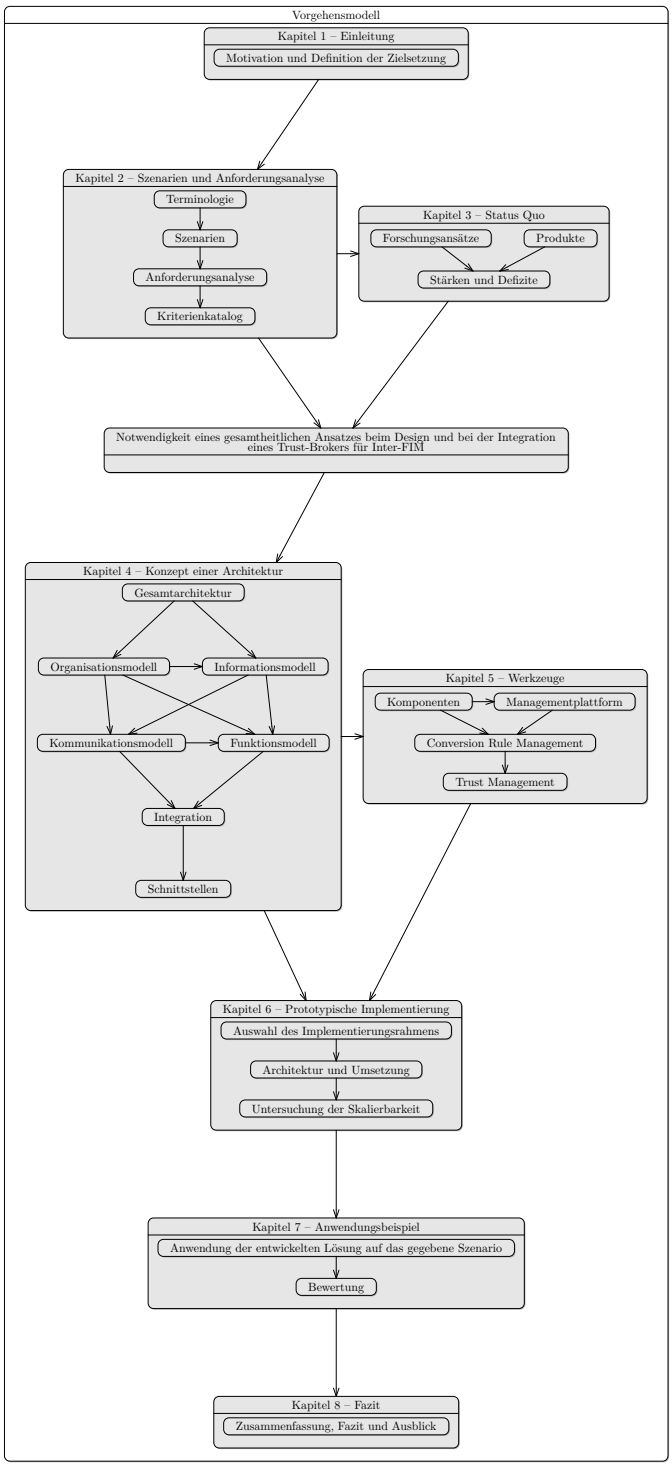

Abbildung 1.3.: Das Vorgehensmodell für diese Arbeit

in den Forschungskontexte einzuordnen und die Defizite der *bisherigen Lösungen* herauszuarbeiten. Dahingegen wurden einzelne Teilaspekte in existierenden Ansätzen der *Forschung* sowie im GÉANT-Umfeld bereits gelöst, so dass die vorliegende Arbeit diese Erkenntnisse miteinbeziehen kann. Ferner werden Implementierungen aus der *Wirtschaft* betrachtet. Die unterschiedlichen Ansätze werden anhand des in Kapitel 2 definierten Anforderungskatalogs hinsichtlich ihrer Stärken und Schwächen bewertet. Dieses Kapitel motiviert die Notwendigkeit dieser Arbeit, da die bisherigen Lösungen die Problematik nur unzureichend lösen können.

Um die im vorherigen Kapitel gefundenen Mängel zu beheben, werden in Kapitel 4 die für die *Architektur* nötigen Modelle, d. h. *Organisationsmodell*, *Informationsmodell*, *Kommunikationsmodell* und *Funktionsmodell*, erarbeitet. Dabei werden die Auswirkungen auf die vorhanden *Schnittstellen* Sicherheitsinfrastrukturen sowie Change Management diskutiert. Beim Zusammenspiel der Einzelkomponenten wird die Skalierbarkeit berücksichtigt.

In Kapitel 5 werden Werkzeuge spezifiziert, die für die Automatisierung und Flexibilisierung der Prozesse notwendig sind. Darunter fallen die *Trusted Third Party*, die *Umsetzung der Modelle* sowie die *Verwaltung der Konvertierungsregeln*, die die Wiederverwendbarkeit der Regeln sicherstellt. Das *Trust Management* sorgt für die Einbeziehung unterschiedlicher LoAs.

Die erarbeiteten Konzepte werden in Kapitel 6 *prototypisch implementiert* und im darauf folgenden Kapitel 7 beispielhaft auf ein *realistisches Anwendungsbeispiel* angewandt. Dies dient zum Nachweis der Lösung der oben genannten Fragestellungen. Abschließend werden in Kapitel 8 die Ergebnisse dieser Arbeit zusammengefasst und ein *Ausblick* auf weitere verwandte Fragestellungen gegeben.

1.4. Schwerpunkt dieser Arbeit und Publikationen

Nachdem Teile der Arbeit vorab veröffentlicht wurden, werden diese im Folgenden chronologisch aufgezählt. Ein Großteil der Publikationen wurde im Rahmen des Projekt GÉANT-TrustBroker (GNTB) veröffentlicht, welches als OpenCall Projekt eine Laufzeit von Oktober 2013 bis März 2015 hatte und anschließend als eigenständiger Task im GÉANT Projekt GN4 Phase 1 (Mai 2015 bis April 2016) weitergeführt wurde. Die Autorin der vorliegenden Arbeit arbeitete zunächst im OpenCall Projekt, bevor sie den Task leitete. Im Projekt wurde eine zentrale Instanz namens GNTB etabliert, um dynamisch Metadaten zwischen IdP und SP auszutauschen. Dabei wurde ein zusätzliches Conversion Rule Repository für die SAML-Implementierung Shibboleth implementiert, um einfach Konvertierungsregeln austauschen zu können. Im Projektantrag [Hom13] ist dies folgendermaßen beschrieben:

> *"The aim of Géant-TrustBroker is the specification of a new service for large-scale authentication and authorization infrastructures, i. e., federations and inter-federation scenarios (e. g., eduGAIN). Géant-TrustBroker will allow users (not*

> *only site administrators) to initiate the first-time contact between service providers (SPs) and the users' identity providers (IDPs) in order to perform the required preparations for identity data exchange in a fully automated manner. Géant-TrustBroker will also solve the real-world challenge of interfederation identity data transformation by hosting a smart data conversion rule repository. Making use of Géant-TrustBroker will be integrated into SAML workflows, so it can be used with widely deployed software packages, e.g., Shibboleth, and the protocols for accessing the Géant-TrustBroker will be submitted for standardization to the IETF or OASIS."* [Hom13]

Die vorliegende Arbeit hat mit dem Projekt gemeinsam, dass eine zentrale Instanz etabliert wird, die dynamisch Metadaten austauschen soll. Zudem besitzen beide ein Werkzeug zum Austausch von Metadaten. Im Gegensatz zum Projekt wird in dieser Arbeit von einer weltweiten Sicht ausgegangen. Basierend darauf wird ein generisches Konzept sowie eine Architektur entwickelt, die es ermöglicht Metadaten, Vertrauen und Konvertierungen weltweit auszutauschen. Diese Managementarchitektur ist nicht auf ein einzelnes Protokoll spezialisiert, sondern betrachtet die Problematik generisch. Zur Verwaltung wird eine Managementplattform etabliert, die über weitere Funktionen verfügt. Ein Bestandteil davon sind die Konvertierungsregeln, die ein generisches Schema aufweisen, um in verschiedenen Implementierungen und Protokollen verwendet werden zu können. Zudem verfügt die Managementplattform über ein Trust Management, um den Metadatenaustausch sicherer zu gestalten.

Zunächst werden die wissenschaftlichen Veröffentlichungen chronologisch beschrieben, bevor Projektdokumentationen aufgezeigt werden. Die Beschreibung enthält die Zusammenfassung der Veröffentlichung, den Anteil der Autorin dieser Arbeit und eine Erklärung, wie diese Publikation in die vorliegende Arbeit passt.

- Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *Géant-TrustBroker: Dynamic, Scalable Management of SAML-Based Inter-Federation Authentication and Authorization Infrastructures* [PMH14b]: Die Veröffentlichung konzentriert sich auf das Konzept von dynamischem Metadatenaustausch und dem damit verbundenen Core Workflow. Die grundlegende Idee von dynamischen Metadatenaustausch stammt von Wolfgang Hommel und wurde durch die Autorin dieser Arbeit und Stefan Metzger ausgearbeitet. Die grundlegenden Workflows wurden durch die Autorin dieser Arbeit spezifiziert, die anschließend zusammen mit Stefan Metzger und Wolfgang Hommel verfeinert wurden. Das Konzept von dynamischen Metadatenaustausch wird für die vorliegende Arbeit verwendet und ausgebaut.

- Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *A SAML Metadata Broker for Dynamic Federations and Inter-Federations* [PMH14a]: Diese Veröffentlichung vergleicht den Ansatz eines SAML Metadaten-Brokers mit dem State of the Art und beschreibt das Datenmodell einer TTP, die für den Metadatenaustausch zuständig ist. Darauf aufbauend werden eine Application Programming Interface (API) und ein Repository für Konvertierungsregeln skizziert. Sowohl State of the Art als auch das

Datenmodell und API wurden hauptsächlich durch die Autorin dieser Arbeit erstellt.

- Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *Géant-TrustBroker: Simplifying Identity & Access Management for International Research Projects and Higher Education Communities* [PMH14c]: Diese Veröffentlichung beschreibt hauptsächlich die Funktionalität und Workflows von GÉANT-TrustBroker. Die grundlegenden Workflows wurden durch die Autorin dieser Arbeit spezifiziert, die anschließend zusammen mit Stefan Metzger und Wolfgang Hommel verfeinert wurden. Die Funktionalität wurde iterativ durch die Autoren festgelegt. Sowohl Funktionalität als auch Workflows finden sich in dieser Arbeit wieder. Sie wurde auf ein weltweites dynamisches Föderationsmanagement angepasst und auf Grund einer neu etablierten Managementarchitektur erweitert.

- Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *Project GÉANT-TrustBroker – dynamic identity management across federation borders* [PMH14g]: Diese Veröffentlichung beschreibt neben dem grundsätzlichen Konzept des dynamischen Metadatenaustausches, den genauer spezifizierten Workflows und der Datenbank die Umsetzung in ein Protokoll. Dieses Protokoll wurde federführend durch Daniela Pöhn entwickelt und mit Hilfe von Stefan Metzger und Wolfgang Hommel optimiert.

- Wolfgang Hommel, Stefan Metzger und Daniela Pöhn: *Dynamic virtual federations with GÉANT-TrustBroker – Closing the gap between NREN federations and eduGAIN* [HMP15]: Dieser Extended Abstract gibt einen Überblick über das Projekt GÉANT-TrustBroker, welches hauptsächlich durch die Autorin und Stefan Metzger bearbeitet wurden. Diese Arbeit baut darauf auf und erweitert den Ansatz mit einer Managementarchitektur und einem Trust Management.

- Wolfgang Hommel, Michael Grabatin, Stefan Metzger und Daniela Pöhn: *DAME: On-demand Internet-scale SAML Metadata Exchange* [GHMP15]: Das Journal Paper beschreibt das Ergebnis des GÉANT-TrustBroker Projektes innerhalb der Projektphase GN3plus und zwei weitere Aspekte. Die Implementierung wurde durch Michael Grabatin realisiert, Stefan Metzger erklärt das Risk Management, während die Autorin dieser Arbeit die Konzepte der Föderationsverwaltung und dynamischen virtuellen Föderationen aufzeigt.

- Daniela Pöhn: *Topology of Dynamic Metadata Exchange via a Trusted Third Party* [Pöh15]: Diese Veröffentlichung beschreibt die Topologie von dynamischem Metadatenaustausch anhand des Munich Network Management (MNM)-Dienstmodells und erweitert, basierend auf der Auswahl, den Core Workflow. Diese Art des verteilten Metadatenaustausches wird in der vorliegenden Arbeit neben dem einfachen dynamischen Metadatenaustausch im Ausblick erläutert.

- Daniela Pöhn: *Risk Management for Dynamic Metadata Exchange via a Trusted Third Party* [Pöh16b]: Diese Veröffentlichung beschreibt das Security Management für die TTP, welches im Kapitel 4 angewandt wird.

- Daniela Pöhn: *Architecture and Concepts for Federated Identity Management with Federations and Inter-federations* [Pöh16a]: Diese Veröffentlichung für das Doctorial Consortium DCISSP beschreibt das Thema dieser Arbeit.

- Wolfgang Hommel und Daniela Pöhn: *Management Architecture for Dynamic Federated Identity Management* [HP16]: Diese Veröffentlichung beschreibt basierend auf dem Service Modell für FIM und dynamisches FIM die Managementarchitektur mit ihren Modellen. Als Beispiel für das Funktionsmodell wird das hier in dieser Arbeit kurz erwähnte Service Management ausführlicher erläutert. Die Managementarchitektur wurde durch Daniela Pöhn entwickelt und mit Hilfe von Wolfgang Hommel in der Veröffentlichung dargestellt.

- Michael Grabatin, Wolfgang Hommel, Stefan Metzger und Daniela Pöhn: *Improving the Scalability of Identity Federations through Level of Assurance Management Automation* [GHMP16]: Diese Veröffentlichung erläutert die Grundlage für die dynamische Überprüfung von verschiedenen Aspekten der Level of Assurance. Dieses Konzept wurde durch die Autorin dieser Arbeit erarbeitet und anschließend in Zusammenarbeit mit Wolfgang Hommel verfeinert. Die Basis für das in dieser Arbeit verwendete Konzept wurde insbesondere mit Hilfe von Wolfgang Hommel in der Veröffentlichung dargestellt.

- Daniela Pöhn und Wolfgang Hommel: *Automated User Information Conversion to improve Identity Federation Scalability* [PH16]: Diese Veröffentlichung behandelt die Automatisierung von Konvertierungsregeln in einem generischen Format. Im Gegensatz zum Shibboleth-spezifischen Repository sollen hier generische einfache Regeln ermöglicht werden, um den IdP-SP-Verbindungsaufbau zu beschleunigen und zu automatisieren. Dieses Konzept wird im Werkzeug Conversion Rule Repository eingesetzt.

Die Projektdokumentationen dokumentierten chronologisch Bestandteile des Projektes als Milestone Documents oder Deliverables.

- Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *Milestone M.1.1.1: Requirements analysis of Géant-TrustBroker* [PMH13a]: Diese erste Projektdokumentation beschreibt anhand der verschiedenen Workflows die Anforderungsanalyse. Die grundlegenden Workflows wurden durch die Autorin dieser Arbeit spezifiziert, die anschließend zusammen mit Stefan Metzger und Wolfgang Hommel verfeinert wurden. Stefan Metzger bearbeitete hauptsächlich die Anforderungsanalyse. Sowohl Funktionalität als auch Workflows finden sich in dieser Arbeit wieder. Sie wurde auf ein weltweites dynamisches Föderationsmanagement angepasst und auf Grund einer neu etablierten Managementarchitektur erweitert.

- Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *Milestone M.1.2.1: Géant-TrustBroker standardisation roadmap* [PMH13b]: Diese Projektdokumentation erläutert das Vorgehen bei der Standardisierung des Core Workflows und wurde von der Autorin dieser Arbeit geschrieben.

- Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *Milestone Document M.2.1.1: GÉANT-TrustBroker protocol specification written* [PMH14d]: Die Projektdokumentation beschreibt die erste Version des Internet-Drafts an die Standardisierungsorganisation Internet Engineering Task Force (IETF). Die erste Version des Internet-Drafts (I-Ds) und des Dokuments wurden von der Autorin der vorliegenden Arbeit geschrieben und durch aktive Mitarbeit von Stefan Metzger und Wolfgang Hommel verbessert. Der bei der Erstellung der Arbeit aktuelle I-D wurde bei der Implementierung und als Grundlage des dynamischen Metadatenaustausches verwendet und erweitert.

- Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *Open Call Project Deliverable D.2.1.1: Géant-TrustBroker Specification* [PMH14f]: Diese Veröffentlichung dokumentiert die Spezifikation des GÉANT-TrustBrokers mit Datenmodell, API, Workflows, Conversion Rule Repository und dem Protokoll. Während das Dokument unter Regie von Stefan Metzger entstand, war die Arbeit wie folgt aufgeteilt: Datenmodell, Workflows wurden hauptsächlich durch die Autorin dieser Arbeit erarbeitet, Stefan Metzger spezifizierte das Conversion Rule Repository, während die erste Version des Protokolls der Autorin der vorliegenden Arbeit durch aktive Mitarbeit von Stefan Metzger und Wolfgang Hommel verbessert wurde.

- Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *Milestone M.4.1.1: TrustBroker service demonstrator* [PMH14e]: Diese Projektdokumentation visualisiert den Demonstrator des Projekts GNTB. Das Dokument wurde hauptsächlich durch Stefan Metzger geschrieben, während Michael Grabatin den Demonstrator federführend implementierte. Die Autorin dieser Arbeit half dabei, beides zu optimieren.

- Daniela Pöhn, Michael Grabatin, Stefan Metzger, David Schmitz und Wolfgang Hommel: *Deliverable OCJ-DS4.1.1 Open Call Deliverable - GÉANT-TrustBroker implementation with documentation* [PGM+15]: Dieses Deliverable dokumentiert die Implementierung des GÉANT-TrustBrokers, die hauptsächlich durch Michael Grabatin in Rahmen seiner Masterarbeit [Gra14] realisiert wurde. Eine erweiterte Implementierung wird in der vorliegenden Arbeit verwendet, um die Realisierbarkeit zu beweisen.

- Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *Deliverable OCJ DS2.2.1 Open Call Deliverable GÉANT-TrustBroker protocol specification* [PMH15]: Diese Projektdokumentation basiert auf dem Milestone Document M.2.1.1 und beschreibt die zum Projektabschluss aktuelle Version des I-Ds, der federführend durch die Autorin der vorliegenden Arbeit erstellt und durch Stefan Metzger und Wolfgang Hommel fachlich unterstützt wurde.

- Remco Poortinga-van Wijnen und Daniela Pöhn: *Deliverable D15.3 Operational GÉANT Trust Broker Pilot Instance* [PvWP16]: Diese Projektdokumentation beschreibt den erweiterten Prototypen, der für einen möglichen operativen Einsatz im GÉANT-Umfeld verwendet werden kann. Dieses Dokument wurde federführend durch die Autorin der vorliegenden Arbeit erstellt, während Michael Grabatin insbesondere für die Programmierung zuständig war. Remco Poortinga-van Wijnen hat die Projektleitung

der Activity übernommen.

Diese Aufzählung der Vorveröffentlichungen zeigt, dass manche Ideen, Abbildungen und Bezeichnungen in vorab erschienen Publikationen präsentiert wurden. Dies ist insbesondere in den Kapiteln 3,4, 5 und 6 der Fall. In Kapitel 3 wird der aktuelle Stand bezüglich Forschung, praktischen Ansätzen, Level of Assurance und Protokollen aufgezeigt. Der in dieser Arbeit beschriebene State of the Art bildete die Grundlage in den oben genannten wissenschaftlichen Veröffentlichungen. Kapitel 5 erweitert den im GÉANT-TrustBroker und in den gemeinsamen Veröffentlichungen beschriebenen Ansatz, wie oben beschrieben. Auf Grund der Managementplattform und ihrer Werkzeuge werden Workflows und Datenhaltung erweitert. Die Trusted Third Party ist die Managementplattform der Managementarchitektur, die weitere Funktionalitäten bereitstellt, wie beispielsweise ein Trust Management, Unterstützung von dynamisch virtuellen Föderationen und eine Föderationsverwaltung. Das Conversion Rule Repository, welches hilft Benutzerinformationen zu konvertieren, wird generisch angelegt, um mehrere Implementierungen und auch Protokolle zu unterstützen. Die Implementierung des GÉANT-TrustBrokers dient als Grundlage die Realisierbarkeit des Ansatzes in Kapitel 6 zu beweisen. Dazu wurde die Projektimplementierung um die Werkzeuge generisches Conversion Rule Management und Trust Management erweitert. Im jeweils zugehörigen Abschnitt dieser Arbeit wird dieses Zusammenspiel detailliert ausgeführt.

1.5. Abgrenzung zu verwandten Forschungsarbeiten

Verschiedene Arbeiten sowohl des MNM-Teams als auch in anderen akademischen und industriellen Forschungseinrichtungen sind die Voraussetzung für die Anwendung des Federated Identity Managements. Im Folgenden werden weitere Arbeiten kurz vorgestellt, die einen engen Zusammenhang mit dieser Arbeit haben:

- Michael Grabatin [Gra14] beschreibt in seiner Masterarbeit Federated Identity Management eine Trusted Third Party. Die vorliegende Arbeit geht über diese u. a. durch die Autorin betreute Arbeit hinaus und definiert verschiedene Managementfunktionalitäten, ein Trust Management sowie generische Konvertierungsregeln.

- Wolfgang Hommel [Hom07] untersucht in seiner Dissertation Federated Identity Management. Die vorliegende Arbeit setzt im Gegensatz zu Hommels Arbeit nicht auf vorhandene Vertrauensbeziehungen voraus. Hommel stellt die Integration von FIM in die lokalen Systeme dar und bildet somit die logische Basis für die vorliegende Arbeit.

- Helmut Reiser [Rei08] definiert in seiner Habilitation ein Framework für föderiertes Sicherheitsmanagement. Hierbei wird zusätzlich Federated Identity Management eingesetzt, um ein effizientes verteiltes Management von Benutzern und Berechtigungen auf Applikationsebene zu erreichen. Reiser baut somit auf den in Hommels Arbeit entstandenen Lösungen auf und bettet sie in einem Rahmenwerk ein.

- Latifa Boursas [Bou09] beschreibt in ihrer Dissertation Trust und Reputation Management in föderierten, dynamischen Umgebungen. In der vorliegenden Arbeit wird die zusätzliche Verwendung von Metadaten vorausgesetzt, wodurch Boursas Ergebnisse als ergänzende Vertrauensebene eingesetzt werden können.

- Michaels Schiffers [Sch07] untersucht in seiner Dissertation das Management dynamischer virtueller Organisationen im Rahmen des Grid Computings. Virtuelle Organisationen sind bezüglich des Identity Managements mit Föderationen vergleichbar.

- Mikael Linden [Lin09] analysiert in seiner Dissertation Federated Identity Management, u. a. am Beispiel der finnischen Föderation Haka. Er legt, ähnlich wie Wolfgang Hommel, die logische Basis für die vorliegende Arbeit.

Kapitel 2

Szenarien und Anforderungsanalyse

Inhalt dieses Kapitels

Auf Grund der Tatsache, dass Inter-FIM auf I&AM und FIM aufbaut, werden zum Verständnis dieser Arbeit die grundlegenden Aspekte der folgenden Technologien in den Abschnitt 2.1 und 2.2 erklärt:

Identity & Access Management: I&AM ist die Weiterentwicklung des organisationsinternen User Managements zur Nutzung gemeinsamer Datenbestände und zur Unterstützung vorhandener Geschäftsprozesse.

Federated Identity Management: FIM dient dem darauf aufbauenden, organisationsübergreifenden Austausch von Benutzerinformationen.

Hierfür gibt es bereits ausgereifte Konzepte und technische Lösungen, jedoch ist ein Verständnis dieser Technologien grundlegend für diese Arbeit. Zusätzlich werden allgemeine Anforderungen, die sich aus aktuell vorhandenen Föderationen ableiten, von Wolfgang Hommels Dissertation [Hom07] (Kapitel 2) übernommen. Ferner wird in 2.2.4 das MNM-Dienstmodell auf FIM bezogen erklärt, um es bei den einzelnen Szenarien und später in Kapitel 5 erneut aufzugreifen.

Inter-Federation Identity Management ist eine noch relativ junge Disziplin, wodurch bislang viele Begriffe noch nicht einheitlich definiert sind. In Abschnitt 2.3 werden daher zur Begriffsbildung die grundlegenden technischen und organisatorischen Aspekte des Inter-FIMs erläutert. Inter-FIM ist die Expansion von FIM auf mehrere Föderationen, so dass der Austausch von Benutzerinformationen über Föderationsgrenzen hinaus möglich ist. Für Inter-FIM existieren ebenfalls bereits Konzepte und Lösungen, jedoch sind diese noch nicht ausgereift.

Um die bereits existierenden Lösungsansätze für Federated und Inter-Federated Identity Management miteinander zu vergleichen sowie deren Stärken und Schwächen herausarbeiten zu können, wird in diesem Kapitel ein Kriterienkatalog definiert. Dessen Komponenten werden dafür aus konkreten Szenarien abgeleitet. Das Identitätsmanagement in den *Communities*, welches sich in den Communities stark untereinander von den eingesetzten Technologien

und Anforderungen unterscheiden, wird im Anschluss ebenfalls beschrieben. Eine grundlegende Betrachtung der Wirtschaft und des *User-Managed Access* im so genannten *User Centric Identity Management (UCIM)*, welches die Einbindung des Nutzers betont, runden den Überblick ab. Die Themenbereiche Inter-FIM (Abschnitt 2.3.6), Wirtschaft (Abschnitt 2.5), Forschungsgruppen mit VOs in Grid sowie die Service Provider Federation (SP-Federation) bei Common Language Resources and Technology Infrastructure (CLARIN) (Abschnitt 2.4) und UMA (Abschnitt 2.6) werden jeweils anhand von Szenarien veranschaulicht. Bei der Auswahl und Darstellung der Szenarien wird insbesondere, wie in Tabelle 2.1 zu sehen, auf ein breites Spektrum an Anforderungen und Realitätsnähe geachtet.

Art	Ausprägung	
FIM in NRENs	Föderation in NRENs	Inter-Föderation eduGAIN
FIM in VOs	Grid-Umfeld	SP-Föderation von CLARIN
FIM in der Wirtschaft	Sektorübergreifendes, fiktives Szenario	
FIM aus Nutzersicht	UMA	

Tabelle 2.1.: Auswahl an Föderationen für die Szenarien

Die Szenarien sind unter folgenden Gesichtspunkten zusammengestellt worden:

- Die praktische Notwendigkeit von dynamischen virtuellen Föderationen sowie einer Architektur zur Automatisierung etablierter Workflows wird verdeutlicht.
- Die Szenarien decken ein breites Spektrum an Anforderungen ab, indem sie die Sichtweise von Beteiligten an Föderationen und Inter-Föderationen aus dem Hochschulumfeld, UCIM, Forschungsgruppen und aus der Wirtschaft aufzeigen.
- Alle Eigenschaften und Anforderungen an die Komponenten zur Automatisierung und zur Etablierung von dynamischen virtuellen Föderationen lassen sich anhand der Szenarien veranschaulichen.

Zwar bietet keines der Szenarien eine schnelllebige Struktur, wie sie bei dynamischen virtuellen Föderationen möglich ist, trotzdem werden die wichtigen Eigenschaften und Anforderungen durch die Szenarien abgedeckt. So wird durch das Inter-FIM Szenario u. a. ein langfristiger Ansatz aufgezeigt, der die Basis für dynamische Kooperationen darstellen soll. Im Anschluss an die Szenarien folgt die jeweilige Ableitung der Anforderungen an die eingesetzte Technologie. Hierbei wird Wert auf die Integration in die bereits existierende Infrastruktur gelegt.

Die ermittelten Anforderungen werden im Abschnitt 2.7 zusammengefasst und um zusätzliche Aspekte ergänzt, die teilweise aus den aktuellen Forschungsarbeiten in Abschnitt 3 abgeleitet werden können. Die Anforderungen werden anschließend gewichtet und in tabellarischer Form als *Anforderungskatalog* in Abschnitt 2.8 dargestellt, so dass bei der Bewertung aktueller Forschungsansätze und der zu entwickelten Architektur darauf Bezug genommen werden kann. Die gewählte Vorgehensweise des Kapitels wird in Abbildung 2.1 graphisch

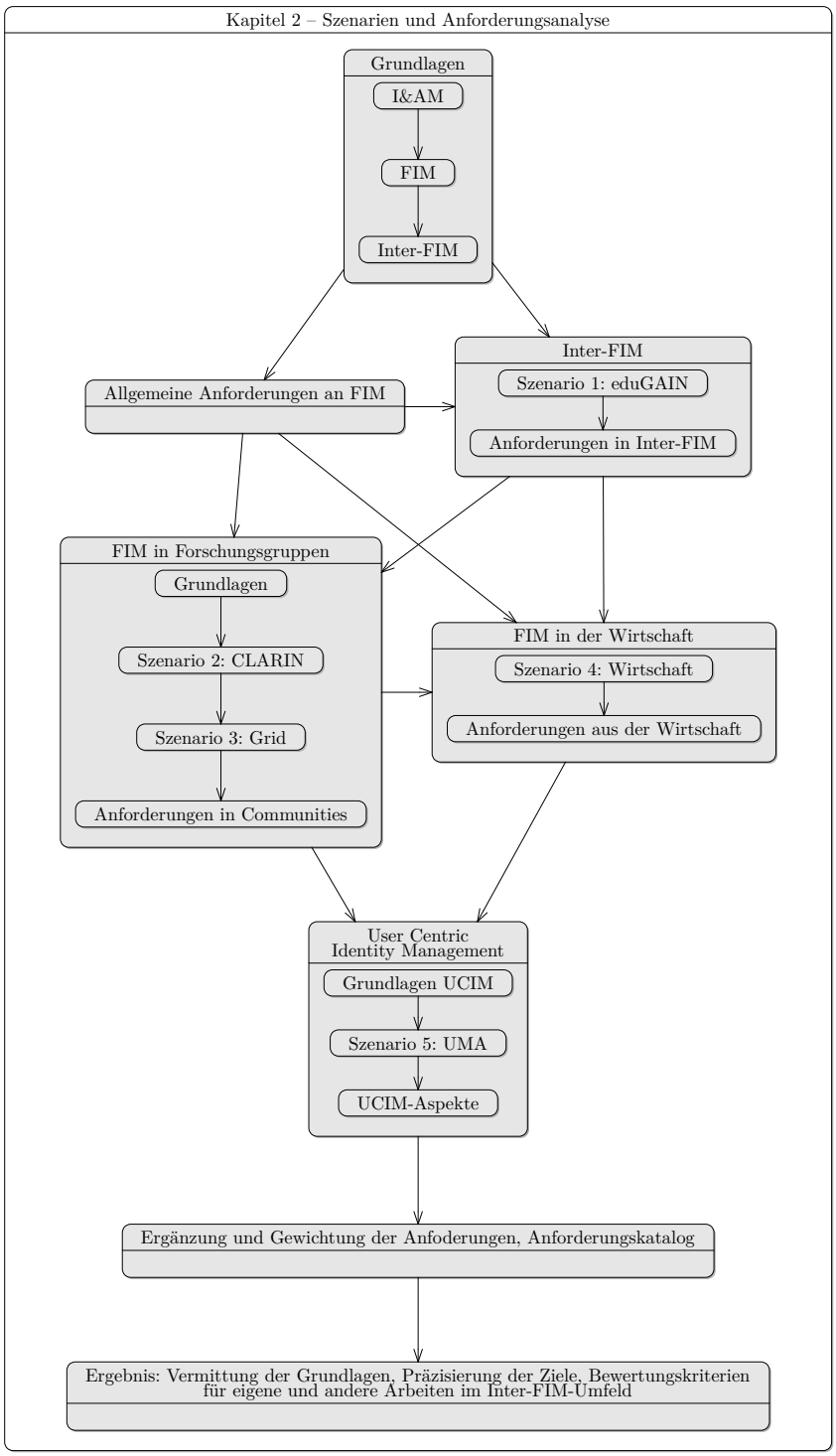

Abbildung 2.1.: Vorgehensmodell in diesem Kapitel

dargestellt.

2.1. Identity & Access Management

Definition 1. Identity Access Management

Identity & Access Management ist die lokale Komponente für das Identifizieren, Authentifizieren und Autorisieren von Nutzern in der Heimatorganisation.

Das *Identity & Access Management* behandelt das Identifizieren, Authentifizieren und Autorisieren von Nutzern. In der Heimatorganisation existiert ein zentraler Datenbestand, Identity Repository (IR) genannt, mit allen relevanten Benutzerdaten und Berechtigungen zu den eigenen Systemen (vgl. [Hom07]). Die Speicherung der Benutzerdaten wird über *Verzeichnisdienste* gelöst, in der Praxis meist LDAP-basierte Enterprise Directory Services, da LDAP auch ein Transmission Control Protocol (TCP)/Internet Protocol (IP)-basierte Request-Response-Protokoll für die Kommunikation zwischen Client und Server definiert, was die Interoperabilität zwischen verschiedenen LDAP-fähigen Komponenten garantiert. Einträge und Berechtigungen werden meist nicht direkt angelegt, sondern indirekt über *autorative Datenquellen*, wie beispielsweise Personalverwaltungssystemen. Falls eine Person multipel gespeichert wurde, müssen die Daten zunächst aggregiert und korreliert werden. Der Datenabgleich zwischen den verschiedenen Datenquellen und Datenabnehmern findet über *Konnektoren* statt.

I&AM konzentrieren sich somit auf die Datenhaltung der organisationsinternen Identity-Daten und die Anbindung daran. Identitäts-Management-Systeme reichern die Informationen aus den Verwaltungssystemen mit weiteren Informationen wie Benutzername und Passwort an und verteilen die daraus entstandenen digitalen Identitäten an organisationsinterne Dienste. Durch das zentrale Vorhalten der Benutzerdaten stellt das I&AM sicher, dass alle Dienste dieselben, konsistenten Informationen benutzen. Für den Nutzer bietet das I&AM den Vorteil nur einen Benutzernamen für alle organisationsinternen Dienste zu benötigen. Die zentrale Datenhaltung kann gebündelt für mehrere Organisationen geschehen, wie im IntegraTUM-Projekt [HKP+08] bei der Etablierung eines I&AM für die Technische Universität München (TUM) und das Leibniz-Rechenzentrum (LRZ) zu sehen. Wichtig für die Realisierung sind technische Schnittstellen zwischen den verschiedenen Systemen.

Durch das Vorhalten von personenbezogenen Daten auf dem zentralen System ist der Datenschutz ein wichtiger Aspekt im I&AM, der gesetzlichen Auflagen unterliegt:

Datensparsamkeit. Die Service Provider sind zur Datensparsamkeit angehalten.

Einverständnis. Bevor Informationen über den Benutzer an einen Dienst gesendet werden, muss das Einverständnis des Betroffenen eingeholt werden.

Selbstauskunft. Der Anwender soll die Möglichkeit haben, stets eine Selbstauskunft zu bekommen. Diese beinhaltet das Einsehen der gespeicherten Daten und die Information über deren Verarbeitung durch die eingesetzten Systeme.

Korrektur. Der Anwender muss falsche Daten korrigieren können.

Einverständnis widerrufen. Sein Einverständnis muss der Nutzer jederzeit widerrufen können.

Dokumentation und Zweckmäßigkeit. Der Datenschutz muss dokumentiert werden und die Verarbeitung der Daten muss zweckmäßig erfolgen.

2.2. Federated Identity Management

Definition 2. Federated Identity Management

Federated Identity Management sind Management-Architekturen, die verteilte - föderierte - Benutzerverwaltung realisieren.

Während Identity & Access Management lokal sehr gut funktioniert, musste eine Lösung bei Kooperationen, beispielsweise wenn Nutzer der Organisation A wegen eines Projekts auf das Wiki der Organisation B zugreifen wollen, gefunden werden. In einfachen Fällen können die Nutzer von Organisation A in das I&AM der Organisation B übernommen werden, was zu erhöhtem Pflegeaufwand führt. Wenn mehrere Entitäten kollaborieren, ist die Variante des *Federated Identity Managements* effizienter und besser skalierbar, wie bereits Wolfgang Hommel [Hom07] (Abschnitt 4.11) in seiner Dissertation beschrieben hat. FIM bezeichnet Management-Architekturen, die eine verteilte Benutzerverwaltung ermöglichen. Jeder Benutzer ist dabei mindestens einer Heimatorganisation, Identity Provider genannt, zugeordnet. Die Anbieter externer Dienste werden als Service Provider bezeichnet, die über FIM-Protokolle Benutzerinformationen abrufen können. Aus Verständnisgründen werden die Grundprinzipien kurz beschrieben.

Im Federated Identity Management werden *organisationsübergreifend* personenbezogene Daten übermitteln, wobei sowohl die beteiligten Organisationen als auch die Benutzer davon profitieren. Zum einen wird für die Organisationen die Datenqualität erhöht, indem nur an einem zentralen Punkt die benutzerbezogenen Informationen gepflegt werden müssen. Gleichzeitig wird die Effizienz gesteigert, was wiederum die Benutzerfreundlichkeit steigert. Der Benutzer muss sich nur das Benutzerkonto seiner Heimatorganisation merken und hat nicht für jede Kooperation zusätzliche Benutzerkonten. Durch SSO wird eine einzige Passworteingabe benötigt, um auch auf Dienste von anderen Organisation zuzugreifen. Ferner ist der Datenschutz leichter einzuhalten.

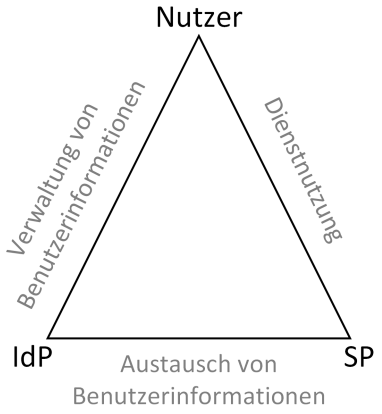

Abbildung 2.2.: Dreieck aus Nutzer, Service Provider und Identity Provider

2.2.1. Rollen im Federated Identity Management

Somit ergibt sich ein Dreieck aus Nutzer und den beteiligten Organisationen, aufgeteilt in die Rollen Identity Provider und Service Provider, wie in Abbildung 2.2 zu sehen. Dabei kann eine Organisation sowohl die Rolle des Identity Provider als auch die des Service Provider einnehmen.

Nutzer: Die Nutzer gehören einer oder mehreren Organisationen an. Die Informationen über einen Nutzer sowie dessen Berechtigungen sind als Attribute in seiner Heimatorganisation gespeichert. Attribute bilden Tupel aus Attributnamen und Wertemenge. Manche Attribute sind verpflichtend (*mandatory*), wie beispielsweise Benutzername, andere sind hingegen optional. Der Nutzer ist einer oder mehreren Rollen zugeordnet, welche parametrisiert sind. Je nach Rolle werden dem Nutzer zusätzliche, bestimmte Attribute zugeschrieben. Damit der Nutzer erfolgreich einen Dienst nutzen kann, sendet die Heimatorganisation dem Service Provider eine Teilmenge an Benutzerinformationen.

Identity Provider (IdP): Der Identity Provider stellt die autorative Quelle für Authentifizierungsbestätigung und in der Regel die primitive Quelle für Autorisierungsbestätigung dar. Ferner ist der IdP für allgemeine Attributsauskünfte zuständig. Jeder Nutzer besitzt mindestens einen Identity Provider. Je nach Anzahl bzw. Architektur der IdPs in einer Föderation besteht ein zentrales oder dezentrales FIM. In einem zentralen FIM existiert ein einziger IdP, der die Identitäten aller Nutzer einer Föderation verwaltet. Im Gegensatz dazu befinden sich in einem dezentralen FIM, wie in den meisten Föderationen im Research & Education (R&E)-Umfeld, mehrere Identity Provider.

Service Provider (SP): Der Service Provider bietet einen oder mehrere Dienste an. Damit der Nutzer den Dienst in Anspruch nehmen kann, sendet sein IdP bestimmte Informationen an den SP. Optional werden Benutzerinformationen von der nachfolgend

erläuterten AA abgerufen. Durch diesen Abruf von Attributen wird der Service Provider auch als identity data consumers bezeichnet. Die Akzeptanz der bezogenen Daten hängt vom Vertrauensverhältnis zwischen IdP und SP ab. Falls das Vertrauen nicht ausreicht, schlägt im primitivsten Fall die Nutzung des Dienstes fehl.

Neben diesen drei Hauptakteuren, können sowohl die bereits erwähnten Attribute Authorities als auch so genannte TTPs in der Interaktion eine Rolle spielen:

Attribute Authority (AA) Attribute Authorities sind immer dann erforderlich, wenn die Benutzerinformationen des Nutzers auf mehrere IdPs verteilt sind. Der wesentliche Unterschied zu Identity Providern besteht darin, dass AAs nicht zur Authentifizierung verwendet werden, sondern zusätzliche Benutzerinformationen liefern. Diese können zur Autorisierung herangezogen werden. Ferner sind die Nutzer organisatorisch stärker an ihren Identity Provider gebunden.

Trusted Third Party (TTP) Diese dritten Parteien, denen mehrere Teilnehmer vertrauen, bieten FIM-spezifische Dienste für die beteiligten Entitäten, aber nicht Nutzern, einer Föderation an. Dies kann zum Beispiel die Verwaltung der Metadaten sowie die Public Keys bzw. Public-Key-Infrastruktur (PKI)-Zertifikate der beteiligten Entitäten oder die Bestätigung der Korrektheit von Benutzerattributen sein. Die Dienste sind in der Mehrheit sicherheitskritisch.

2.2.2. Organisatorische Komponenten und Trust Management des Federated Identity Managements

Definition 3. Föderation

Föderation ist ein Zusammenschluss von mehreren Identity Providern, Service Providern und meist Trusted Third Parties zum Zweck des Federated Identity Managements.

Eine *Föderation* ist eine Einheit an Organisationen, die ein Auftraggeber-Auftragnehmer-Verhältnis haben, wodurch ein Service Provider Zugriff auf die Benutzerinformationen bekommt. Dafür muss untereinander eine geeignete Vertrauensbeziehung zum Zweck des FIM-basierten Datenaustausches aufgebaut werden. Die Qualität und Verfügbarkeit der Daten wird über vertragliche Vereinbarungen zwischen den Organisationen in den Föderationen geregelt. Die Organisationen haben ein gewisses Vertrauen untereinander und können sowohl die Rolle eines IdPs als auch eines SPs annehmen. Eine Föderation besteht aus mindestens einem Identity Provider und einem Service Provider. Diese Organisationseinheiten sind im Hochschulumfeld meist auf nationaler Ebene angesiedelt, wie beispielsweise die AAI des DFN-Verein, die laut [Ter16] aus 218 IdPs und 255 SPs innerhalb von Deutschland besteht.

Föderationen können unterschiedliche Strukturen aufweisen, wie Mikael Linden [Lin09]

aufzeigt:

Ad hoc federation. Bilaterale Kooperationen zwischen Organisationen, die einer Föderation beitreten möchten.

Hub-and-spoke federation. Eine große Organisation im Zentrum, die die Regeln der Föderationen zum Vorteil der eigenen Organisation bestimmt. Die weiteren Organisationen sind über transitives Vertrauen miteinander verbunden.

Identity network. Eine unabhängige Organisation fokussiert sich auf die technischen und administrativen Aspekte der Föderation. Im einfachsten Fall besteht zwischen allen Teilnehmern direktes Vertrauen, wodurch der *Circle of Trust* vollvermascht ist. Da diese Lösung schlecht skaliert, schließt häufig jede Organisation mit der TTP einen Vertrag und vertraut jeder anderen Organisation, die ebenfalls einen solchen Vertrag besitzt (*brokered trust*). Die TTP kann hierbei ein Konsortium sein.

Die meisten Föderationen im Hochschulumfeld in Europa haben die Ausprägung eines Identity networks, u. a. auch die DFN-AAI. Theoretisch wären auch andere Strukturen, wie vollvermaschte Föderationen möglich. Dies ist aber logistisch supoptimal. In einer Hub-and-spoke federation und in einem Identity network ist die Organisation in der Mitte für die organisatorischen Aspekte und für den Betrieb der für die Föderation notwendigen Infrastruktur oder Werkzeuge zuständig.

Zur Formalisierung der Vertrauensbeziehungen in Föderationen können einzelne SLAs abgeschlossen werden, die Bedingungen an die übermittelnden Daten, deren Qualität und Verfügbarkeit stellen. Beispielsweise schließt der DFN-Verein mit SPs und IdPs innerhalb der DFN-AAI Verträge, die u.a.

- die Qualitätsanforderungen an das Identity-Management,
- die Ausgestaltung der technischen Schnittstellen,
- den Austausch von Attributen
- und das Vorgehen bei Verstößen regeln.

Ferner setzt der DFN-Verein voraus, dass alle Personen, die in die Einrichtung aufgenommen werden, eine digitale Identität mit Attributen, die der Rolle der Person entsprechen, erhalten. Änderungen müssen zeitnahe erfolgen und das Attributschema der Föderation unterstützt werden. Außerdem müssen die Prozesse soweit schriftlich dokumentiert sein, dass das Sicherheitsniveau aus der Dokumentation ableitbar ist [DFN10]. Zudem können Partner einer bilateralen Kooperation ergänzende Verträge aushandeln, was insbesondere für kommerzielle Partner essentiell ist. Der IdP kann zusätzlich zu den in Abschnitt 2.1 beschriebenen Konditionen bestimmte Sicherheitsmaßnahmen fordern und Vorschriften bezüglich dem Umgang mit personenbezogenen Daten machen.

2.2.3. Klassifikation

Zur *Klassifikationen* von Föderationen wird die Morphologie aus der Dissertation von Michael Schiffers [Sch07] erweitert, die die Ausprägungen von virtuellen Organisationen beschreibt. Diese Morphologie bildet die Grundlage zur Klassifikation von Föderationen, wie in Abbildung 2.3 zu sehen. Verschiedene Aspekte sind zu Merkmalen und Ausprägungen logisch zusammengefasst, die teilweise abhängig voneinander sind.

Kooperation der Föderation. Dieses Merkmal beschreibt die Art der Kooperation, d. h. die *Kooperationsstruktur*, die *Anzahl der Teilnehmer* sowie die *Gruppenstruktur*. Die Anzahl der Teilnehmer ist abhängig von der Gruppenstruktur. So besitzen allgemein offene Gruppen keine feste Anzahl der Teilnehmer, während abgeschlossene Gruppenstrukturen sowohl einfache als auch bilaterale Kooperationen erlauben. Die Kooperationsstruktur Ad hoc Föderation wiederum bietet sich vor allem für allgemein offene, komplexe Netzwerke an, während ein Identity Network wie in den meisten existierenden nationalen Föderationen eine etwas festere Struktur benötigt. Die Kooperationsstruktur hat wiederum Auswirkungen auf die Art des Vertrauens.

Kooperationsstruktur. Die Struktur fügt der Bindungsintensität die Festlegung gemeinsamer Rahmenbedingungen hinzu. Sie kann die Ausprägungen *Ad hoc*, *Hub-and-spoke* sowie *Identity network* annehmen.

Anzahl der Teilnehmer. Die Größe der Kooperation ist ein entscheidender Aspekt bei des Informationsaustausches. Bei *bilateralen* Kooperationen können die Benutzerdaten noch bei beiden Organisationen vorgehalten werden, während *einfache* Netzwerke eine größere Reichweite haben. *Komplexe* Netzwerke sind mit aktuell vorhandenen Föderationen kaum abzubilden.

Gruppenstruktur. Föderationen bzw. Kooperationen können *allgemein offen* sein, wie bei OpenID Connect. Diese offenen Gruppenstrukturen weisen gleichzeitig einen hohen Grad an Dynamik auf. Der Beitritt weiterer Organisationen kann zugleich *mit Einschränkungen* bedacht sein, beispielsweise durch ein Genehmigungsverfahren der Föderationsverwaltung. Gleichzeitig gibt es *abgeschlossene* Föderationen, die keine dynamischen Elemente enthalten.

Dimensionen der Föderation. Die Dimension der Föderation gliedert sich auf in eine *räumliche* und eine *organisatorische* Komponente, die nicht voneinander abhängen müssen. Ein Beispiel, wo diese Abhängigkeit vorhanden ist, stellen nationale Föderationen wie die DFN-AAI und die internationale Inter-Föderation eduGAIN dar.

Räumliche Dimension. Kooperationen und somit auch Föderationen unterscheiden sich anhand ihrer räumlichen Ausdehnung. Neben *lokalen* Kooperationen gibt es *regionale*, *nationale* und *internationale* Zusammenarbeiten.

Organisatorische Dimension. Die Mikro-Ebene spiegelt die Betrachtungsweise einer

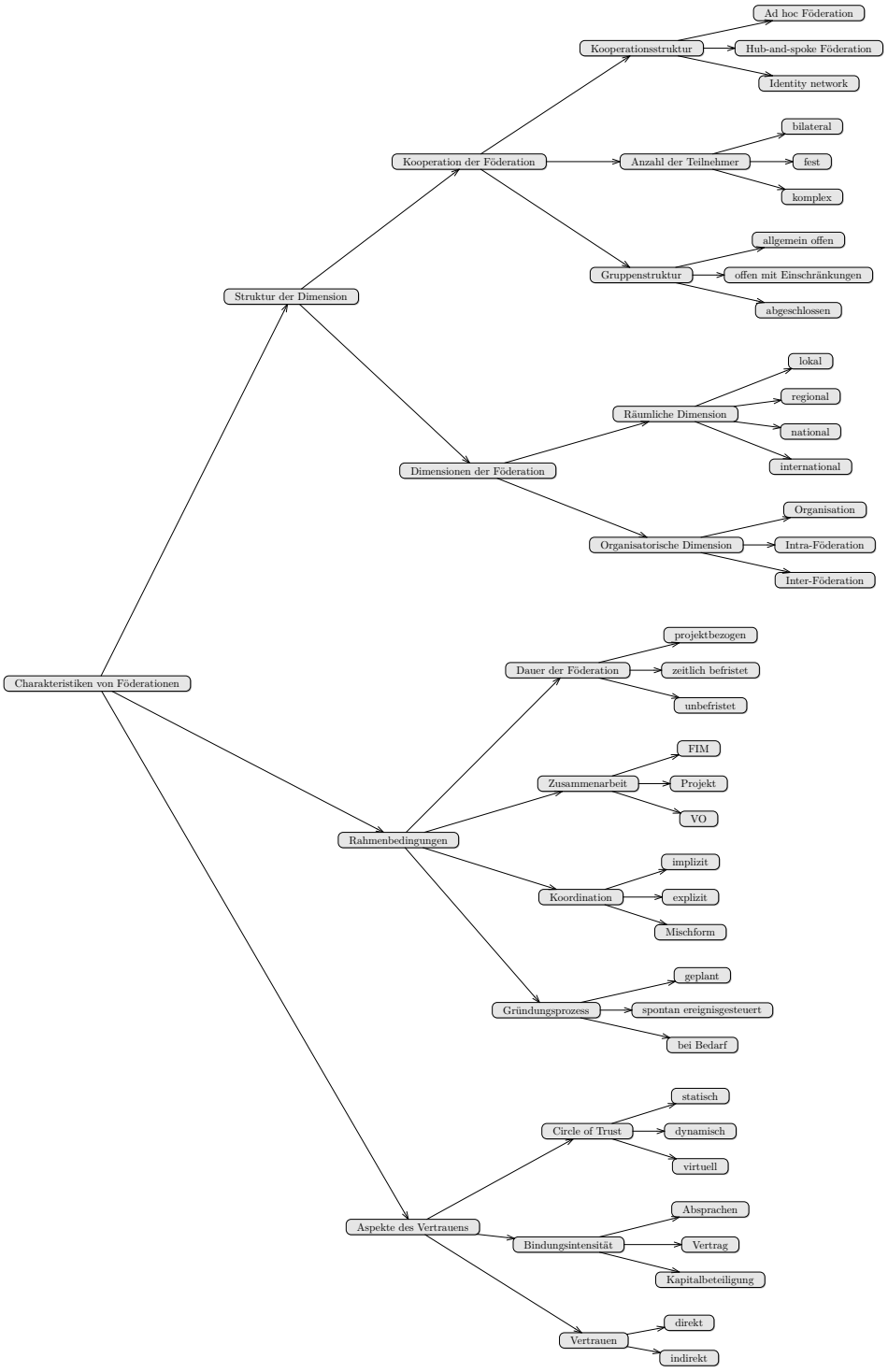

Abbildung 2.3.: Klassifikation von Föderationen

einzelnen *Organisation* wieder. Die meso-Ausprägung beschäftigt sich mit der Kooperation mehrerer Organisationen, d. h. *Föderation*. Die Betrachtungsweise der makro-Ebene zeigt die Kooperation mehrerer Kooperationen, wie in einer *Inter-Föderation*.

Rahmenbedingungen. Die Rahmenbedingungen bestehen aus den Aspekten *Dauer der Föderation*, der *Art der Zusammenarbeit*, die *Kooperationsform* und den *Gründungsprozess*. Die Dauer der Föderation basiert auf der Art der Zusammenarbeit. Während bei der Zusammenarbeit auf Grund von FIM von einer unbefristeten Föderation auszugehen ist, ist die Dauer von Projekt-Föderationen auf die Projektlaufzeit beschränkt. Dies hat zudem Auswirkungen auf den Gründungsprozess. Eine Gründung einer unbefristeten Föderation wird stärker geplant als eine Projekt-Föderation, die auf Grund des Bedarfs entsteht. Spontan ereignisgesteuerte Föderationen sind in der Regel mit der Eigenschaft einer Ad hoc Föderation verknüpft. Die Koordination erfolgt insbesondere bei langfristigen Föderationen explizit, während eine implizite Koordination eher bei kurzlebigen Föderationen anzunehmen ist.

Dauer der Föderation. Die Dauer der Föderation ist unterschiedlich. Nationale Föderationen sind beispielsweise *unbefristet*, während Projekt-Föderationen nur für die *Dauer eines Projektes* anhalten. Ferner kann es eine *zeitliche Begrenzung*, unabhängig von einer Projektdauer, geben.

Zusammenarbeit. Diese Dimension wurde im Rahmen dieser Arbeit hinzugefügt. Die Zusammenarbeit zwischen den Organisationen ist unterschiedlicher Natur. In nationalen Föderationen bezieht sich die Zusammenarbeit auf *FIM*, während in *Projekten* und *Communities* das gemeinsame Erreichen eines bestimmten Zieles mit Hilfe von Identity Management im Vordergrund steht.

Koordination. Bei einer *expliziten* Koordination wird die Integration einer institutionellen Koordinationsinstanz befürwortet, während bei einer *impliziten* Koordination eine rein lokale Abstimmung vorliegt.

Gründungsprozess. Der Gründungsprozess ist meist *geplant*, er kann jedoch auch *spontan ereignisgesteuert* oder *bei Bedarf* geschehen. Je dynamischer der Gründungsprozess ist, desto höher sind in der Regel die Automatisierungsanforderungen.

Aspekte des Vertrauens. Es gibt mehrere Aspekte, die unter dem Merkmal Vertrauen zusammen gefasst sind. Der *Circle of Trust* hängt auch von der Kooperation der Föderation ab. Wenn beispielsweise die Gruppenstruktur abgeschlossen ist, hat der Circle of Trust (CoT) die Ausprägung statisch. Bei einer komplexen Anzahl der Teilnehmer wird der CoT virtuell ausfallen, da keine vollständigen Informationen über andere Mitglieder möglich sind. Bei einer *Bindungsintensität* mit der Ausprägung Absprache ist wiederum ein virtueller CoT üblicher als bei einer Kapitalbeteiligung. Direktes *Vertrauen* impliziert einen statischen oder dynamischen CoT.

Circle of Trust. Ein weiterer, neuer Aspekt ist der CoT, wie von Latifa Boursas [Bou09] beschrieben. Dieser kann *statisch*, *dynamisch* oder auch *virtuell* sein. Ein statischer CoT impliziert, dass Mitglieder über einen bestimmten Zeitraum nicht die Kooperation verlassen oder neu hinzukommen. Im Gegensatz dazu können in einem dynamischen CoT Organisationen dynamisch zur Kooperation hinzustoßen und so die Mitgliederzahl verändern. Virtuelle Aspekte werden vor allem dann bemerkbar, wenn Mitglieder indirekt in der Kollaboration involviert sind und somit keine vollständigen Informationen über andere Mitglieder besitzen.

Bindungsintensität. Die Bindungsintensität beschreibt den Grad, zu dem die Teilnehmer einer Kooperation bzw. Föderation ihre Autonomie aufgeben. Die Bindungsintensität kann ebenfalls zwischen zwei kooperierenden Organisationen betrachtet werden. Sie kann die Ausprägungen *Absprache*, *Vertrag* und *Kapitalbeteiligung* annehmen.

Vertrauen. Zuletzt wird das Vertrauen zwischen Organisationen betrachtet. Dieses kann bilateral, *direkt*, sein, wenn beide Organisationen Teil eines CoT sind oder wenn sie Geschäftsbeziehungen miteinander pflegen. *Indirektes*, transitives Vertrauen kann über eine dritte Person, beispielsweise ein Geschäftspartner, der beide Organisationen kennt, vorliegen.

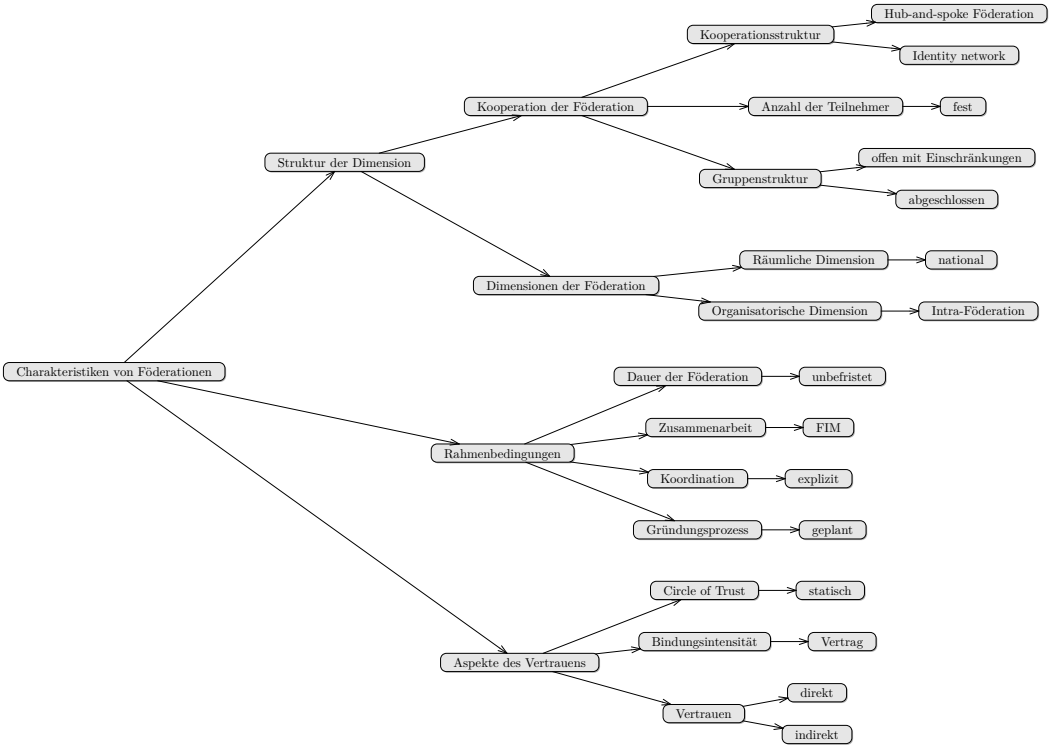

Abbildung 2.4.: Klassifikation von Föderationen in NRENs

Föderationen in *NRENs* haben folgenden Ausprägungen (vgl. Abbildung 2.4):

- Die Kooperationsstruktur entspricht meist einem Identity network, aber auch Hub-and-spoke Föderationen sind möglich.
- Die Anzahl der Teilnehmer ist meist fest durch die nationale Begrenzung auf R&E.
- Folglich ist die Gruppenstruktur zwischen offen mit Einschränkungen und abgeschlossen. Es können weitere Organisationen eintreten, wenn sie von der Föderationsverwaltung aufgenommen werden. Durch die Beschränkung auf das NREN und R&E ist der Teilnehmerkreis jedoch begrenzt.
- Die räumliche Dimension beschränkt sich auf das Land und ist somit national.
- Währenddessen ist die organisatorische Dimension eine Intra-Föderation.
- Die Föderation ist auf unbefristete Dauer ausgelegt.
- Die Zusammenarbeit ist, aus Föderationssicht, auf FIM ausgelegt, während einzelne Kooperationen Projekte bearbeiten.
- Die Koordination der Föderation erfolgt meist explizit durch die Föderationsverwaltung, in Deutschland beispielsweise durch die DFN-AAI.
- Die Gründung der Föderation ist folglich geplant, allerdings aus Gründen des dafür vorhandenen Bedarfs.
- Jede teilnehmende Organisation hat in der Regel einen Vertrag mit der Föderationsverwaltung. Die Kooperationen zwischen einzelnen Organisationen können trotzdem alle drei Ausprägungen, von Absprache bis hin zu SLA mit finanziellen Bedingungen, annehmen.
- Das Vertrauen in der Föderation ist, bedingt durch die geschlossene Form, eher statisch.
- Parallel dazu ist das Vertrauen zwischen einzelnen teilnehmenden Organisationen meist direkt, kann jedoch ebenfalls indirekt ausfallen.

Die dargestellte Klassifikation wird in dieser Arbeit eingesetzt, um die unterschiedlichen vorgestellten Föderationsarten genauer zu klassifizieren.

2.2.4. Dienstmodell und der Management-Aspekt

Im Folgenden werden die Begriffe des *Dienstes* und des *Managements* näher erläutert, um im Kapitel 5 erneut darauf zurück zu greifen und die zu entwickelnde Lösung daran beschreiben zu können.

Dienstmodell

Der Begriff des *Dienstes* ist für diese Arbeit aus unterschiedlichen Gründen elementar. Zum einen bieten Service Provider Dienste für Nutzer von IdPs an, zum anderen betreiben Föderationen und Inter-Föderationen Dienste. Beide Arten von Diensten müssen bei der Konzeption beachtet werden, insbesondere wenn unabhängige Dienste etabliert werden sollen. Dies zeigt bereits, dass es unterschiedliche Sichtweisen auf Dienste geben kann. Der Begriff des Dienstes kann unterschiedlich ausgelegt werden, wie u. a. von Gabi Dreo Rodosek [Rod02] aufgezeigt.

Aus den unterschiedlichen Auslegungen lassen sich zwei grundsätzliche Aspekte herausstellen. Zum einen gibt es den statischen Aspekt, der die charakteristischen Eigenschaften eines Dienstes beschreibt. Zum anderen existiert der dynamische Aspekt, der den Dienst als Funktion der Zeit oder anderen Kausalgefügen betrachtet.
Der dynamische Aspekt des Dienst-Begriffs wird insbesondere beim Dienstlebenszyklus deutlich. Zunächst muss ein Dienst in der *Planungsphase* geplant werden. Der Dienstanbieter unterbreitet beispielsweise einem Dienstinteressenten ein Angebot über die Diensterbringung und das darüber verfügbare Leistungsspektrum. Ein Interessent prüft in der Regel mehrere Angebote, daher benötigt es in dieser Phase unterschiedliche Analysen. Diese betreffen beispielsweise den Schwerpunkt des Bedarfs oder die verfügbaren Komponenten. Folglich werden die vorher skizzierten statischen Aspekte des Dienstes analysiert und festgelegt, die in dieser Arbeit anschließend näher beschrieben werden. In der *Verhandlungsphase* wird ein Vertrag zwischen dem Interessenten und dem Anbieter ausgehandelt. Dieser betrifft die vorher in der Planung besprochenen Aspekte und endet in der Regel mit dem Abschluss eines Vertrags. Die *Bereitstellungsphase* konzentriert sich auf die Realisierung der vorher vertraglich vereinbarten Dienstfunktionalität. Dazu gehören die Bereitstellung der Ressourcen, Installation, Konfiguration und das Testen des Dienstes. Der Kunde nimmt am Ende dieser Phase den Dienst ab. Anschließend wird der Dienst in der Betriebsphase in Betrieb genommen. Neben dem tatsächlichen Betrieb des Dienstes sind hier auch Managementaufgaben wichtig, wie z. B. der Betrieb eines Helpdesks oder einer Hotline. Laut Heinz-Gerd Hegering et al. [HAN99] wird der Betrieb aus Managementsicht in den Routinebetrieb, den Störungsbearbeitungsbetrieb und den Änderungsbetrieb aufgeteilt. Alle drei Betriebsmodelle betrachten unterschiedliche betriebliche Abläufe, die für den Betrieb eines Dienstes notwendig sind. Die *Anpassungsphase* beschäftigt sich mit Änderungen der Dienstfunktionalität, der Dienstimplementierung und des Dienstmanagements. Im Gegensatz zum Änderungsbetrieb werden in der Anpassungsphase grundsätzliche planerische Eingriffe am Dienst vorgenommen. Dies kann zum Beispiel die Einstellung eines Dienstes sein. Allgemein hängt diese Phase eng mit der Planungsphase und der Verhandlungsphase zusammen und hat Auswirkungen auf die Bereitstellung und den Betrieb. Der Dienstlebenszyklus endet mit der *Auflösungsphase*, in der die verwendeten Ressourcen wieder frei gegeben werden.

Der statische Dienstaspekt wird im Folgenden näher betrachtet, da er die Dienst-Eigenschaften allgemein und FIM-spezifisch darlegt. Das *Dienstmodell* des MNM Teams (vgl. [GHH+01] und [GHK+01]) befasst sich mit den managementspezifischen Aspekten beim Or-

ganisationsmodell in Hinblick auf Dienste und Dienstschnittstellen. Aus dem Modell sollen zum einen die Begriffe, die für ein dienstorientiertes Management verwendet werden, möglichst eindeutig aus dem Modell abzuleiten sein. Andererseits sollen konkrete Szenarien durch die Festlegung von Entitäten, Rollen, Interaktionen und Beziehungen modellierbar sein.

Basismodell

Das MNM-Dienstmodell bietet mehrere Sichten auf einen Dienst. Das *Basismodell* definiert die Rollen der Dienstleister (*provider*), Nutzer (*user*) und Kunden (*consumer*). Diese Rollen werden der Dienstleisterseite bzw. der Dienstnehmerseite zugeteilt. Eine Domäne wird verwendet, um die Zuständigkeitsbereiche festzulegen und zu beschreiben. Der Kunde geht über den Dienst eine vertragliche Beziehung mit dem Dienstleister ein. Dabei ist der Kunde über ein Customer Service Management (CSM) in den Managementinteraktionen involviert. Der Dienstnutzer greift über einen speziellen Dienstzugang auf die Nutzungsfunktionalität eines Dienstes zu. Der Provider nimmt den Gegenpart zum Kunden ein, der sich um die Dienstimplementierung und das Dienstmanagement kümmert. Der Dienst selbst ist keiner Seite zugehörig.

Im FIM-spezifischen *Basismodell* werden die Akteure des FIMs den Seiten Dienstnehmer, Dienstleister sowie seitenunabhängig zugeteilt. Der SP erbringt gegenüber dem Nutzer einen Dienst, jedoch benötigt der Service Provider vom Identity Provider die Nutzerdaten, was als eigenständiger Dienst betrachtet werden kann. In der Gesamtübersicht wird der IdP trotzdem der Dienstnehmerseite zugeordnet, da aus der Nutzersicht argumentiert wird und um die Unterschiede besser zu verdeutlichen. Die Seiten sehen wie folgt aus (vgl. Abbildung 2.5):

Dienstnehmerseite: Auf Seiten des Dienstnehmers befinden sich der *user* und der *identity provider*. Der Identity Provider ist nicht direkter *customer*, sondern der Nutzer steht im Vordergrund, während der IdP die Benutzerinformationen weiter gibt. Der Identity Provider gehört üblicherweise einer Föderation, *identity provider federation*, zu.

Seitenunabhängig: Ebenso, wie im MNM-Basismodell, befindet sich der *service* in der Mitte der Abbildung und ist seitenunabhängig. Der Service wird vom Service Provider betrieben und vom Nutzer verwendet. Zusätzlich kann sich eine *attribute authority*, die zusätzliche Benutzerinformationen liefert, auf dieser Ebene befinden. Die AA kann einer Föderation zugehören, beispielsweise der SP Föderation oder der IdP Föderation. Je nach Vorgehen ist die Attribute Authority entweder beim Nutzer, beim SP, beim IdP oder bei der Föderation angesiedelt.

Dienstleisterseite: Neben dem *service provider* ist gegebenenfalls eine *service provider federation* auf der Dienstleisterseite. Der SP kann eine Heimatföderation, die Service Provider Föderation, haben, jedoch ist es auch möglich, dass der SP mehreren oder keinen Föderationen angehört.

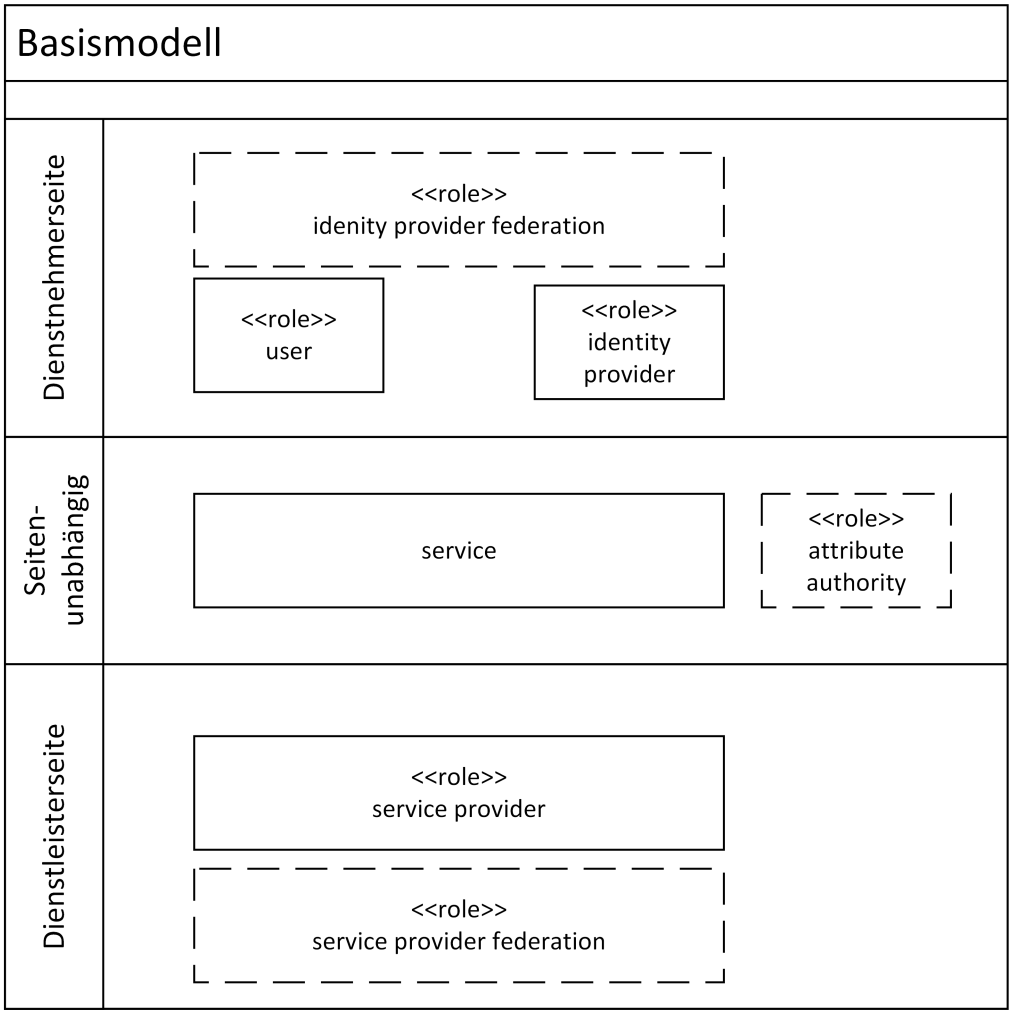

Abbildung 2.5.: Basismodell für Federated Identity Management

Dienstsicht

Der Dienst wird in der *Dienstsicht* näher spezifiziert, um ein kongruentes Verständnis zu erzielen. Dieser bietet eine Funktionalität, die einem Dienstnehmer über eine Schnittstelle zur Verfügung gestellt wird, vgl. [Rod02] und [GHH+01]. Die Dienstgüte wird dabei vorab festgelegt. Die angebotene Funktionalität besteht aus der Nutzungsfunktionalität an sich sowie der Managementfunktionalität. Wesentlich für die Funktionalität ist dabei die Menge der Interaktionen zwischen Dienstleisterseite und Dienstnehmerseite, die dem eigentlichen Zweck des Dienstes entsprechen. Alle übrigen Interaktionen werden der Managementfunktionalität zugeordnet. Dazu gehören u. a. eine Hotline und der Support des Dienstes. Die Interaktionen können auf Anwendungstransaktionen, Protokolltransaktionen und Workflows abgebildet werden. Die zusätzlichen Rollen des FIM-spezifischen Basismodells sind ebenfalls

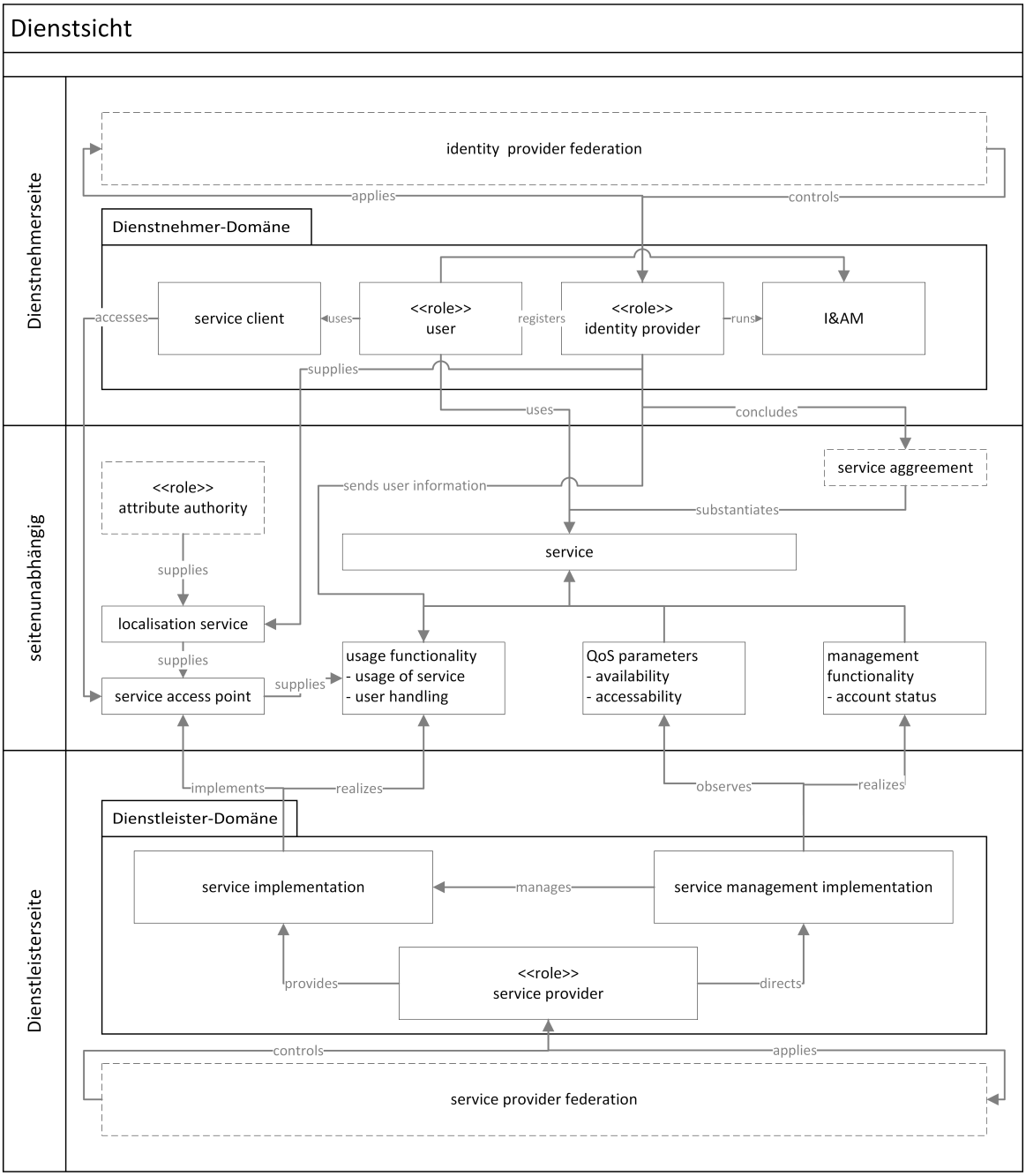

Abbildung 2.6.: Dienstsicht für Federated Identity Management

in der Dienstsicht (vgl. Abbildung 2.6) eingebunden:

Dienstnehmerseite: Der *identity provider* registriert den *user*, der einen *service client* verwendet, um auf einen Dienst zuzugreifen. Der IdP besitzt ein lokales *I&AM*, um seine

Nutzer zu verwalten. Diese vier Komponenten gehören der Dienstnehmer-Domäne an. Zusätzlich kann es eine *identity provider federation* geben, die ihre IdPs kontrolliert und bei der sich IdPs registrieren müssen. Die Informationen aus dem Identity & Access Management werden für den Service Provider benötigt, um die Authentifizierung und gegebenenfalls die Autorisierung beim Dienst sicher zu stellen.

Seitenunabhängig: In der Mitte und somit seitenunabhängig stehen der Dienst und alle Komponenten, die dazu gehören.

- Das Zentrum der unabhängigen Ebene bildet der *service*, der vom Nutzer genutzt wird. Das *service agreement* begründet die Dienstnutzung.
- Währenddessen bilden die *usage functionality*, *QoS parameters* und *management functionality* die Basis für den Dienst. Die Funktionalität des Dienstes kann beispielsweise eine zusätzliche Benutzerverwaltung oder die Nutzung eines Dienstes sein. Qualitätsmerkmale eines Dienstes werden hier beispielhaft als Verfügbarkeit und Erreichbarkeit angegeben. Als Managementfunktionalität kann z. B. der Status des Kontos gelten.
- Der Nutzer greift über den Service Client auf den *service access point* zu. Der service access point liefert den Zugriff auf die Funktionalität des Dienstes und somit auf den Dienst selbst.
- Durch den *localisation service* erhält der service access point die Information über den IdP des Nutzers, der im localisation service verfügbar sein muss. Zusätzlich kann eine *attribute authority* angebunden sein.

Dienstleisterseite: Auf der Dienstleisterseite existiert die Dienstleister-Domäne, bestehend aus *service provider*, *service implementation* und die *service management implementation*. Diese Komponenten sind für die Funktionalität bzw. das Management und die Qualitätssicherung des Services zuständig. Folglich stellt der SP eine service implementation bereit, die von der service management implementation verwaltet wird.

Implementierungssicht

Die Dienstimplementierung realisiert primär die Nutzungsfunktionalität des vereinbarten Dienstes. Dabei wird eine Nutzungsschnittstelle implementiert, um Zugriff auf die Funktionalität des Dienstes zu gewährleisten. Die Dienstimplementierung ist jedoch nicht nur rein technisch zu betrachten. Eine Kombination aus dem gesamten organisatorischen Wissen, dem Personal, der Hardware und der Software ist die für die Dienstrealisierung notwendig. Diese Dienstimplementierung wird durch die *Implementierungssicht* des MNM-Dienstmodells unterstützt. Die Dienstmanagementimplementierung umfasst alle erforderlichen Maßnahmen und Ressourcen, die benötigt werden um sicher zu stellen, dass der Dienst geplant, installiert und betrieben werden kann. Dies bedeutet für die FIM-spezifische Implementierung

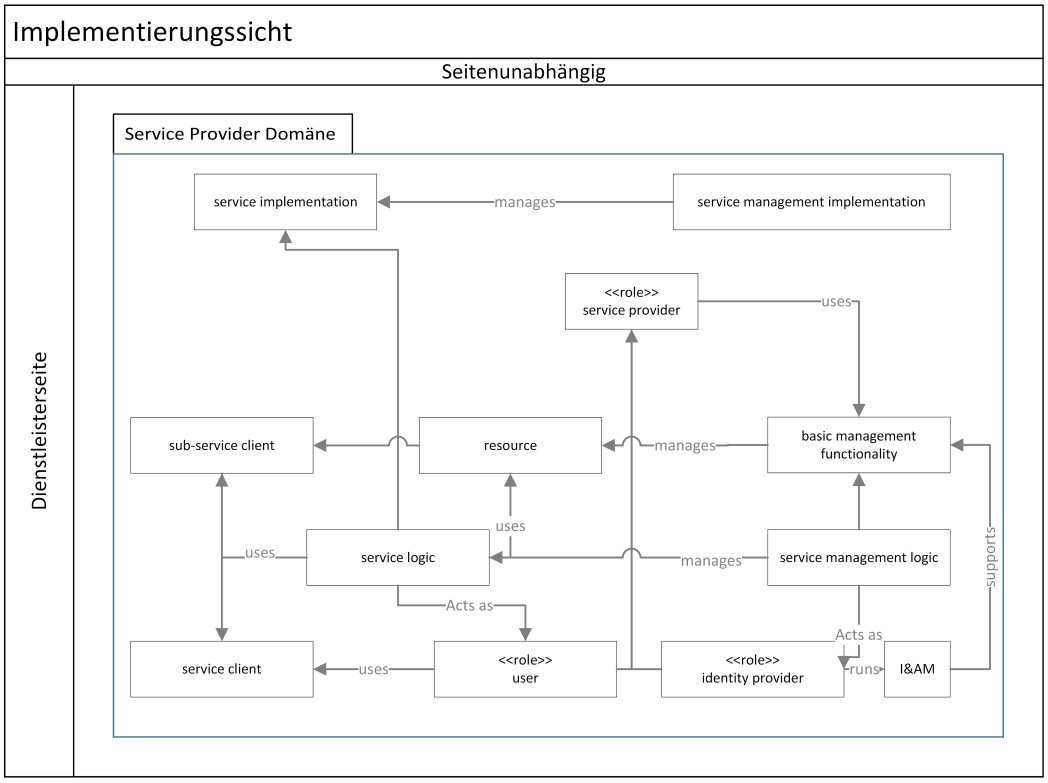

Abbildung 2.7.: Implementierungssicht für Federated Identity Management

folgende Zusammenhänge (vgl. Abbildung 2.7):

- Der *service provider* verwendet die *basic management functionality*.
- Die basic management functionality verwaltet die *resource*.
- Die resource bedient den *sub-service client*, falls dieser verfügbar ist.
- Die *service logic* verwendet den sub-service client sowie den *service client*, der vom *user* verwendet wird.
- Die service logic interagiert für den Nutzer.
- Der Nutzer sowie der *identity provider* haben eine Geschäftsbeziehung mit dem Service Provider.
- Die *service management logic* agiert als IdP. Sie verwaltet die service logic, aber auch die basic management functionality. Hier ist ebenfalls die Schnittstelle zum I&AM des IdPs angesetzt.

Das MNM-Dienstmodell ermöglicht, wie hier am Beispiel Federated Identity Management zu sehen, die Beschreibung von beliebigen Diensten. Das Modell wird erneut im Kapitel 5 angewandt, um die Zusammenhänge der Lösung passend aufzuzeigen.

Management von Diensten

Die Kooperation verteilter und heterogener Hardware und Softwarekomponenten wird immer wichtiger. Dies ist beispielsweise bei Web Services und im Grid-Umfeld zu beobachten. Zugleich steigen auch die Kooperationen, bei denen Federated Identity Management eine Rolle spielt. Daher ist ein geeignetes *Management* wichtig, um die Systeme und ihre Komponenten zu bewältigen. Laut [HAN99] müssen daher alle Komponenten einer Umgebung einem integrierten Managementansatz der Gesamt-IT unterworfen werden. Da sich verteilte Systeme, so wie in dieser Arbeit beschrieben, erheblich bezüglich ihrer Architektur, Größe, Komponenten, verwendete Software, Rahmenbedingungen und Ausrichtung unterscheiden, ist auch eine Managementlösung für alle Teilnehmer von Federated Identity Management nahezu utopisch. Daher wird in den Kapiteln 4 und 5 eine Lösung für eine möglichst große Anzahl an Teilnehmer konzipiert. Die Lösung soll dabei aus mehreren Bausteinen bestehen, die entsprechend orchestriert werden können. Dazu werden standardisierte Managementarchitekturen benötigt, die eine systemübergreifende Kombination von Managementmodulen ermöglichen. Ein Rahmenwerk für managementrelevante Standards wird *Managementarchitektur* genannt. Hierbei werden unterschiedliche Modelle betrachtet:

Informationsmodell: Beschreibung der relevanten Managementobjekte. Im Informationsmodell werden die syntaktischen und semantischen Möglichkeiten zur Modellierung und Beschreibung von Ressourcen und Informationen festgelegt. Diese Managementinformationen enthalten Informationen, die zu Zwecken des Managements ausgetauscht werden müssen. Die sogenannten Managementobjekte sind managementrelevante Abstraktionen realer Ressourcen.

Organisationsmodell: Festlegung der am Managementprozess beteiligten Systeme und deren Rollen, Kooperationsformen und jeweiligen Zuständigkeitsbereiche (Domänen). Policies leiten aus den übergeordneten Zielen bzw. Prozessen Vorgaben für das technische Management ab. Im Organisationsmodell gibt es sowohl aktive als auch passive Komponenten. Aktive Rollen steuern und werden daher Managementsysteme oder Manager genannt. Passive Komponenten werden dahingegen als Agenten oder Agentensysteme bezeichnet.

Kommunikationsmodell: Beschreibung der Zugriffsmechanismen auf Managementobjekte. Das Kommunikationsmodell definiert Prinzipien und Konzepte zum Austausch von Managementinformationen zwischen den im Organisationsmodell definierten Rollen. Das bedeutet, dass spezifiziert wird, welche Partner in welchem Kommunikationsmechanismus über welche Syntax und Semantik bezüglich Datenstrukturen Informationen austauschen.

Funktionsmodell: Definition generischer Managementfunktionen. Das Funktionsmodell gliedert die Gesamtaufgabe in verschiedene Funktionsbereiche und allgemeine Managementfunktionen. Die Funktionen werden in der Regel auf Managementplattformen und Agentensystem implementiert. Über geeignete Programmschnittstellen (API) können unterschiedliche Anwendungen darauf zugreifen. Dies kann beispielsweise für das Fehlermanagement, Konfigurationsmanagement, Abrechnungsmanagement, Leistungsmanagement und Sicherheitsmanagement relevant sein.

Die Management-Aspekte sind ebenfalls relevant für diese Arbeit. So müssen IdPs und SPs Informationen untereinander und mit den jeweiligen Föderationen austauschen. Entsprechend gibt es verschiedene Rollen, Kooperationsformen und Zuständigkeitsbereiche. Das Kommunikationsmodell beschreibt die Prinzipien und Konzepte zum Austausch von Managementinformationen. Wichtige Informationen im Federated Identity Management können beispielsweise Metadaten, Verlässlichkeitsklassen und Kennzahlen zum Vertrauen sein, die bei der Konzipierung der Lösung in Kapitel 4 sowie den zusätzlich benötigten Werkzeugen in Kapitel 5 beachtet werden müssen. Beim Design zusätzlicher Komponenten ist auch das Funktionsmodell entscheidend.

2.2.5. Technische Komponenten des Federated Identity Managements

Damit Benutzerinformationen vom Identity Provider zum Service Provider überhaupt übertragen werden können, müssen die Kommunikationsendpunkte bekannt sein. Diese Informationen werden über Metadaten [CMPM05] bekannt gemacht, die beispielsweise im FIM-Protokoll SAML als XML vorliegen und u. a. folgende Daten beinhalten:

EntityID. Jede Entität hat eine eindeutige Identität (*ID*) zur Identifizierung. Im Folgenden wird, auf Basis von SAML und der aktuellen Verwendung des Begriffs EntityID in FIM, eine Entität als ein Service Provider, Identity Provider oder Attribute Authority angesehen.

Zertifikate. Ein oder mehrere Zertifikate dienen zur Überprüfung der Identität des Kooperationspartners und schützen gegen Impersonation.

Ansprechpartner. Zusätzlich können Ansprechpartner genannt sein, die beispielsweise beim Security Incident Response benötigt werden.

Üblicherweise aggregiert die Föderationsverwaltung regelmäßig die Metadaten all ihrer Teilnehmer und signiert diese Datei. Die Entitäten können durch ihre Software die aggregierten Metadaten regelmäßig und automatisch nachladen. Durch die Signatur wird sichergestellt, dass die Föderationsmetadaten tatsächlich von der Föderation stammen.

Während bei Webdiensten in der Wirtschaft häufig OpenID Connect eingesetzt wird, basiert die Kommunikation der Entitäten im Hochschulumfeld auf SAML. SAML ist ein

XML-basiertes Framework zur Übermittlung von Authentifizierungsbestätigungen, Autorisierungsbestätigungen und Attributsinformationen. Das standardisierte Protokoll, entwickelt von dem Konsortium der Organization for the Advancement of Structured Information Standards (OASIS), erlaubt es Geschäftspartnern allgemeine Attributsauskünfte (*Attribute Assertion*) für Identitäts- und Profildaten eines speziellen Nutzers auszutauschen. Eine Aussage in SAML (*Statement*) besteht aus einer oder mehreren Zusicherungen [RHPM08]:

Authentication Statement: Zusicherung, dass ein eindeutig identifizierbarer Nutzer sich zu einer bestimmten Zeit authentifiziert hat.

Attribute Statement: Die vom SP angeforderten Attribute über einen Nutzer werden häufig im Anschluss an das Authentication Statement durch den IdP versendet.

Authorization Decision Statement: Der IdP definiert in dieser Antwort, ob ein bestimmter Nutzer autorisiert ist eine bestimmte Handlung durchzuführen oder auf eine festgelegte Ressource zuzugreifen.

SAML erlaubt es weitere Profile einzubinden, die zusätzliche Funktionalitäten bereitstellen, wie beispielsweise das *Web Browser SSO Profile* [HCH+05]. Das *Web Browser SSO Profile* unterstützt SSO in Webanwendungen, wodurch der Benutzer sich nur bei seinem Identity Provider anmelden muss, aber weitere Dienste bei verschiedenen SPs ohne ein erneutes Anmelden nutzen kann. Gleichzeitig wird SAML nicht direkt verwendet, sondern durch Implementierungen, wie das am häufigsten eingesetzte Shibboleth [Shi15]. Das Open-Source-Produkt Shibboleth stammt ursprünglich von Internet2 Middleware und wird inzwischen durch ein Konsortium weiter entwickelt. Shibboleth ermöglicht Single Sign On in Föderationen zwischen Entitäten mit gewachsenen I&AM-Infrastrukturen durch folgende Komponenten:

Service Provider Software. Die SP-Software läuft auf einem oder mehreren Webservern des SPs und unterstützt ihn u. a. durch die Bearbeitung von Assertions.

Identity Provider Software. Die IdP-Komponente liest beispielsweise nativ Informationen aus LDAP aus, authentifiziert Nutzer und lässt eine Einschränkung der Attribute, die an einen Service Provider gesendet werden, zu.

Metadata Aggregator. Das Tool liest, validiert, filtert und transformiert die Metadaten verschiedener Entitäten. Folglich wird der Metadata Aggregator auf Föderationsebene eingesetzt.

Discovery Service. Der Lokalisierungsdienst, der früher die Bezeichnung Where Are You From (WAYF) trug, erlaubt dem Nutzer seinen Identity Provider aus einer Liste auszuwählen.

SimpleSAMLphp [UNI14] von UNINETT besteht aus verschiedenen Modulen für Service Provider und Identity Provider. Zusätzlich kann der Lokalisierungsdienst DiscoJuice einge-

bunden werden, der ebenfalls im Abschnitt 3.2 näher erläutert wird.

2.2.6. Datenschutz im Federated Identity Management

Die Regeln zum Datenschutz aus I&AM gelten in FIM weiter, jedoch ist die Umsetzung von Teilaspekten vielschichtig durch die verteilte Architektur:

Einverständnis. Bevor Informationen über den Benutzer an einen Service Provider gesendet werden, muss das Einverständnis des Betroffenen eingeholt werden. Dies wird durch die Erweiterung uApprove [SWI14] gelöst, welche die übermittelten Informationen anzeigt und das Einverständnis des Nutzers, d. h. akzeptieren (`Accept`) oder ablehnen (`Reject`), einholt. In der japanischen Weiterentwicklung uApprove.jp [OYN+12] hat der Nutzer zusätzlich die Möglichkeit einzelne Attribute abzuwählen, bevor die verbliebenen Informationen dem Service Provider zur Verfügung gestellt werden.

Selbstauskunft. Der Anwender soll die Möglichkeit haben, stets eine Selbstauskunft zu bekommen. Diese beinhaltet das Einsehen der gespeicherten Daten und die Information über deren Verarbeitung durch die eingesetzten Systeme. Insbesondere der zweite Aspekt ist bei der verteilten Architektur im Federated Identity Management nicht trivial. IdPs stellen hierfür eine Übersicht bereit, an welchen Service Provider sie welche Informationen gesendet haben. Gleichzeitig wäre eine Übersicht von Seiten der SPs wünschenswert.

Wie bereits im vorherigen Abschnitt erwähnt, können die Attribute, die an einen Service Provider übermittelt werden, durch den Identity Provider eingeschränkt werden. Dies ist aus Datenschutz-Sicht wichtig, da der SP nur, analog zu I&AM, die benötigten personenbezogenen Daten erhalten soll. Der IdP muss diese Einschränkung in seiner Identity Provider Software konfigurieren. Alternativ kann ein Identity Provider ablehnen, einem Service Provider Informationen weiter zu geben, wenn beispielsweise der Grund für die Anfrage inkompatibel mit den Ansprüchen des IdPs ist. Folglich sind beim Austausch von personenbezogenen Daten zwischen Organisationen in Föderationen, aber auch in Inter-Föderationen, verschiedene Vertrauensbeziehungen zu beachten:

- Service Provider wollen Daten aus zuverlässigen Datenquellen mit einer hohen Datenqualität. Daher ist es für SPs wichtig die Menge der IdPs und AAs einschränken zu können.

- Identity Provider und Attribute Authorities wiederum geben nur Daten an zuverlässige SPs weiter. Dabei ist es essentiell, dass der SPs nachweisen kann, dass er derjenige ist, für den er sich ausgibt. Dies kann über eine Authentifizierung mittels signierten Metadaten geschehen.

- Benutzer haben ein Vertrauensverhältnis zu ihrem IdP, der ihre Daten speichert.

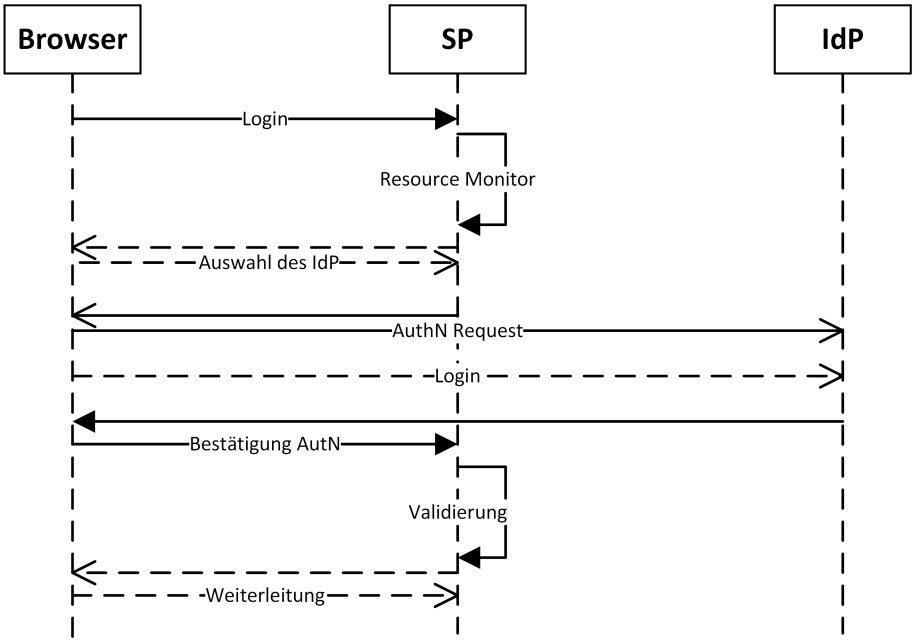

Abbildung 2.8.: Workflow im Federated Identity Management

Gleichzeitig haben Nutzer ein Interesse daran, dass ihre Daten nur an seriöse SPs übermittelt werden.

2.2.7. Workflows im Federated Identity Management

Nachfolgend wird ein beispielhafter Workflow im Federated Identity Management skizziert, der in Abbildung 2.8 veranschaulicht wird.

1. Schritt: Der Benutzer möchte einen Dienst nutzen.

2. Schritt: Der Resource Monitor überprüft, ob der Nutzer eine aktive Session hat und, wenn nicht, leitet ihn zum SP weiter, um den SSO-Prozess zu starten.

3. Schritt: Der Benutzer wählt seinen Identity Provider beim Lokalisierungsdienst aus.

4. Schritt: Der Service Provider bereitet einen *Authentication Request* vor und sendet diesen mitsamt dem Nutzer zum Identity Provider des Nutzers.

5. Schritt: Der Benutzer authentifiziert sich bei seinem IdP.

6. Schritt: Der Identity Provider sendet den Nutzer und einen *Authentication Response* zum

Service Provider.

7. Schritt: Der SP validiert den Response und leitet den Nutzer zum Dienst weiter, der diesen erfolgreich verwenden kann.

2.2.8. Anforderungen aus aktuellen Föderationen

In der Dissertation von Wolfgang Hommel [Hom07] (Kapitel 2) werden allgemeine Anforderungen an Architektur- und Werkzeugkonzepte für föderiertes Identitäts-Management aufgestellt, die ebenfalls für diese Arbeit gelten.

Funktionale Anforderungen

Die folgenden funktionalen Anforderungen sind allgemein gültig:

- Die Behandlung von Fehlern muss unterstützt werden [FA-Fehlermanagement].
- Der Nutzer muss zusätzlich die Möglichkeit haben, eine Auswahl an zu versendenden Attributen zu treffen [FA-Interaktion].
- Es müssen geeignete Schnittstellen zu den internen Systemen bestehen [FA-Konnektor].
- Organisationen müssen parallel in mehreren Rollen, wie SP und IdP, agieren können [FA-Rollen].
- Die Implementierung und die Teilnahme an diesen Föderationen müssen in akzeptabler Zeit möglich sein [FA-Realisierbarkeit].
- Der Service Provider muss den Abruf von Daten vom Identity Provider initiieren können. Im Gegensatz zu Wolfgang Hommels Dissertation (Kapitel 2) wird hier nur von einem Pull-Verfahren ausgegangen. Push ist optional [FA-Pull&Push].
- Aktualisierte Daten müssen an die entsprechenden Entitäten weiter geleitet werden können, damit diese Entitäten nicht mit veralteten Daten, beispielsweise Freigaben, arbeiten [FA-Aktualisierung].

Nichtfunktionale technische Anforderungen

Basierend auf aktuellen Föderationen ergeben sich zusätzliche nichtfunktionale technische Anforderungen:

- Die Benutzeroberflächen müssen, insbesondere bei Fehler, intuitiv zu bedienen und selbsterklärend sein [NFA-Usability].
- Das System muss unabhängig von der Hardware und dem Betriebssystem verwendet werden können [NFA-Portabilität].
- Das System muss performant sein, beispielsweise auf Anfragen in akzeptabler Zeit reagieren [NFA-Performanz].

Sicherheitsanforderungen

In Hinblick auf existierende Strukturen in Föderationen gilt Folgendes:

- Die Verwaltung und Aktualisierung sicherheitsrelevanter Konfigurationsparameter, wie Metadaten, muss weitgehend automatisiert werden können, so dass diese Metadaten nur noch an einer zentralen Stelle gepflegt werden müssen [SEC-Metadaten].
- Das System und seine Teilkomponenten müssen auditierbar sein, um jede Änderung nachvollziehen zu können [SEC-Auditing].
- Das System muss sich nahtlos in bestehende Netzwerk- und Systemsicherheitsprozesse integrieren lassen [SEC-Systemsicherheit].
- Die Übertragung der Benutzerinformationen muss sicher sein. Dies beinhaltet, dass die Gegenseite identifiziert und authentifiziert werden muss. Zusätzlich muss die Integrität, beispielsweise durch kryptographische Prüfsummen, und die Verschlüsselung der Daten gewährleistet werden [SEC-Datenübertragung].
- Benutzer müssen sich vor der Nutzung eines Dienstes authentifizieren können [SEC-Authentifizierung].
- Die Abfrage von Benutzerinformationen muss durch eine so genannte Attribute Release Policy (ARP) auf ausgewählte Attribute eingeschränkt werden können, um beispielsweise keine rein internen Informationen an externe SPs zu liefern [SEC-ARPs].

Organisatorische Anforderungen

Die folgenden Anforderungen reflektieren die Sicherheitsinfrastruktur der nationalen Föderation:

- Die Lösung muss Schnittstellen zu den organisationsinternen Supportprozessen, wie dem Service Desk und dem Change Management, aufweisen [ORG-Supportprozesse].

- Registrierte Organisationen müssen validiert und überprüft werden können. Dies kann beispielsweise über eine Instanz der Föderation geschehen [ORG-Validierung].
- Für die Migration müssen geeignete Lösungen bereitgestellt werden [ORG-Migration].
- Die Anforderung [FA-Realisierbarkeit] ist auch aus organisatorischer Sicht zu betrachten.

Datenschutzrechtliche Anforderungen

Die Anforderungen [FA-Interaktion] und [SEC-ARPs] gelten ebenfalls beim Datenschutz.

Zusätzlich kann die Anforderung [DSA-Zustimmung] übernommen werden. Es ist erforderlich, dass Benutzer über die Weitergabe ihrer Daten an Dritte informiert werden und dieser zustimmen.

2.3. Inter-Federated Identity Management

Definition 4. Inter-Föderation

Inter-Föderation ist der Zusammenschluss mehrerer Föderationen.

Im Gegensatz zu Federated Identity Management verläuft der Informationsaustausch im Inter-FIM über Föderationsgrenzen hinweg. Dies wurde notwendig, damit Nutzer Dienste anderer Föderationen mit den Vorteilen von Federated Identity Management, d. h.

- Effizienzsteigerung,
- höhere Datenqualität durch eine zentrale Datenhaltung beim Identity Provider, sowie
- Benutzerfreundlichkeit, z. B. durch SSO,

nutzen können. So forschen Wissenschaftler in Organisationen und Projekten, die auf mehrere Föderationen verteilt sind, aber auch Firmen arbeiten verstärkt bereichsübergreifend. Theoretisch wäre es möglich, dass Identity Provider und Service Provider in mehreren Föderationen gleichzeitig teilnehmen. Dies wäre jedoch mit zusätzlichem Aufwand verbunden, u. a. weil jede Föderation ihr eigenes Schema besitzt und jede Entität dafür mindestens Konvertierungsregeln erstellen, wenn nicht sogar zusätzliche Attribute anlegen muss. Alternativ gäbe es auch die Möglichkeit Benutzerkonten zu dupliziert, was jedoch zu redundanter Datenhaltung, somit auch Verschlechterung der Datenqualität und Mehraufwand auf Seiten

der Administratoren, sowie zur Aufhebung von SSO führen würde. Aufgrund der Schwierigkeiten eine weltweite Föderation zu etablieren, u. a. durch die verschiedenen Schemata, Anforderungen und rechtlichen Grundlagen, wurden Inter-FIM-Technologien entwickelt. Da aktuelle Deployments von Inter-FIM auf FIM-Technologien aufbauen, wird nachfolgend vor allem auf die Unterschiede eingegangen.

2.3.1. Architekturen und Inter-FIM-Modelle

Inter-FIM ist aktuell mit einer Umbrella-Föderation gleichzusetzen, d. h. eine Föderation über Föderationen, die FIM einsetzen. Analog zu Federated Identity Management müssen in Inter-Federation Identity Management die Metadaten aller Teilnehmer, eine Teilmenge aller potentiellen Teilnehmer, aggregiert werden, um die Endpunkte der Kommunikation zu kennen. Dies geschieht in der Inter-Föderation eduGAIN über den MDS, der die Metadaten der teilnehmenden Entitäten der Föderationen akkumuliert und aggregiert, was zu einer wachsenden Metadaten-Datei führt. Beispielsweise umfasst der Metadatensatz von eduGAIN Stand Januar 2016 etwa 237.000 Zeilen XML-Code für 2523 teilnehmenden Entitäten [Ter16].

Ferner sind für den Informationsaustausch eine Standardisierung der Semantik und Syntax der auszutauschenden Informationen notwendig. Dies wird im Bereich des FIM und Inter-FIM über ein so genanntes Attributs-Schema gelöst. So beschreiben die Schemata

- `eduPerson` von Internet2 die akademischen Informationen über Wissenschaftler, Mitarbeiter und Studenten,
- `dfnEduPerson` die deutsche Erweiterung des internationalen Standards `eduPerson` um Informationen für eLearning-Angebote, wie beispielsweise Studiengang, und
- `SCHAC` weitere akademische Informationen, die für FIM und Inter-FIM benötigt werden.

Während Föderationen eigene Erweiterungen festlegen, damit alle benötigten Informationen ausgetauscht werden können, beschränken sich Inter-Föderationen auf Grund ihrer Struktur auf wenige Attribute, die von jedem IdP unterstützt werden können. Dadurch ist es möglich, dass nicht jeder Service Provider alle Attribute erhält, die er für seinen Dienst benötigt, wie aus den Zwiebelschichten der Schemata in Abbildung 2.9 hervor geht. Es gibt wenige allgemeine Attribute im Zentrum. In der nächsten Schicht befinden sich globale Schemata, wie `eduPerson` und `SCHAC`, die weitere Attribute festlegen, die fast überall in Research & Education benötigt werden. Die nächste Schicht bilden nationale Schemata, wie `dfnEduPerson`, die nationale Attribute bestimmen. Diese Attribute unterscheiden sich von Schema zu Schema. Zusätzlich kann es regionale oder lokale Schemata geben, die nur von wenigen Diensten genutzt werden.

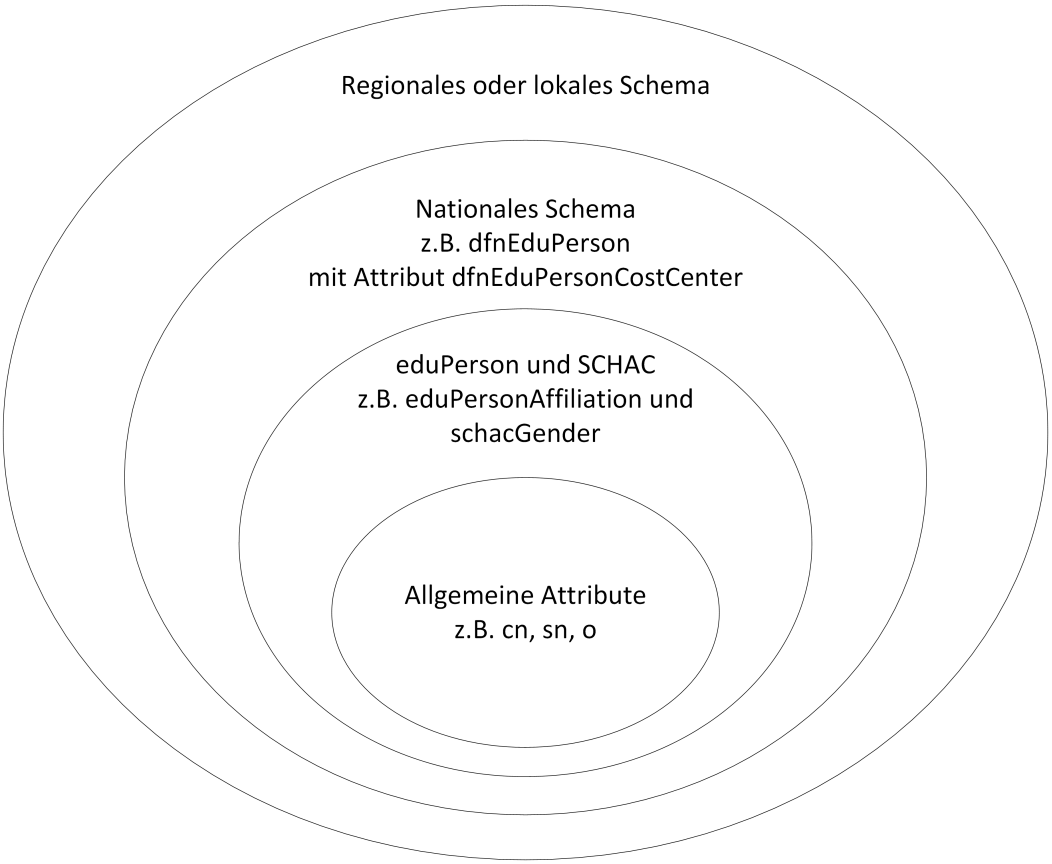

Abbildung 2.9.: Schemata in Zwiebelschichten dargestellt, basierend auf [Lin09]

2.3.2. Trust-Modelle

Damit die Nutzerinformationen ausgetauscht werden können, ist zusätzlich Vertrauen (*Trust*) notwendig. Grundsätzlich vertrauen die teilnehmenden Föderationen untereinander. Jedoch hängt die Ausprägung stark von den Policies und Richtlinien der einzelnen Föderationen ab. Beispielsweise vertrauen alle Teilnehmer der Inter-Föderationen KALMAR2, d. h. die NRENs der Länder Finnland, Norwegen, Schweden, Estland, Island und Dänemark, sich gegenseitig. Das bedeutet, dass die Verträge nicht auf Teilnehmerebene, sondern auf Föderationsebene geschlossen wurden. Ein schlechter skalierendes Trust-Modell herrscht in den meisten Ländern vor. So müssen die eduGAIN-Teilnehmer des NRENs DFN-AAI bilaterale Verträge mit ihren Kooperationspartnern abschließen, bevor Nutzerinformationen ausgetauscht werden können. Gleichzeitig sind bei der Umbrella-Föderation keine Verlässlichkeitsklassen spezifiziert, da jede Föderation unterschiedliche Anforderungen hat und dadurch eigene Klassen pflegt. Dies erschwert wiederum auf Grund der fehlenden Vergleichsmöglichkeit den Abschluss von bilateralen Verträgen insbesondere bei kritischen Diensten.

2.3.3. Klassifikation von Inter-Föderationen

Basierend auf der im vorherigen Abschnitt dargestellten Klassifikation von Föderationen lassen sich Inter-Föderationen wie folgt beschreiben (vgl. Abbildung 2.10):

- Die Kooperationsstruktur kann sowohl Hub-and-spoke Föderation als auch Identity networks sein. Ad hoc Föderationen als Inter-Föderationen sind jedoch schwierig.
- Die Anzahl der Teilnehmer ist auf Grund der Föderationsstrukturen fest oder komplex.
- Die Gruppenstruktur kann alle drei Ausprägungen, d. h. allgemein offen, offen mit Einschränkungen und fest, annehmen.
- Die räumliche Dimension lässt sich nicht weiter einschränken. Tendenziell sind Inter-Föderationen eher national oder international ausgerichtet, jedoch sind auch regionale und sogar lokale Dimensionen denkbar.
- Die organisatorische Dimension entspricht einer Inter-Föderation.
- Die Dauer der Inter-Föderation ist nicht näher festgelegt. Sie kann alle drei Ausprägungen annehmen.
- Ebenso lässt sich der Hauptgrund für die Zusammenarbeit nicht näher spezifizieren.
- Die Koordination ebenfalls alle drei Formen annehmen.
- Der Gründungsprozess ist tendenziell geplant oder bei Bedarf. Theoretisch ist auch spontan ereignisgesteuert möglich.
- Der Circle of Trust kann alle drei Ausprägungen annehmen.
- Auch bei der Bindungsintensität sind alle drei Formen denkbar.
- Das Vertrauen zwischen zwei Organisationen kann sowohl direkt, beispielsweise bei Kooperationen, als auch indirekt sein.

Inter-Föderationen stellen somit eine Metaorganisation dar, um Federated Identity Management über Föderationsgrenzen hinaus zu gewährleisten. Die Inter-Föderation eduGAIN (vgl. Abbildung 2.11) hat beispielsweise folgenden spezifischen Charakteristiken:

- Die Kooperationsstruktur besteht aus Hub-and-spoke Föderationen sowie Identity networks, die bei eduGAIN über Verträge ihrer Föderation mit GÉANT zu einer Inter-Föderation verbunden sind. Die Struktur der nordischen Inter-Föderation KALMAR2 ist äquivalent. Technisch vertrauen sich die Entitäten untereinander, d. h. die Ausprägung ist am besten mit einem Identity network vergleichbar.

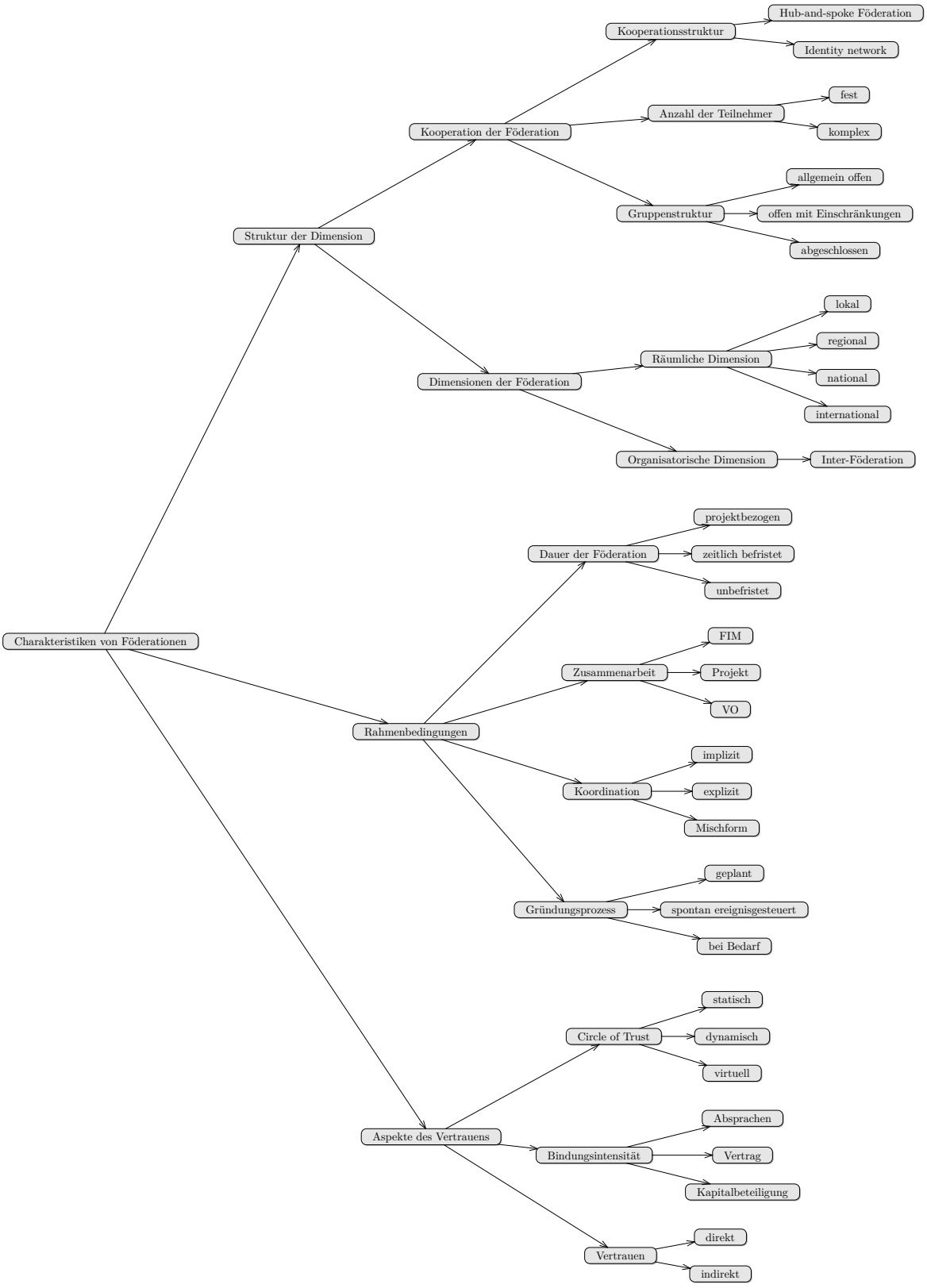

Abbildung 2.10.: Klassifikation von Inter-Föderationen

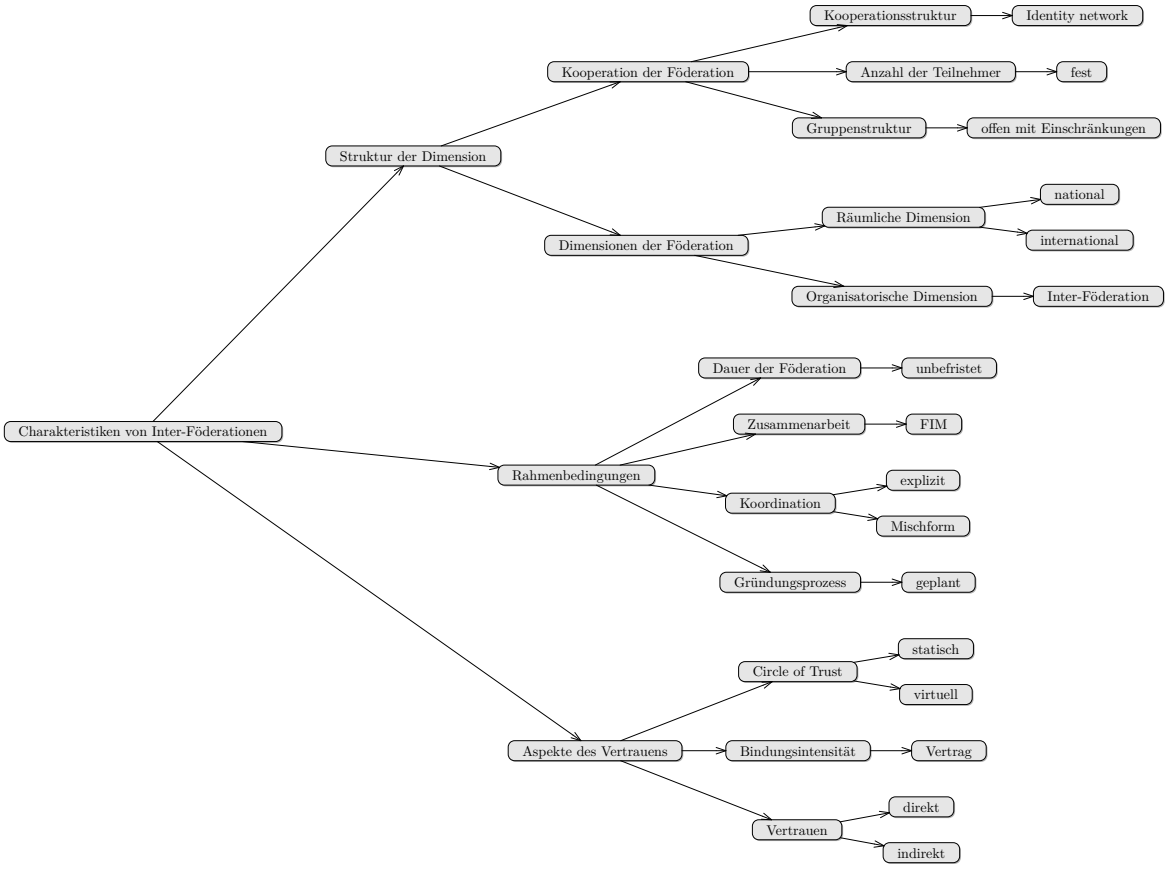

Abbildung 2.11.: Klassifikation von Inter-Föderationen am Beispiel von eduGAIN

- Die Anzahl der Teilnehmer ist auf Grund der Föderationsstrukturen fest, jedoch können ganze Föderationen beitreten.
- Die Gruppenstruktur von eduGAIN, aber auch KALMAR2, entspricht auf Föderations- und Inter-Föderationsebene der in den einzelnen Föderationen und ist somit offen mit Beschränkungen.
- Die Inter-Föderation ist bei eduGAIN und als auch bei KALMAR2 international ausgerichtet.
- Die organisatorische Dimension entspricht einer Inter-Föderation.
- Die Inter-Föderation eduGAIN ist langfristig ausgelegt ohne einem festgelegten Enddatum.
- Bei der Zusammenarbeit steht Federated Identity Management im Vordergrund.
- Die Koordination kann sowohl explizit als sein auch eine Mischform aufweisen. Die Bindung zur eigenen Föderation ist größer als zur Inter-Föderation, wodurch die Koordination vermehrt über die Föderationen abläuft.
- Der Gründungsprozess ist im Fall von eduGAIN geplant.
- Der Circle of Trust ist tendenziell statisch, wobei auf Grund der Größe der Metaorganisation keine Organisation vollständige Informationen über die anderen Teilnehmer haben kann. Nachdem auch ganze Föderationen beitreten können, kann ebenfalls die Ausprägung dynamisch angenommen werden.
- Die Bindungsintensität geschieht über einen Vertrag, während die Bindungsintensität zur lokalen Föderation größer ist als zur Inter-Föderation.
- Das Vertrauen zwischen zwei Organisationen kann sowohl direkt, beispielsweise bei Kooperationen, als auch indirekt sein. Das technische Vertrauen zwischen teilnehmenden Organisationen ist durch den zentralen Metadatenaustausch bereits hergestellt.

2.3.4. Datenschutz

Auf internationaler Ebene sind die rechtlichen Bestimmungen des Datenschutzes unterschiedlich ausgeprägt. Dies wirkt sich auf den Abschluss bilaterale Verträge in Inter-Föderationen aus. Als Grundlage hierfür dienen u.a.:

- Europäische Union (EU) Richtlinie 95/46/EG [ER95], die es Service Providern verbietet Benutzerinformationen abzufragen, die sie nicht für die Nutzbarkeit eines Dienstes benötigen.

- Code of Conduct (CoCo) [Lin13], der die EU-Gesetzgebung in einen Vertrag kleidet, der von Service Providern in der Inter-Föderation eduGAIN unterschrieben werden muss. Trotzdem sind die Ausprägungen pro Land unterschiedlich. So erlaubt beispielsweise Schweden die Nutzung von Microsoft Office 365 und des Clouddienstes Box im Hochschulumfeld, während dies in Deutschland umstritten ist.

- Außerhalb der EU kann der Code of Conduct dann zum Einsatz kommen, wenn die Datenschutzrichtlinien denen der EU entsprechen. Für alle übrigen Länder gibt es aktuell keinen allgemein gültigen Vertrag.

Für Kommunikationen innerhalb von Deutschland ist das Bundesdatenschutzgesetz zuständig, welches für alle öffentlichen Stellen des Bundes und für die Privatwirtschaft gilt, sofern personenbezogene Daten erhoben, verarbeitet oder genutzt werden. Laut [BIT08] ist ein vergleichbares Schutzniveau in der EU durch die Richtlinie 95/46/EG gesichert. Allgemein gilt der Grundsatz eines Verbots mit Erlaubnisvorbehalt, zum Beispiel wenn der Betroffene selbst einwilligt oder wenn er zuvor über den konkreten Verwendungszweck informiert und auf die vorgesehene Datenübermittlung hingewiesen wurde. Explizit muss einer der folgenden Gründe eintreffen [ER95]:

- Einwilligung der betroffenen Person ohne jegliche Zweifel.

- Erforderlichkeit der Verarbeitung zur Erfüllung eines Vertrages, bei der die betroffene Person entweder ein Vertragspartner selbst ist oder auf Antrag der betroffenen Person erfolgt.

- Erforderlichkeit der Verarbeitung zur Erfüllung einer rechtlichen Verpflichtung von Verantwortlichen der betroffenen Person.

- Erforderlichkeit der Verarbeitung zur Wahrung lebenswichtiger Interessen der betroffenen Person.

- Erforderlichkeit der Verarbeitung zur Wahrnehmung einer Aufgabe, die im öffentlichen Interesse liegt oder in Ausübung öffentlicher Gewalt erfolgt.

- Erforderlichkeit der Verarbeitung zur Verwirklichung des berechtigten Interesses, welches von dem für die Verarbeitung Verantwortlichen oder von dem Dritten wahrgenommen wird, dem die Daten übermittelt werden. Dies gilt, sofern nicht das Interesse oder die Grundrechte und Grundfreiheiten der betroffenen Person überwiegen.

Der Consent (Einwilligung) ist abstreitbar, wenn die Person die Daten übersenden muss, um einen Dienst zu nutzen, der für die Arbeit erforderlich ist. Für die Länder Norwegen, Island, Schweiz, Kanada, Argentinien, Isle of Man, Guernsey und Liechtenstein ist das Datenschutzniveau auf äquivalentem Niveau. Für Länder außerhalb der EU gilt ansonsten, dass die Datenübertragung insbesondere dann unterbleiben soll, wenn kein angemessenes Datenschutzniveau gegeben ist. Eine Ausnahme hiervon ist möglich, wenn der Nutzer der

Übertragung selbst einwilligt oder wenn ein entsprechender Vertrag mit verbindlichen Regelungen besteht. Wenn sich eine der beteiligten Organisationen in den USA befindet und diese die Prinzipien des Safe Harbour Pakets befolgt, welches u. a. sieben Datenschutzprinzipien aufstellt, galt ein angemessenes Datenschutzniveau erreicht[1].

2.3.5. Workflow

Grundsätzlich stimmt der Workflow bei Inter-Federation Identity Management mit dem aus FIM überein, jedoch nur, wenn bereits ein bilateraler Vertrag existiert oder wenn im Lokalisierungsdienst alle möglichen IdPs angezeigt werden und nicht nur diejenigen, mit denen der Service Provider bereits Verträge abgeschlossen hat. Zusätzlich muss der Identity Provider automatisch alle vom SP geforderten Attribute im gewünschten Format übermitteln können und dürfen. Ansonsten validiert der Service Provider den unvollständigen Response und leitet den Nutzer zum Dienst weiter, der diesen ggf. eingeschränkt verwenden kann. Im Folgenden werden mögliche Gründe genannt, wieso eine Dienstnutzung mit Verzögerung, eingeschränkt oder gar nicht möglich ist:

- Grund: Der IdP des Nutzers ist im Lokalisierungsdienst nicht auswählbar.
  Folge: In diesem Fall kann der Nutzer den Dienst nicht sofort nutzen, sondern muss zunächst seinen IdP über die gewünschte Nutzung informieren.

- Grund: Der SP benötigt Attribute, die der IdP nicht kennt.
  Folge: Daher kann der IdP nicht alle geforderten Attribute liefern. Der SP entscheidet anschließend, ob der Dienst überhaupt verwendet werden kann. Der Nutzer kann den Dienst gar nicht oder nur eingeschränkt nutzen. In diesem Fall muss der Nutzer ebenfalls seinen Identity Provider informieren.

- Grund: Der Service Provider benötigt Attribute, für die der Identity Provider keine Konvertierungsregel vorrätig hat.
  Folge: Folglich kann auch hier der IdP nicht alle geforderten Attribute versenden. Der Nutzer muss sich bei seinem IdP melden.

- Grund: Der Identity Provider hat die Möglichkeit einen Satz an Attributen an alle SPs zu liefern, die den Code of Conduct akzeptiert haben. Wenn der SP den CoCo nicht akzeptiert hat, ist diese Option nicht anwendbar.
  Folge: Der Nutzer muss sich an seinen IdP wenden. Entweder konfiguriert dieser für den speziellen Service Provider eine eigene Regel oder erlaubt die Nutzung nicht.

- Grund: Der SP gehört keiner Kategorie von Entitäten an bzw. einer, an den der IdP automatisch keine Attribute versendet.

[1] Court of Justice of the European Union: Press Release No 106/15, Luxembourg, 23 September 2015 `http://curia.europa.eu/jcms/upload/docs/application/pdf/2015-09/cp150106en.pdf` [Online, abgerufen am 06.01.2016].

Folge: Der Service Provider erhält nicht die gewünschten Attribute. Die weiteren Folgen sind äquivalent zum zweiten Aspekt.

- Grund: Der SP hat in seinen Metadaten nicht angegeben, welche Attribute er benötigt.
  Folge: Daher kann der Identity Provider nicht automatisch wissen, welche Attribute vom Service Provider gebraucht werden. Das weitere Vorgehen entspricht dem zweiten Aspekt.

- Grund: Der IdP versendet keine Attribute an SPs, denen er nicht vertraut.
  Folge: Auch hier muss der Nutzer seinen Identity Provider über die gewünschte Nutzung des Dienstes informieren.

Eine vertiefende Betrachtung des Inter-Federation Identity Management und der daraus resultierenden Anforderungen erfolgt im nächsten Abschnitt, dem Hauptszenario.

2.3.6. Inter-FIM-Szenario: LRZ in der Inter-Föderation eduGAIN

Das Leibniz-Rechenzentrum ist eine der Entitäten in der Föderation DFN-AAI, die ebenfalls an der Inter-Föderation *eduGAIN* teilnimmt. Gleichzeitig betreibt das LRZ die Identity Provider für die TUM und die Ludwig-Maximilians-Universität (LMU), die ebenfalls an der Inter-Föderation teilnehmen. Das LRZ unterscheidet sich von den meisten Hochschulrechenzentren dadurch, dass es für alle Hochschulen im Umkreis zuständig, jedoch gleichzeitig organisatorisch wie auch juristisch unabhängig von ihnen ist.

Ausgangssituation

Die Kennungen und Berechtigungen der 150.000 Studenten, Mitarbeiter und Professoren werden zentral durch das LRZ in einem Identity & Access Management-System verwaltet. Mehrere miteinander über ein Meta-Directory synchronisierte Identity Repositories werden eingesetzt, um die Daten in verschiedenen Formaten für unterschiedliche Dienste bereit zu stellen und diese abzugleichen. Auch der Shibboleth-IdP des LRZs verwendet das LDAP-basierte Meta-Directory als Quelle.

Neben direkten physikalischen Schnittstellen existiert eine Einbindung in Geschäftsprozesse:

- Bei der Einstellung und beim Ausscheiden von Mitarbeitern,

- Anbindung an ein Trouble Ticket System,

- Ausdehnung des Security Managements auf das FIM-System, während gleichzeitig Identifikation, Authentifizierung und Autorisierung von Benutzern zur klassischen IT-

Managementfunktion des Sicherheitsmanagements gehört,

- Berücksichtigung des FIM-Systems beim Change Management,
- Abhängigkeiten zu FIM-Diensten beim Incident und Problem Management und
- Anpassungen beim Configuration und Release Management.

Die TUM, die LMU, wie auch das LRZ haben je einen Vertrag mit der DFN-AAI geschlossen, um an der nationalen Föderation teilzunehmen. Die DFN-AAI agiert dabei als zentraler Vertragspartner, der u. a. Qualitätsansprüche regelt. Anbieter haben ein gewisses Schutzbedürfnis für ihre Ressourcen, gleichzeitig hat jede Heimatorganisation unterschiedliche Verfahren zur Identifizierung, Authentifizierung von Nutzern und zur Pflege der Daten. Alle drei Verlässlichkeitsklassen der DFN-AAI haben Anforderungen zu den folgenden drei Kriterien:

I: Verfahren zur Identifizierung des Nutzers.

A: Verfahren zur Authentifizierung des Nutzers beim Zugriff auf einen Dienst.

D: Verfahren zur Datenhaltung und Prozesse zur Pflege von digitalen Identitäten.

Entsprechend werden die Identity Provider in der Föderation der DFN-AAI in drei Klassen eingeteilt, die festgelegte Mindestanforderungen enthalten [DFN15a]:

Test: Klasse, die nur für Testzwecke verwendet werden soll. Es gibt keine Mindestanforderungen.

Basic: Klasse, die minimale Anforderungen stellt und somit für die Masse an IdPs möglich ist:

- Die Identifizierung des Benutzers muss anhand einer eindeutigen Adresse geschehen.
- Ausweis gegenüber dem lokalen I&AM muss anhand einer eindeutig zuzuordnenden digitalen Adresse geschehen. Dadurch wird jedoch nicht sichergestellt, dass sich hinter der digitalen Identität die vermutete Identität verbirgt.
- Die Benutzerinformationen müssen innerhalb von 3 Monaten korrigiert bzw. aktualisiert werden.

Advanced: Klasse mit strengeren Anforderungen, die die IdPs erfüllen müssen, um teilzunehmen:

- Die Identifizierung des Benutzers muss anhand von amtlichen Dokumenten gegenüber einer Vertrauensinstanz geschehen, beispielsweise bei der Immatrikulation

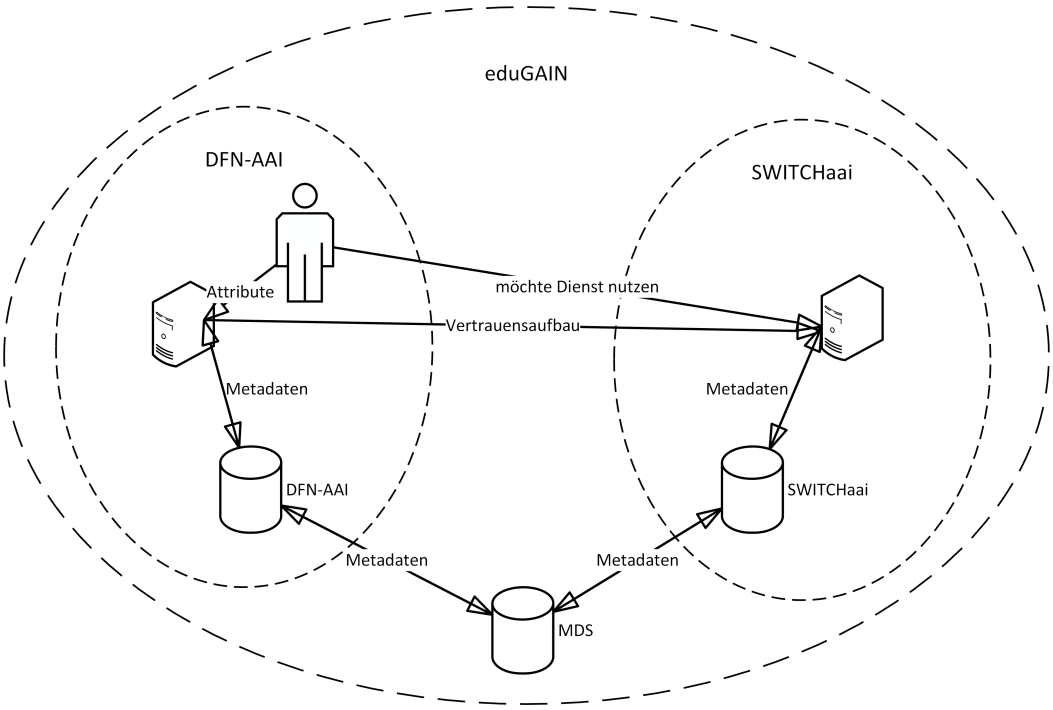

Abbildung 2.12.: Architektur der Inter-Föderation eduGAIN

mit Personalausweis.

- Zum Ausweis der Identität muss ein personalisiertes Benutzerkonto mit Kennung und Passwort oder Zertifikat verwendet werden.

- Die Benutzerinformationen müssen innerhalb von 2 Wochen korrigiert bzw. aktualisiert werden.

Für jede Verlässlichkeitsklasse gibt es entsprechende aggregierte Metadaten, die anzeigen, welche Entitäten diese Mindestanforderungen erfüllen. Das LRZ kann der Verlässlichkeitsklasse *Advanced* zugeordnet werden. Jeder Service Provider in der nationalen Föderation erhält über einen *MetadataProvider* die Metadaten des LRZs, da diese in den Metadaten der Klasse *Advanced*, aber auch in der *Advanced+Basic* enthalten sind. Das LRZ als Identity Provider muss im Gegenzug die Metadaten von *Advanced* sowie *Basic* laden. Die aggregierten und signierten Metadaten der Teilnehmer werden automatisch von der DFN-AAI geladen. Damit die Metadaten aggregiert werden können, müssen die Teilnehmer zunächst ihre Metadaten über ein Webinterface manuell hochladen. Teilnehmer an der Inter-Föderation eduGAIN, wie der IdP LRZ, müssen zusätzlich, und unabhängig von der Verlässlichkeitsklasse, die Metadaten von eduGAIN über einen *MetadataProvider* laden, wie in Abbildung 2.12 dargestellt. Der Metadatensatz der Umbrella-Föderation enthält aktuell (Januar 2016) 270.000 Zeilen XML-Code, der von 2601 SPs und IdPs stammt. Jeder einzelne Teilnehmer hat jedoch

nur zu einer Teilmenge der darin enthaltenen Entitäten eine Kooperation. Das LRZ hatte beispielsweise vom 19.01.2014 bis zum 26.01.2014 19 Authentifizierungen für SPs, davon

- 10 für den SP mit der EntityID `https://sp.tshhosting.com/shibboleth`
- 3 für `https://www.edupsy.moodle.elearning.lmu.de/shibboleth/`
- 2 für den Dienst Gigamove `https://gigamove.rz.rwth-aachen.de/shibboleth`
- 1 für `https://www.gwi.moodle.elearning.lmu.de/shibboleth/`
- 1 für den Web-Konferenz-Dienst des DFN-Verein `https://webconf.vc.dfn.de/shibboleth`
- 1 für Foodle `https://foodl.org/simplesaml/module.php/saml/sp/metadata.php/saml`
- 1 für den SP mit der EntityID `https://cast.itunes.uni-muenchen.de/shibboleth`

Die DFN-AAI, Teilnehmer an der Inter-Föderation, aggregiert die Metadaten der eduGAIN-Entitäten aus ihrer Föderation, die dem eduGAIN Metadata Profile entsprechen müssen. Dieser signierte Metadatensatz wird vom MDS abgerufen, validiert, mit den Metadaten der anderen teilnehmenden Föderationen aggregiert und signiert. Die DFN-AAI filtert aus Gründen der Konsistenz die eigenen eduGAIN-Teilnehmer aus dem Metadatensatz heraus, der anschließend von eduGAIN-Teilnehmern, wie dem Leibniz-Rechenzentrum, geladen wird.

Während in der nationalen Föderation DFN-AAI [DFN15b] die Attribute aus den Schemata `eduPerson`, `SCHAC` und `dfnEduPerson` von den IdPs zur Verfügung gestellt werden müssen, sind die Schemata `eduPerson` und `SCHAC` in der Inter-Föderation vielleicht (`MAY`) eingebunden [LS15]. Daher gibt es nur eine Empfehlung von wenigen Attributen, welche IdPs in der Lage sein sollen herauszugeben:

- `displayName` aus dem Schema `eduPerson`: Bevorzugte Bezeichnung der Person, die angezeigt werden soll. Meist besteht der `displayName` aus Vorname und Name der Person, wobei die Reihenfolge vom jeweiligen Land abhängt. Zum Beispiel `Daniela Pöhn`.
- `commonname (cn)` aus dem Schema `eduPerson`: Bezeichnung der Person, in der Regel Vorname und Nachname, wobei die Reihenfolge ebenfalls vom jeweiligen Land abhängt.
- `mail` aus dem Schema `eduPerson`: E-Mail-Adresse des Nutzers
- `eduPersonAffiliation` und `eduPersonScopedAffiliation`, beide aus dem Schema `eduPerson`: Bezeichnung der Beziehung des Benutzers zur Heimatorganisation, zum Beispiel `staff` bzw. `staff@lrz.de`.

- `eduPersonPrincipleName` aus dem Schema `eduPerson`: Eindeutige Bezeichnung des Benutzers innerhalb des Geltungsbereiches, wie `di34koj@lrz.de`
- `eduPersonTargetedID` aus dem Schema `eduPerson`: Persistente Identifier (ID) des Benutzers.
- `schacHomeOrganization` aus dem Schema `SCHAC`: Domain Name der Heimatorganisation, z. B. `lrz.de`.
- `schacHomeOrganizationType` aus dem Schema `SCHAC`: Uniform Resource Name (URN) der Heimatorganisation, beispielsweise `urn:mace:terena.org:schac:homeOrganization-Type:eu:higherEducationalInstitution`.

Die Attribute müssen als Minimalvorgabe das *MACE-Dir SAML Attribute Profile* [Int16] einhalten. Das Profil der Internet2 Middleware Architecture Committee for Education – Directory (MACE-Dir) Arbeitsgruppe gibt die Syntax der Assertion an. So soll das Attribut u. a. über eine Uniform Resource Identifier (URI) und ein Object Identifier (OID) spezifiziert sein, bevor der Wert genannt wird, wie in Zeilen 1 und 2 im folgenden Beispiel 2.1 zu sehen.

```
1  <saml2:Attribute
2    xmlns:x500="urn:oasis:names:tc:SAML:2.0:profiles:attribute:X500"
3    x500:Encoding="LDAP"
4    NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
5    Name="urn:oid:2.5.4.42"
6    FriendlyName="givenName">
7      <saml2:AttributeValue xsi:type="xsd:string">
8              Daniela
9      </saml2:AttributeValue>
10 </saml2:Attribute>
```

Listing 2.1: Syntax einer Assertion

Jedoch kann die Syntax und Semantik der Attribute selbst vom jeweiligen Land oder sogar von der jeweiligen Entität abhängen. Somit kann ein Service Provider, der vom IdP LRZ Benutzerinformationen anfordert, nicht sicher sein, ob

- der IdP LRZ ihm vertraut,
- der IdP LRZ die angeforderten Informationen vollständig unterstützen kann,
- der IdP LRZ erst manuell seine Konfiguration bezüglich der Konvertierung und Filterung der Attribute anpassen muss,
- der IdP LRZ ihm alle Benutzerinformationen sendet,
- das Verständnis über ein Attribut dasselbe ist,

- der IdP LRZ die benötigten Qualitätsanforderungen einhält.

Folglich ist es möglich, dass der IdP LRZ die vom Service Provider für seinen Dienst benötigten Attribute nicht liefern kann. Durch diese Problematik kann es der Fall sein, dass ein Nutzer vom LRZ einen Dienst nicht oder nur unvollständig verwenden kann.

Workflows

Nachdem sich die Entitäten in der Inter-Föderation eduGAIN nur bedingt vertrauen, verkompliziert sich der Inter-FIM Workflow aus dem letzten Abschnitt wie folgt:

1. Schritt: Der IdP LRZ zeichnet einen Vertrag mit der DFN-AAI.

2. Schritt: Der IdP LRZ lädt seine Metadaten über ein Webinterface zur DFN-AAI und erhält im Gegenzug die Föderationsmetadaten.

3. Schritt: Der IdP LRZ entschließt sich explizit per Opt-In an der Inter-Föderation eduGAIN teilzunehmen.

4. Schritt: Der Benutzer des LRZs möchte einen Dienst beim gewünschten Service Provider nutzen.

5. Schritt: Der Resource Monitor überprüft, ob der Nutzer eine aktive Session hat und, wenn nicht, leitet ihn zum SP weiter um den SSO-Prozess zu starten.

6. Schritt: Der Benutzer wählt seinen IdP LRZ beim Discovery Service aus.

7. Schritt: Der SP bereitet einen *Authentication Request* vor und sendet diesen zum Identity Provider des Nutzers. Parallel dazu wird der Nutzer zum IdP umgeleitet.

8. Schritt: Der Benutzer authentifiziert sich bei seinem IdP.

9. Schritt: Der IdP LRZ überprüft, ob der SP den CoCo akzeptiert und eine Entity Category angegeben hat.

10. Schritt: Der Identity Provider leitet den Nutzer und sendet einen *Authentication Response* zum Service Provider. Dieser *Authentication Response* enthält je nach Status des SPs entweder die benötigten Attribute oder nicht.

11. Schritt: Der Service Provider validiert den Response und leitet den Nutzer zum Dienst weiter, der diesen verwenden kann.

12. Schritt: Der Nutzer stellt fest, dass er den Dienst nicht vollständig nutzen kann und gibt ein Incident-Ticket beim Helpdesk seines IdPs auf. In diesem Ticket beschreibt er

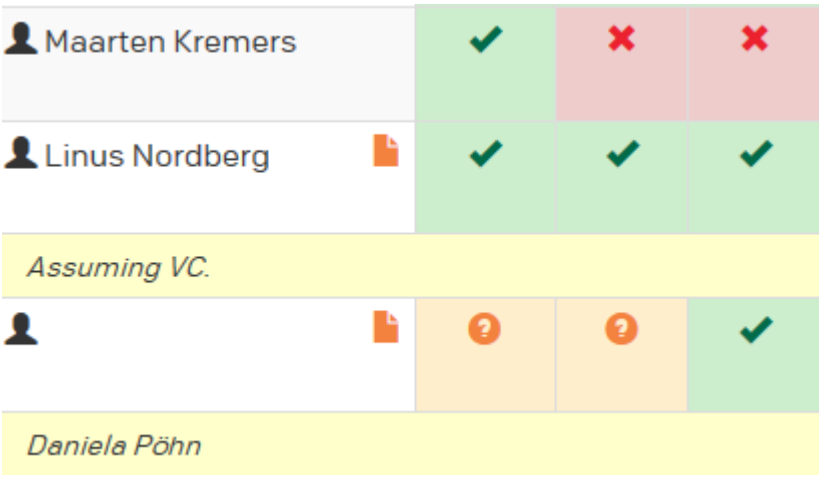

Abbildung 2.13.: Eingeschränkte Nutzung am Beispiel einer Foodle-Abfrage

das Problem mit seinen eigenen Worten, da es keine geeignete Fehlermeldung gibt.

13. Schritt: Sobald der IdP LRZ alle benötigten Informationen vom Nutzer erhalten hat, passt er seine Konfiguration manuell an. Dazu muss er bei Bedarf Konvertierungsregeln für Attribute in der `attribute-resolver.xml` erstellen und den Attributfilter `attribute-filter.xml` anpassen.

14. Schritt: Der Nutzer des IdP LRZ kann den Dienst erfolgreich verwenden.

Inzwischen wird die Mitgliedschaft in einer Entity Category, die SPs mit bestimmten Merkmalen zu Gruppen zusammenfasst, überprüft. Diese können von Föderationen und Inter-Föderationen festgelegt werden. Innerhalb der DFN-AAI sind die folgenden Entity Categories verfügbar:

- Mitglieder des bwIdM-Projektes, `http://aai.dfn.de/category/bwidm-member`.
- Einrichtungen, die am KELDAT-Projekt im Bereich der Tiermedizin teilnehmen, `http://aai.dfn.de/category/vetmed-member`,
- `http://aai.dfn.de/category/DiepRuR`, welche noch nicht produktiv ist.
- Die Zustimmung zum Code of Conduct in eduGAIN, `http://www.geant.net/uri/dataprotection-code-of-conduct/v1`.
- Service Provider, die Forschung und Lehre unterstützen, `https://refeds.org/category/research-and-scholarship`.

Es ist jedoch möglich, dass der SP zwar den CoCo akzeptiert hat, aber dies noch nicht in den Föderationsmetadaten eingetragen wurde, wie beispielsweise 2014 beim Dienst GÉANT Intranet geschehen, oder trotzdem wichtige Attribute nicht herausgegeben werden. Ferner sind nicht alle SPs Teil einer Entity Category oder haben den CoCo unterzeichnet. Dadurch hat der Nutzer den Mehraufwand die Vertrauensbeziehung zu initiieren. Gleichzeitig muss der IdP manuell die lokale Konfiguration anpassen, was zu einer Wartezeit für den Benutzer führt. Dies hat wiederum zur Folge, dass der Nutzer womöglich einen anderen Weg versucht den Dienst zu nutzen, z. B. durch Anlegen einer lokalen Nutzerkennung, oder einen anderen Dienst ausprobiert, bei dem er auf keine Probleme trifft. Zusätzlich kann es sein, dass der Nutzer seinen Identity Provider gar nicht über seine Nutzungswünsche informiert und den Dienst so verwendet, wie er ihn anfangs angetroffen hat. Beispielsweise kann der Benutzer ein Foodle-System benutzen, bei dem er zwar seine Terminwünsche eintragen kann, diese aber nicht mit seinem Namen vermerkt sind, wie in Abbildung 2.13 ersichtlich. Stattdessen fügt der Nutzer seinen Namen per Kommentarfunktion ein. Ein weiterer Aspekt sind die unterschiedlichen Implementierungen von SAML, die bezüglich einzelner Aspekte, beispielsweise der Eindeutigkeit von Attributen, nicht ohne zusätzliche Konfiguration kompatibel sind.

Datenschutz und Trust

Die nationalen Föderationen, wie die DFN-AAI, stellen vertrauenswürdige Infrastrukturen zur Verfügung und jeder Teilnehmer an der DFN-AAI muss zunächst einen Vertrag abschließen; trotzdem herrscht nur bedingtes Vertrauen vor. Je nach Föderation sind die Entitäten aufgefordert eigene bilaterale Verträge abzuschließen und Vertrauen aufzubauen. Dies skaliert mit zunehmender Teilnehmerzahl schlecht. Der Code of Conduct, der auch vom LRZ geprüft wird, zielt darauf hinaus, dass IdPs leichter Benutzerinformationen verschicken. Da laut der Datenschutzrichtlinie der EU nicht mehr persönliche Informationen vom Service Provider verlangt werden können als er unbedingt benötigt und er die Informationen nicht an Dritte weiterleitet, sollen die angeforderten Attribute auch ausgeliefert werden. Für die Verarbeitung der Daten durch Dritte muss eine datenschutzrechtliche Freigabe vorliegen. Im Gegensatz dazu stehen verschiedene Policies, möglichst keine benutzerbezogene Attribute, wie beispielsweise E-Mail-Adresse, zu verschicken. Die Übertragung der Daten muss die Vertraulichkeit, beispielsweise durch Verschlüsselung, sicherstellen. Gleichzeitig muss der IdP dem Benutzer gegenüber Auskunft geben, welche Daten er an den jeweiligen SP übermittelt und diesen fragen, ob er der Übermittlung zustimmt. Dieser User Consent wird beispielsweise durch das Tool uApprove [SWI14] abgefragt, welches im nachfolgenden Kapitel näher beschrieben wird. Der Nutzer kann entweder der Weitergabe seiner Daten zustimmen oder dies ablehnen.

Basismodell und Dienstsicht

Bei der Anwendung des Basismodells auf Inter-FIM (vgl. Abbildung 2.14) mit dem Beispiel eduGAIN zeigt bereits eine komplexere Modellierung im Vergleich zum Basismodell für FIM.

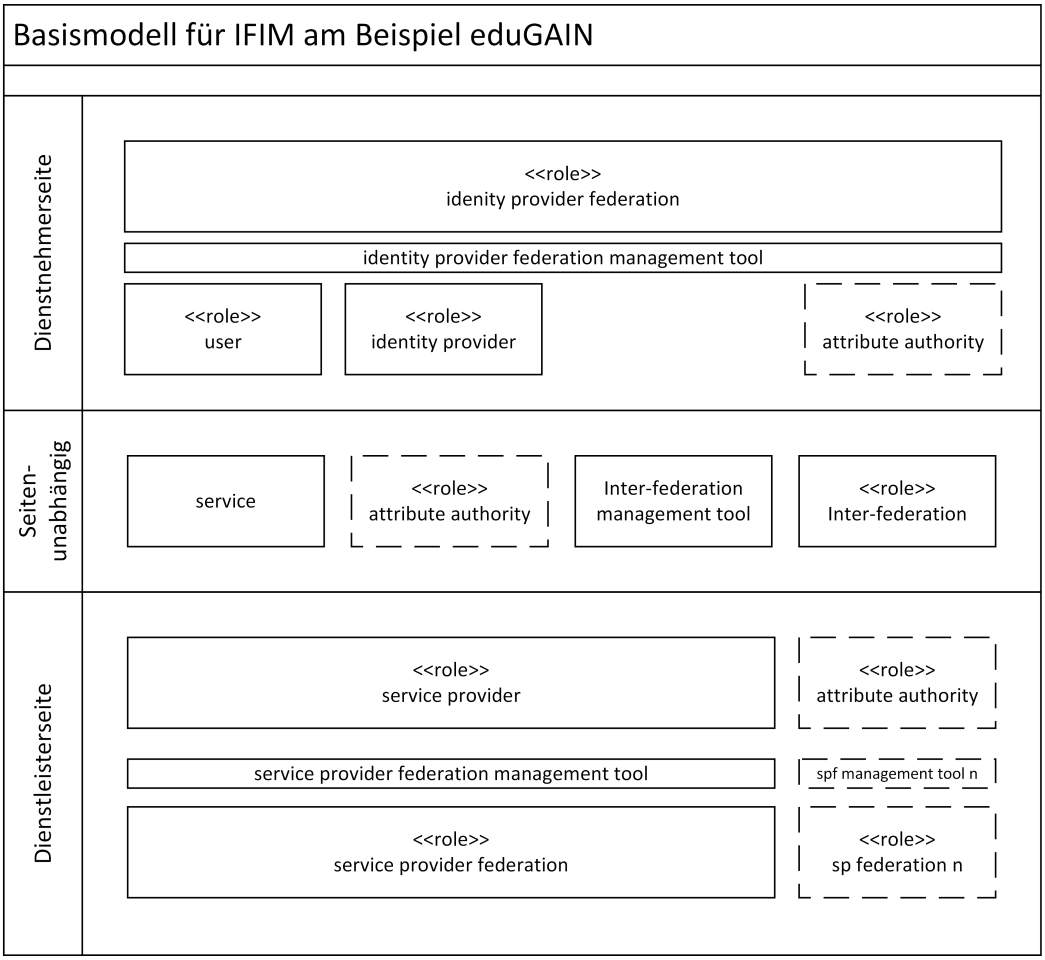

Abbildung 2.14.: Basismodell der Inter-Föderation eduGAIN

- Seitenunabhängig wird die Rolle *Inter-Föderation* hinzugefügt.
- Sowohl IdP Föderation, SP Föderation als auch die neue Rolle Inter-Föderation betreiben ein *Management Tool* um die jeweilige Föderation zu verwalten.
- Jede Föderation kann zusätzlich zu einer unabhängigen Attribute Authority noch *eigene AAs* betreiben.
- Der Service Provider kann *mehreren Föderationen* zugehören, die alle ein *eigenes Management Tool* betreiben.

Diese zusätzlichen Werkzeuge und Rollen spiegeln sich ebenfalls bei der Dienstsicht wieder, wie in Abbildung 2.15 zu sehen. Das Delta zur Dienstsicht von Federated Identity Management lautet wie folgt:

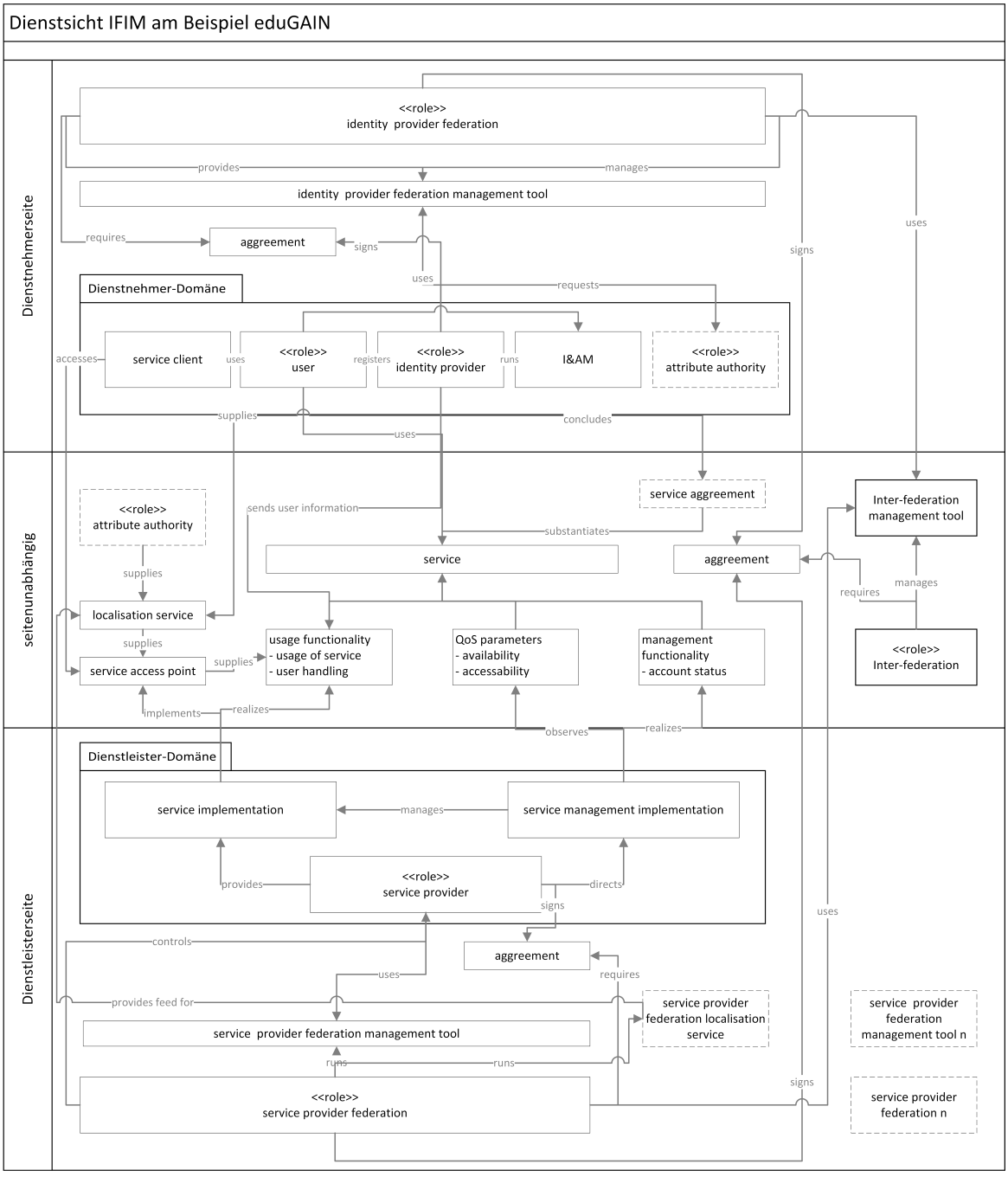

Abbildung 2.15.: Dienstsicht der Inter-Föderation eduGAIN

- IdP Föderation und SP Föderation betreiben ein *Management Tool*, welches von den IdPs bzw. SPs verwendet wird.
- Zudem schließen die Föderationen *Verträge* mit ihren IdPs und SPs.
- Jeder Service Provider (und theoretisch auch Identity Provider) kann *mehreren Föderationen* zugehören, die ihr eigenes Management Tool betreiben. Diese Föderationen können, müssen aber nicht, einen Vertrag mit der Inter-Föderation abgeschlossen haben.
- Die Rolle *Inter-Föderation* betreibt ebenfalls ein *Management Tool*, bei eduGAIN ist dies Metadata Distribution Service.
- Die Inter-Föderation schließt wiederum *Verträge* mit den zugehörigen Föderationen.
- Neben einem seitenunabhängigen Lokalisierungsdienst kann zudem ein Lokalisierungsdienst auf Dienstleisterseite betrieben werden.
- Etwas vereinfacht dargestellt können IdP Föderation und SP Föderation ebenfalls eigene *AAs* betreiben.

Defizite

Aus dem Szenario leiten sich folgende Defizite ab:

- Der Identity Provider muss explizit der Teilnahme an der Inter-Föderation eduGAIN zustimmen.
- Der Service Provider kann Benutzerinformationen anfordern, die der IdP nicht hat.
- Die Semantik und Syntax der gelieferten Attribute ist nicht einheitlich festgelegt.
- Das Vertrauen zwischen Identity Provider und Service Provider muss erst manuell aufgebaut werden.
- Es ist möglich, dass der Identity Provider nicht alle benötigten Informationen sendet, beispielsweise weil er nicht in der Lage ist oder er dem Service Provider nicht bzw. nicht genügend vertraut. Folglich kann der Nutzer nicht alle Dienste einwandfrei nutzbar sind.
- In diesem Fall muss im Normalfall der Nutzer aktiv werden, der teilweise nicht weiß, wer sein Identity Provider ist und was er machen muss.
- Der Identity Provider muss manuell die Konfiguration anpassen.

- Durch das manuelle Eingreifen kommt es zu einer Wartezeit für den Benutzer, der dabei das Interesse verlieren kann oder nach anderen Wegen sucht den Dienst zu nutzen.
- Die Größe der aggregierten Metadaten skaliert schlecht. Obwohl nur ein Bruchteil des Metadatensatzes benötigt wird, hat jede Entität die gesamten aggregierten Metadaten in der lokalen Software gespeichert. Dies verlangsamt die Software insbesondere bei älterer Hardware.
- Der Service Provider kann nicht prüfen, welche Qualitätsanforderungen der Identity Provider einhält.
- Die Dienste von SPs außerhalb der Inter-Föderation eduGAIN und der nationalen Föderation DFN-AAI können nicht mit den Vorteilen des FIMs genutzt werden.
- Der Nutzer kann nur dem Versand der aufgelisteten Attribute zustimmen oder ablehnen, jedoch keine individuelle, feingranulare Auswahl treffen. Der Nutzer kann nicht mehr Attribute dem Service Provider zur Verfügung stellen als sein Identity Provider versendet.
- Zusätzlich kann es sein, dass die erhaltenen Metadaten nicht aktuell sind, weil die Metadaten der Föderationen und Inter-Föderation in bestimmten Abständen aktualisiert werden.
- Die Informationen an den Nutzer bei Fehlern sind kaum aussagekräftig. Zudem gibt es kein integriertes Werkzeug, um die Fehlermeldung an den IdP weiter zu geben. Dadurch kann es sein, dass der Identity Provider nicht alle benötigten Informationen zur Fehlerbehebung erhält.
- Der IdP muss bei der Teilnahme an der Inter-Föderation eduGAIN mehrere Metadaten erstellen bzw. seine Metadaten anpassen.

Im folgenden Abschnitt wird beschrieben, welche Verbesserungen durch den Einsatz einer geeigneten Föderation erreicht werden können.

Ziele

In diesem Szenario wird davon ausgegangen, dass der Aufbau von Vertrauensbeziehungen zwischen Identity Providern und Service Providern möglich ist und dass diese über einen Zeitraum konstant bleiben. Gleichzeitig soll beachtet werden, dass nicht alle Dienste, die genutzt werden sollen, Mitglied einer Föderation oder Inter-Föderation sind. Um die Probleme beim Aufbau von Vertrauensbeziehungen zu lösen, wird durch eine virtuelle Föderation eine maximale Reichweite mit einer Vereinfachung durch Automatisierung für die beteiligten Entitäten angestrebt. Dadurch würden sich folgende Verbesserungen ergeben:

- Die Föderationen werden entsprechend den Nutzungswünschen der Benutzer gebildet, d. h. die Metadaten werden nicht vorab aggregiert, sondern nach Bedarf ausgetauscht. Dadurch muss der Dienst nicht unbedingt Mitglied in der eigenen Föderation bzw. Inter-Föderation sein.

- Der IdP benötigt nur Metadaten der Geschäftspartner, anstelle von mehreren Metadatensätzen, d. h. pro Föderation und Inter-Föderation.

- Es werden nur die benötigten Metadaten ausgetauscht, wodurch die Größe des Metadatensatzes reduziert wird. Dies wiederum wirkt sich positiv auf die Performance aus.

- Das Vertrauen wird automatisch und on demand gebildet.

- Die Daten werden automatisch und on demand aktualisiert.

- Der Administrator soll trotzdem die Möglichkeit haben informiert zu werden, bestimmte Konfigurationen selbst vorzunehmen oder eingreifen zu können, falls dies aus bestimmten Gründen erforderlich ist.

- Die Konvertierung von Attributen wird erleichtert. Dadurch können unterschiedliche Schemata verwendet werden, was Sonderwünschen der Service Provider entgegen kommt. Folglich ist die Wahrscheinlichkeit größer, dass jeder SP alle benötigten Attribute erhält.

- Der Benutzer muss im Normalfall nicht mehr aktiv werden.

- Durch geeignete Schnittstellen zum Incident Response Prozess und zum Change Management können Fehler schnell behoben und die dafür nötigen Änderungen prozessorientiert durchgeführt sowie gut dokumentiert werden. Gleichzeitig kann der Nutzer in Ausnahmefällen durch ein Werkzeug unterstützt Fehler an seinen IdP weitergeben.

- Service Provider autorisieren nur Nutzer, deren IdPs die geforderten Qualitätsanforderungen einhalten können. Dazu ist es möglich die Verlässlichkeitsklassen der einzelnen Föderationen aufeinander abzubilden und zu vergleichen.

- Der Benutzer hat eine stärkere Kontrolle über die weitergegebenen Informationen.

- Die Benutzung der Dienste ist einfacher und fehlerfrei möglich, was zu einer gesteigerten Akzeptanz durch die Nutzer führt.

- Die Einhaltung der Datenschutzrichtlinien wird durch entsprechende Werkzeuge unterstützt.

- Schnittstellen zu internen Systemen bleiben erhalten bzw. werden, wie im Fehlermanagement und Sicherheitsmanagement, verbessert.

Das Konstrukt der Umbrella-Föderation eduGAIN würde durch gebrauchsgesteuerte Föderationen abgelöst.

2.3.7. Anforderungen

Zusätzlich zu den bisher genannten Anforderungen aus dem Federated Identity Management, gelten folgende Punkte.

Funktionale Anforderungen

Die funktionalen Anforderungen werden durch die Anforderungen und Gegebenheiten aus der Inter-Föderation um folgende Anforderungen ergänzt:

- Die Bildung und Verwaltung mehrerer Föderationen muss unterstützt werden [FA-Föderation].
- Die generierten Metadaten passen zu allen teilnehmenden Föderationen, wodurch [FA-Metadaten] um die generische Struktur erweitert wird.
- Der Nutzer muss den Austausch von Daten und somit Aufbau des Vertrauensverhältnisses initiieren können, um u. a. die Wartezeit zu verringern [FA-Initiierung].
- Der Vertrauensaufbau muss on demand und automatisch geschehen können, um so dynamisch an geänderte Anforderungen der Nutzer reagieren zu können [FA-Automatisierung].
- Es müssen alle benötigten Attribute unabhängig vom Schema der Inter-Föderation oder Föderation versendet und interpretiert werden können. Dies hat zur Folge, dass die Dienste in vollem Umfang genutzt werden können [FA-Schema].
- Die Benutzerinformationen müssen über nationale Grenzen hinweg versendet werden können [FA-Grenzüberschreitend].
- Die Architektur der Föderationen muss gleichzeitig eine möglichst langfristige und dauerhafte Lösung ermöglichen [FA-Langlebigkeit].
- Der Administrator muss die Möglichkeit haben die Integration der Metadaten und den Vertrauensaustausch zu konfigurieren. Bei Bedarf muss der IdP bzw. SP Administrator über Änderungen informiert werden können, um manuell eingreifen zu können und um einen Überblick über Änderungen zu haben [FA-Konfiguration].

Nichtfunktionale technische Anforderungen

Basierend auf der aktuellen Lösung ergeben sich folgende zusätzliche nichtfunktionale technische Anforderungen:

- Es soll möglich sein, dass eine Entität mehreren Föderationen zugehört [NFA-Koexistenz].
- Das System muss unterschiedliche SAML-Implementierungen, beispielsweise Shibboleth und SimpleSAMLphp, akzeptieren, um einen möglichst großen Nutzerkreis abzudecken [NFA-Implementierungsunabhängigkeit].

Sicherheitsanforderungen

Trotz der Automatisierung des Datenaustausches durch [FA-Initiierung] muss die Sicherheit gewährleistet werden. Folglich gelten [FA-Automatisierung] und [FA-Initiierung] auch als Sicherheitsanforderungen.

Organisatorische Anforderungen

Die folgende Anforderung reflektiert die Kooperationen von Organisationen der nationalen Föderation:

- Die Organisationen müssen sich in virtuellen Föderationen registrieren können [ORG-Registrierung].
- Ferner muss ein organisationsübergreifendes Modell zum Datenaustausch verwendet werden. Folglich muss die FIM-Lösung mit dem internen Schema und dem ggf. verwendeten externen Schema umgehen können, was die Anforderung [FA-Schema] um organisatorische Aspekte erweitert.
- Die Bildung und Verwaltung nach verschiedenen Föderationsmodellen muss passend durch die Lösung unterstützt werden. Folglich gilt [FA-Föderation] auch als organisatorische Anforderung.
- Die Metadaten [FA-Metadaten] enthalten Informationen über die Organisationen und müssen passend generiert werden.
- Zusätzlich gilt die Anforderung [FA-Konfiguration] auch als organisatorische Anforderung.

Datenschutzrechtliche Anforderungen

Da durch die Initiierung und Automatisierung des Vertrauensaufbaus durch den Nutzer Benutzerinformationen an den gewählten Service Provider nach [DSA-Zustimmung] geschickt werden, ist die Initiierung ebenfalls datenschutzrechtlich relevant. Dies muss dem Benutzer geeignet dargestellt werden. Somit gilt [FA-Initiierung] ebenfalls für den Datenschutz.

2.4. Federated Identity Management in Forschungsgruppen

Definition 5. Communities

Communities sind Forschungsgruppen, die eigene Föderationen, so genannte Virtuelle Föderationen, gründen, um zusammenarbeiten zu können.

In diesem Abschnitt werden die Anforderungen der Forschungsgruppen (*Communities*) erklärt und die Problematiken anhand von zwei Szenarien vertieft. Da u. a. Unzulänglichkeiten der aktuellen Inter-Föderation eduGAIN zur Gründung der Föderationen der Forschungsgruppen geführt haben, können ausgewählte Anforderungen aus den Communities auf dynamische virtuelle Föderationen übernommen werden. Nach einer kurzen Beschreibung der Motivation, illustrieren zwei Szenarien aus den Communities die Workflows, Datenschutzaspekte und Defizite.

2.4.1. Motivation

Existierende Föderationen und Inter-Föderationen werden bisher den Anforderungen der Communities nicht gerecht. Auf Grund der beschriebenen Problematik bilden Communities, wie beispielsweise im Bereich des Climate Science (z. B. Earth System Grid Federation (ESGF)), eigene Föderationen, die parallel zu den nationalen Föderationen existieren. Im Gegensatz zu den nationalen Föderationen ist die Zusammenarbeit der Forschungsgruppen nicht nur auf FIM bezogen.

Die zugehörige Datenbasis erweitert teilweise die Attribute der Inter-Föderationen um zusätzliche, spezifische Informationen, die beispielsweise im Grid-Umfeld für die Autorisierung benötigt werden und die nicht im gemeinsamen Datenschema der Inter-Föderation eduGAIN enthalten sind. Für die Wissenschaftler ergibt sich das Problem, dass sie diverse digitale IDs besitzen, eine ID pro Community und Föderation, plus etliche Social IDs für ihr Privatleben. Somit wird ein wesentlicher Vorteil von FIM und Inter-FIM – der Zugriff auf beliebige Dienste mit der Kennung der Heimateinrichtung – eingebüßt.

Szenario 1 behandelt die Community CLARIN, die die Anforderungen einer web-basier-

ten Community an IdPs aufzeigt. Im Szenario 2 wird auf die Bedürfnisse im Grid-Umfeld eingegangen.

2.4.2. Szenario 2: CLARIN im europäischen Kontext

Common Language Resources and Technology Infrastructure ist ein Forschungsverbund zur Archivierung und Verarbeitung von Sprachdateien. CLARIN stellt linguistische Daten, Werkzeuge und Infrastruktur für die Disziplinen Geisteswissenschaften und Sozialwissenschaften bereit. Der Verbund besteht aus Wissenschaftlern mehrerer Disziplinen, die CLARIN als Plattform zum Austausch von Sprachdateien verwenden. Gleichzeitig sind mehrerer Institutionen, u. a. aus Deutschland, Mitglied in dem Forschungsverbund, der Dienstleistungen anbietet. Der Dachverband CLARIN European Research Infrastructure Consortium (ERIC) ist auf europäischer Ebene dafür zuständig, die Infrastruktur bereit zu halten, zu aktualisieren und zu verbessern.

Ausgangssituation

CLARIN ERIC bietet europaweit ein Portal für die verschiedenen Archive, Dienste und Werkzeuge an. In CLARIN-D [CLA16], dem deutschen Dachverband, sind das z. B.:

- Archive, d. h. eine thematische heterogene Sammlung von Sprachdateien, wie das Bayerische Archiv für Sprachsignale und Archiv für gesprochenes Deutsch.
- Korpora, eine maschinenlesbare Sammlung von Texten, um den Sprachgebrauch zu einem bestimmten Zeitabschnitt oder in einer bestimmten Varietät zu beschreiben, z. B. *HEMPEL* und das *Darmstadt Corpus for Scientific Texts*.
- Lexikalische Ressourcen, beispielsweise das Aussprachelexikon *PHONOLEX*.
- Werkzeuge und Dienste, wie die *WebMAUS*, die vollautomatisch Sprachaufnahmen in Phoneme segmentiert und beschriftet.

Damit CLARIN diese Dienste europaweit mit Hilfe von Federated Identity Management anbieten kann, gibt es einen Zusammenschluss der Dienstbetreiber von CLARIN. Im Gegensatz zur organisatorischen Einheit der Föderation aus dem vorherigen Abschnitt, besteht die SP-Federation von CLARIN aus einer Einheit aus verschiedenen Service Providern. Der Homeless IdP wird einzig für Wissenschaftler betrieben, die keine Heimatorganisation in den kooperierenden Föderationen haben. Diese koordinierte SP-Föderation ist wiederum mit zahlreichen nationalen Föderationen verbunden, damit möglichst viele ihrer Mitglieder mittels FIM-Technologien die Dienste von CLARIN nutzen können. Die Zusammenarbeit erfolgt rein um die Ziele der Community zu verfolgen, während die Art der Kooperation als komplex angesehen werden kann, da Teilnehmer der nationalen Föderationen nicht unbedingt mitein-

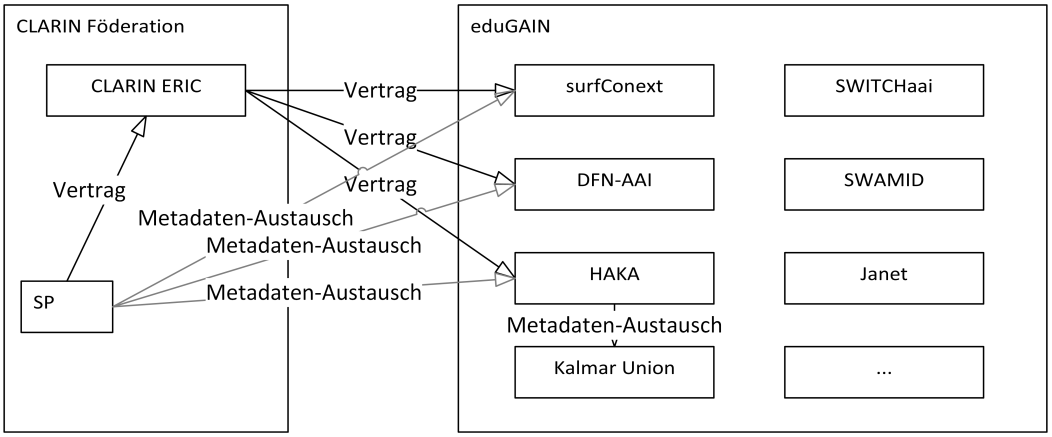

Abbildung 2.16.: Architektur der CLARIN-Föderation

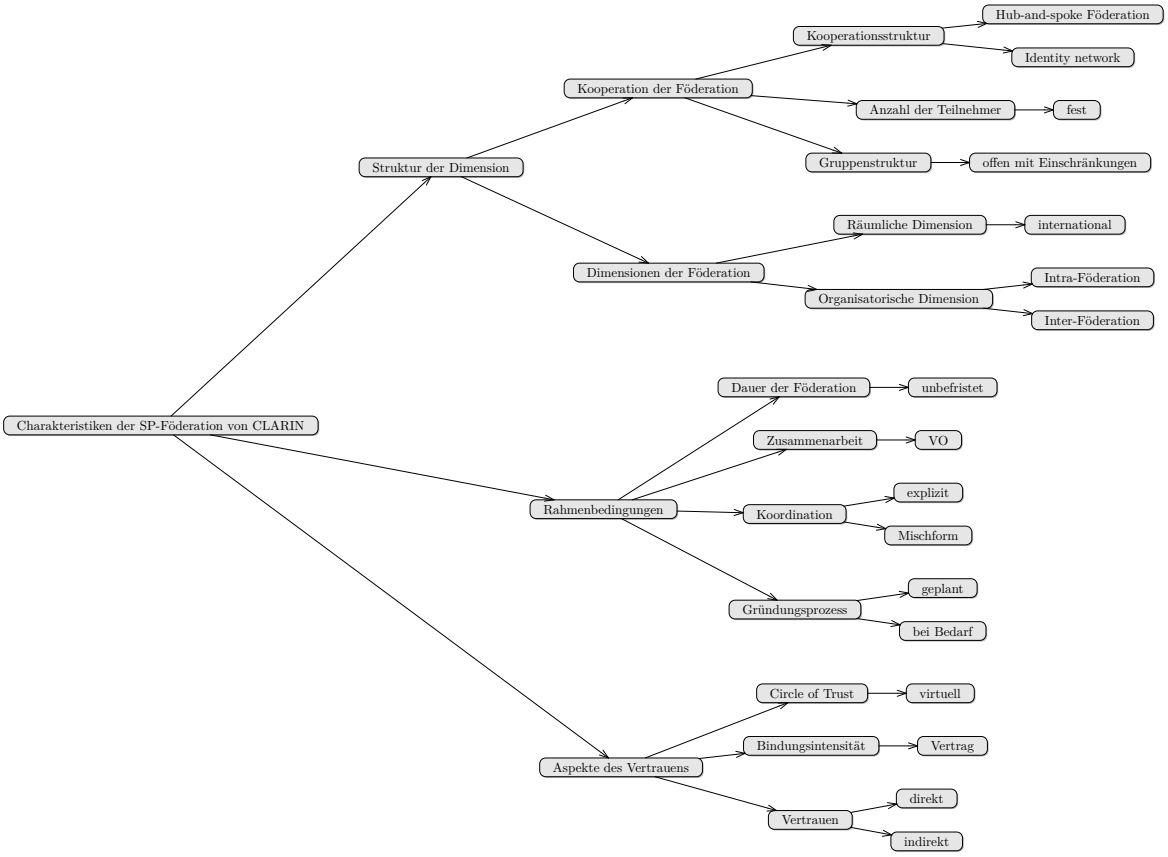

Abbildung 2.17.: Klassifikation der SP-Föderation von CLARIN

ander kooperieren können. Um an der Service Provider Federation teilzunehmen, muss jede CLARIN-Institution einen Vertrag mit CLARIN ERIC unterzeichnen, der u. a. Finanzie-

rung, Datenschutz, Haftung, Rechte und Pflichten klärt. CLARIN ERIC schließt daraufhin Verträge mit den nationalen Föderationen, so dass die Nutzer in CLARIN-Institutionen, aber auch in anderen Instituten und Hochschulen, über Federated Identity Management auf die Sprachdateien zugreifen können. Gleichzeitig kooperieren die nationalen Föderationen, die eine Teilmenge der Inter-Föderation eduGAIN darstellen, nicht durch diesen Zusammenschluss. Dieses Konzept ist in der Abbildung 2.16 dargestellt. Basierend auf der Kategorisierung von Föderationen aus Abschnitt 2.2 kann die CLARIN Service Provider Federation wie folgt beschrieben werden (vgl. Abbildung 2.17):

- Die Kooperationsstruktur besteht aus dem Identity network von CLARIN sowie aus Hub-and-spoke Föderationen und Identity networks in den kooperierenden Föderationen, die über Verträge miteinander verbunden sind.
- Die Anzahl der Teilnehmer ist auf Grund der Föderationsstrukturen fest.
- Die Gruppenstruktur entspricht der in den einzelnen Föderationen und ist somit offen mit Beschränkungen.
- Die Service Provider Federation ist international.
- Die organisatorische Dimension entspricht aus meso-Sicht einer Föderation und aus makro-Sicht auf Grund der Kooperationen mit Föderationen einer Inter-Föderation.
- Die Service Provider Federation ist langfristig ausgelegt ohne einem festgelegten Enddatum.
- Bei der Zusammenarbeit steht die Bereitstellung der Dienste für die Community im Vordergrund.
- Die Koordination erfolgt explizit innerhalb der SP-Federation, während die Koordination in den kooperierenden Föderationen auch implizit sein kann.
- Der Gründungsprozess ist geplant auf Grund des bestehenden Bedarfs, der durch eduGAIN nicht gedeckt wurde.
- Der Circle of Trust ist tendenziell statisch, wobei auf Grund der Größe der Metaorganisation keine Organisation vollständige Informationen über die anderen Teilnehmer haben kann.
- Die Bindung geschieht über einen Vertrag, während die Bindungsintensität zur lokalen SP-Federation größer ist als zu den kooperierenden Föderationen.
- Das Vertrauen zwischen zwei Organisationen kann sowohl direkt, beispielsweise bei Kooperationen, als auch indirekt sein.

Folglich ist die SP-Föderation aus makro-Sicht gleichzeitig eine Inter-Föderation, durch die die teilnehmenden nationalen Föderationen nicht untereinander kooperieren. Bei der Inter-Föderation CLARIN steht im Gegensatz zur Inter-Föderation eduGAIN die Bereitstellung der Dienst für die Community im Vordergrund.

Externe Nutzer können den CLARIN Homeless IdP benutzen, um vollen Zugriff zu erhalten. Die Benutzerverwaltung findet beim Identity Provider des Nutzers statt, der entweder das Attribut `eduPersonPrincipalName` oder `eduPersonTargetedID` an den Service Provider senden muss, damit der Nutzer vollen Zugriff auf den Dienst bekommt. Für eine optimale Nutzung aller CLARIN-Dienste wird ein Set aus folgenden Attributen benötigt:

- `eduPersonPrincipalName` oder `eduPersonTargetedID`,
- `cn`,
- `mail` und
- `organizationName (o)` oder `schacHomeOrganization`.

Das Set hat theoretisch den Vorteil, dass die Attribute im Einzelfall nicht neu verhandelt werden müssen. So benötigt beispielsweise `Lat.csc.fi` im Moment nur `eduPersonPrincipalName`. Jedoch sind zudem `cn` und `mail` sinnvoll, um beispielsweise Bestätigungsmails zu verschicken. `Korp.csc.fi` dagegen verwendet `eduPersonPrincipalName` und `eduPersonAffiliation`, um wissenschaftlichen Mitarbeitern und Postdocs automatischen Zugang zu Clarin_ACA-lizensierten Resourcen zu geben. Dafür wird das Attribut `mail` bei Korp nicht benötigt.

Damit Nutzer der nationalen Föderationen auf die Dienste zurückgreifen können, müssen ihre IdPs den SPs vertrauen, die benötigten Attribute freigeben und verschicken. Dies ist je nach Föderation unterschiedlich geregelt. Während in der finnischen Haka-Föderationen die Entitäten sich gegenseitig vertrauen und automatisch die benötigten Attribute erhalten, muss dieses Vertrauen in der deutschen DFN-AAI erst manuell aufgebaut werden, so dass CLARIN-SPs keine Attribute erhalten, obwohl sie Teil der DFN-AAI sind. In diesem Fall bemerkt der Nutzer das fehlende Vertrauen darin, dass er als `anonymous` eingeloggt ist und nur die Basisdienste nutzen kann, während eingeloggte und privilegierte Nutzer sich die Sprachdateien anhören können. Damit der Nutzer erfolgreich authentifiziert und autorisiert wird, muss je nach Verständnis des IdPs entweder der Service Provider oder der Nutzer den Vertrauensaufbau anstoßen.
Als Gegenmaßnahme wird seit Anfang 2014 der Einsatz des CoCo getestet, der besagt, dass die Service Provider nur die Attribute verlangen, die sie tatsächlich benötigen. Dazu müssen die SPs den Code of Conduct zeichnen und die Beteiligung in den Metadaten anzeigen. Die geringere Teilnehmerzahl in der Inter-Föderation eduGAIN als die Teilnehmer ausgewählter nationaler Föderationen ist teilweise durch die Opt-In Methode bedingt, wodurch Entitäten explizit sich für die Teilnahme an eduGAIN entscheiden müssen. Ferner haben viele IdPs noch nicht die Konfiguration für die automatische Freigabe von Attributen bei der Akzeptanz

des CoCos auf Seiten des SPs angepasst.

Föderation	Entitäten	IdPs
DFN-AAI	471	216
ACOnet	147	42
Belnet	109	41
eduID.cz	280	70
IDEM GARR	191	73
Kalmar2	3417	1652
ArnesAAI	269	60
UK Federation	2917	1632
SURFconext	286	286
CLARIN	4099	4072
eduGAIN	2523	1498

Tabelle 2.2.: Reichweite der SP-Föderation und eduGAIN im Vergleich

Nachdem die Reichweite der CLARIN-SP-Föderation zudem größer war als die der Inter-Föderation eduGAIN, wurde die SP-Federation der Community als die benutzerfreundlichere Lösung angesehen. Aktuell (vgl. Tabelle 2.2, Stand Januar 2016) enthält eduGAIN 1498 IdPs, während CLARIN laut Anzahl der IdPs in den einzelnen Föderationen 4072 IdPs plus ihren Homeless IdP erreicht.

Workflows

Im Gegensatz zu dem FIM-Workflow aus Abschnitt 2.2, wo der Service Provider Teil einer nationalen Föderation ist, muss ein SP bei CLARIN über den Dachverband CLARIN ERIC in mehreren nationalen Föderationen teilnehmen.

1. Schritt: Der Service Provider zeichnet einen Vertrag mit CLARIN ERIC.

2. Schritt: CLARIN ERIC zeichnet Vertrag mit nationalen Föderationen.

3. Schritt: Der SP nimmt dadurch an mehreren Föderationen teil.

4. Schritt: Der SP sendet seine Metadaten an die jeweiligen Föderationen und erhält im Gegenzug die Föderationsmetadaten.

5. Schritt: Der Benutzer möchte einen Dienst nutzen.

6. Schritt: Der Resource Monitor überprüft, ob der Nutzer eine aktive Session hat und, wenn nicht, leitet ihn zum Service Provider weiter um den SSO-Prozess zu starten.

7. Schritt: Der Benutzer wählt seinen Identity Provider beim CLARIN Lokalisierungsdienst

aus.

8. Schritt: Der Service Provider bereitet einen *Authentication Request* vor und sendet diesen mitsamt dem Nutzer zum Identity Provider des Nutzers.

9. Schritt: Der Benutzer authentifiziert sich bei seinem IdP.

10. Schritt: Der Identity Provider sendet den Nutzer und einen *Authentication Response* zum Service Provider.

11. Schritt: Der SP validiert den Response und leitet den Nutzer zum Dienst weiter, der diesen erfolgreich verwenden kann, wenn der IdP dem SP vertraut und die angeforderten Attribute versendet.

Gleichzeitig ist nicht gesichert, dass der Nutzer den Dienst in vollem Umfang einsetzen kann, da auf Grund eines nicht vorhandenen Vertrauensverhältnisses nicht alle IdPs die benötigten Attribute an den SP weiterleiten. Die aggregierten Metadaten in Föderationen und eduGAIN werden in unterschiedlichen Abständen aktualisiert, so dass eine Entität veraltete Daten erhalten kann. Zudem müssen Entitäten Änderungen in den Metadaten selbst bemerken.

Datenschutz und Trust

Die Service Provider von CLARIN halten sich an die EU-Datenschutzrichtlinie 95/46/EG. Das bedeutet, dass CLARIN-SPs innerhalb der EU personenbezogene Daten nur zum Veröffentlichen von Material für CLARIN verwenden und nicht darüber hinaus. Beispielsweise sind CLARIN-SPs verpflichtet die Daten nicht an Dritte herauszugeben und für ein ausreichendes Maß an Sicherheit ihrer Dienste zu sorgen. Außerhalb der EU sollen die Service Provider sich ebenfalls daran halten und adäquate Datenschutzrichtlinien möglichst aufweisen. Dazu muss jeder SP den Code of Conduct unterzeichnen. Die aktuelle Praxis zeigt, dass die Teilnehmer der Inter-Föderation eduGAIN sich noch nicht gegenseitig vertrauen. Viele IdPs versenden nicht die angeforderten Attribute, so dass die Nutzer als `anonymous` nicht den vollen Umfang der Dienste nutzen können. Sprachdateien mit Kindern genießen besonderen Schutz und können daher nur von authentifizierten und autorisierten Nutzern angesehen werden. Daher ist der Zugang für viele Wissenschaftler momentan nicht gegeben.

Basismodell und Dienstsicht

Im Gegensatz zum Basismodell für eduGAIN ist die Anzahl der SP-Federation durch CLARIN festgesetzt (vgl. 2.18):

- Jeder CLARIN SP hat Stand Januar 2016 *11 Föderationen*, die jeweils ihr eigenes

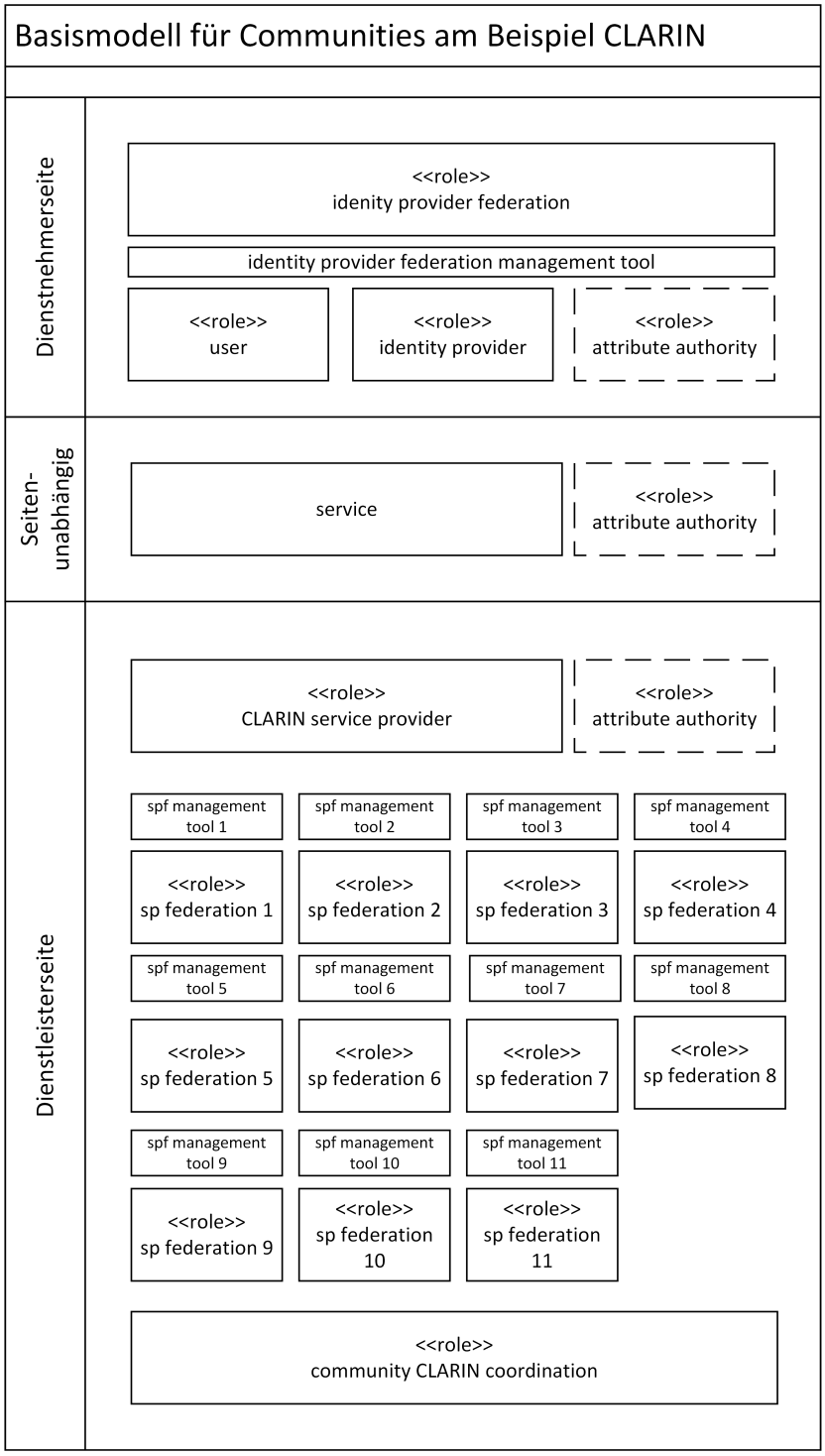

Abbildung 2.18.: Basismodell der Community CLARIN

Management Tool betreiben und somit einen anderen Zugriff zu Metadaten und der Verwaltung erlauben.

- Zusätzlich befindet sich auf Dienstleisterseite *CLARIN ERIC* als Community Organisation.

In der Dienstsicht werden diese Änderungen durch die folgenden Abhängigkeiten sichtbar (vgl. Abbildung 2.19):

- Jeder CLARIN SP schließt einen *Vertrag* mit CLARIN ERIC.
- CLARIN ERIC schließt im Gegenzug *Verträge* mit den 11 Föderationen ab.
- Vereinfacht dargestellt muss jeder CLARIN SP sich an die Richtlinien der Föderationen halten und wird durch diese kontrolliert.

Defizite

Folglich gibt es vier signifikante Defizite in der Einbindung von CLARIN in die europäische Inter-Föderation eduGAIN:

- Das Vertrauen zwischen CLARIN-SP und IdPs muss erst manuell aufgebaut werden, da ansonsten nicht die benötigten Attribute gesendet und die Dienste nicht genutzt werden können. CLARIN-SPs akzeptieren den Code of Conduct, jedoch bekommen trotzdem viele Dienste nicht die geforderten Attribute von den Heimatorganisationen der Nutzer. Je nach Sichtweise des IdPs muss entweder der Nutzer selbst oder der Service Provider diesen Prozess anstoßen, was zu Wartezeiten für den Nutzer führt, in denen er den Dienst nicht entsprechend nutzen kann.
- Parallel dazu haben Identity Provider inzwischen (Stand Januar 2016) die Möglichkeit alle CLARIN-SPs durch eine Entity Category auf einmal zu akzeptieren. Dies war vorher nicht gegeben, wodurch für IdP-Administratoren ein erheblicher manueller Aufwand für die Konfiguration entstand. Diese Entity Category muss jedoch noch Akzeptanz finden.
- Auf Grund der mangelnden Reichweite von eduGAIN wurde auf die Lösung als Service Provider Federation zurückgegriffen, die sich mehreren nationalen Föderationen anschließt. Gleichzeitig skaliert die Option von Opt-In schlecht, da jede Entität sich explizit für die Teilnahme an der Inter-Föderation eduGAIN entschließen muss.
- Nachdem nicht alle Institute in Föderationen liegen, in denen CLARIN teilnimmt, haben nicht alle potentiellen Nutzer die Möglichkeit auf die Ressourcen von CLARIN zuzugreifen. Parallel dazu existieren Forscher, deren Arbeitgeber nicht in einer Föderation teilnimmt oder die keinem Institut zugeordnet sind. Daher fungiert der CLARIN

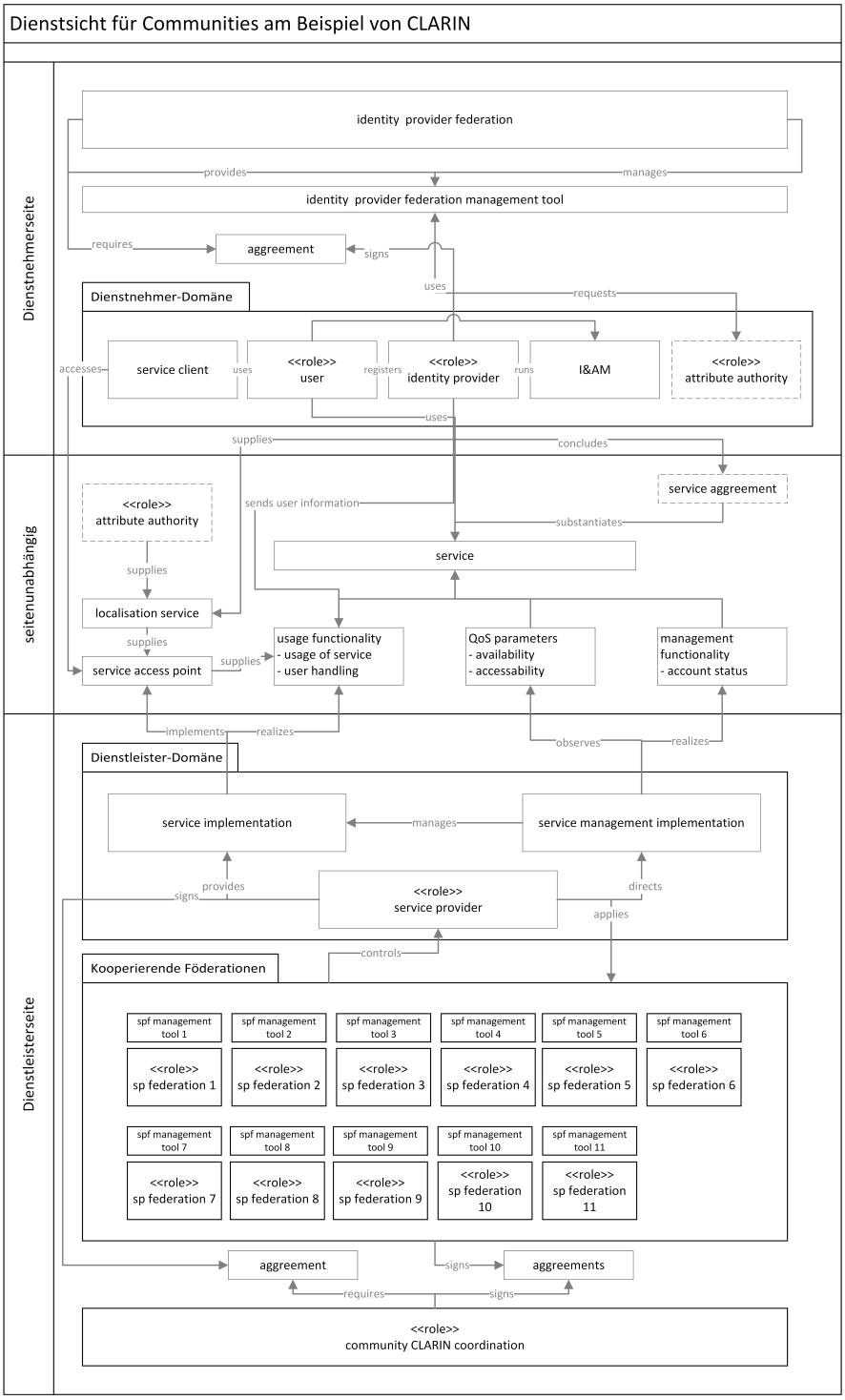

Abbildung 2.19.: Dienstsicht der Community CLARIN

Homeless IdP als SAML IdP für die Service Provider, anstatt ein privates Nutzerkonto zuzulassen oder die entsprechende Organisation in die CLARIN-Föderation aufzunehmen. Dies zeigt, dass die Lösung keine direkte Möglichkeit bietet OpenID Connect für private Benutzerkonten einzubinden.

Ein allgemeines Problem in Föderation ist die Aktualität der Daten sowie die fehlende Möglichkeit Änderungen zu bemerken. Im Folgenden wird beschrieben, welche Ziele durch den Einsatz einer geeigneten Architektur erreicht werden können.

Ziele

Nicht alle Nutzer sind Teil dieser Föderationen, wodurch eine Schnittstelle zu einem Homeless IdP benötigt wird. Eine effektive Option ist die Einbindung aller benötigten Identity Provider. Um die Probleme beim Aufbau von Vertrauensbeziehungen zu lösen, wird durch eine passende Architektur eine maximale Reichweite mit einer Vereinfachung für die beteiligten Entitäten angestrebt. Dadurch würden sich folgende Verbesserungen ergeben:

- Durch das automatische, dynamische Aufbauen von Vertrauensbeziehungen, u. a. durch den Austausch von Metadaten, erreicht die virtuelle Föderation von CLARIN alle beteiligten Service Provider und Identity Provider. Dadurch müssen die SPs nicht in mehreren Föderationen teilnehmen und alle Nutzer mit einer Heimatorganisation haben die Möglichkeit CLARIN-Dienste zu benutzen.

- Gleichzeitig skaliert eine virtuelle Föderation besser und verringert die Anzahl der benötigten Verträge zwischen IdPs, SPs, Föderationen und anderen Verbänden.

- Auf Grund der Automatisierung der Konfiguration wird der Aufwand auf Seiten von SPs und IdPs erheblich minimiert und die Wartezeit für die Nutzer reduziert. Folglich ermöglicht dies Nutzern einen Dienst deutlich früher zu nutzen, im Idealfall direkt nach der Auswahl des IdPs.

- Nutzer stoßen automatisch den Aufbau von Vertrauensbeziehungen an, können daraufhin die CLARIN-Dienste in vollem Umfang nutzen und haben nicht den Aufwand die Vertrauensbeziehung zwischen ihrem Identity Provider und einem CLARIN-SP manuell initiieren zu müssen.

- Über den bereitgestellten dynamischen Abruf von Metadaten sind die Daten immer aktuell. Durch Werkzeuge werden Änderungen direkt an die betreffenden Entitäten propagiert und aktualisierte Daten unverzüglich verteilt.

- Über die bereitgestellten Werkzeuge und die Einbindung des Code of Conduct werden die europäische Datenschutzrichtlinie und die Schutzbestimmungen der Sprachdateien überprüft und eingehalten.

- Die Teilnahme an virtuellen Föderationen soll, im Gegensatz zu eduGAIN, in akzeptabler Zeit möglich, wodurch eine wichtige Anforderung von CLARIN erfüllt wird.

- Virtuelle Föderationen sollen unterschiedliche Protokolle, wie OpenID Connect, und verschiedene Implementierungen, z. B. Shibboleth, akzeptieren.

Die Einbindung einer virtuellen Föderation in den Forschungsverbund CLARIN hätte die Besonderheit, dass das bereits etablierte System einer Service Provider Federation durch eine virtuelle Föderation abgelöst werden würde, an denen alle SPs von CLARIN, aber auch die IdPs der Nutzer teilnehmen würden.

2.4.3. Szenario 3: Grid im europäischen Umfeld

Das Leibniz-Rechenzentrum fungiert u. a. als Hochleistungsrechenzentrum, indem es Clustersysteme und Supercomputer zur Verfügung stellt, die nach einem entsprechenden Genehmigungsverfahren genutzt werden können. Gleichzeitig können für europäische Grid-Projekte Ressourcen reserviert werden. Dieses Szenario betrachtet die Integration des LRZs in internationale *Grid Computing* Projekte aus Sicht des Federated Identity Managements.

Ausgangssituation

Im Grid Computing existiert ein verteiltes dezentrales System, bei dem keine gemeinsame Administration der Ressourcen stattfindet. Jede Organisation hat eigene Policies, Batch-Systeme, Hardware und Benutzerverwaltung. Jedoch existiert eine gemeinsame Benutzeradministration in der VO. Allgemein ermöglichen *Virtuelle Organisationen* laut Michael Schiffers [Sch07] Gruppen von Organisationen oder bzw. und Individuen die kontrollierte, gemeinsame Verwendung von Ressourcen und Diensten für eine kooperative Zusammenarbeit, wobei sich die Organisationsdynamik von üblichen Organisationen unterscheidet. Die interorganisationale Gestaltung bildet ein kooperatives, flexibles Netzwerk rechtlich selbständiger Organisationen, die sich, basierend auf der Morphologie von Michael Schiffers, wie folgt charakterisieren lassen (vgl. Abbildung 2.20):

- Die Kooperation erfolgt meist über eine zentrale Organisation, der alle Teilnehmer vertrauen.

- Virtuelle Organisationen sind international.

- Die organisatorische Dimension bezüglich FIM entspricht einer Föderation oder einer Inter-Föderation.

- Die Zusammenarbeit erfolgt zur gemeinsamen Verwendung von Ressourcen und Diensten innerhalb der VO.

Abbildung 2.20.: Klassifikation der VOs im Grid-Umfeld

- Die Koordination ist meist explizit geführt.
- Der Circle of Trust ist virtuell.
- Die Bindungsintensität geschieht über einen Vertrag.
- Das Vertrauen zwischen zwei Organisationen kann sowohl direkt, beispielsweise bei Kooperationen, als auch indirekt sein.

In Europa werden im Grid Computing drei unterschiedliche Tools eingesetzt; eines davon ist die Globus Software. Europaweit ist European Globus Community Forum (EGCF) ein

Zusammenschluss der europäischen Nutzergemeinde, d. h. Entwickler, Administratoren und Benutzer, der Globus Software, um die europäischen Interessen zu bündeln, zu informieren, Treffen zu koordinieren und Projekte zu verbinden. Das LRZ ist das Kompetenzzentrum für Globus im deutschlandweiten Grid Computing und leistet somit auch Support. Das Globus Toolkit (GT) 5 bietet diverse Tools für verschiedene Aufgaben. Es ist eine OpenSource Software, die weltweit entwickelt wird und beispielsweise folgende Funktionalitäten umfasst:

- Job Submission durch das Tool Globus Resource Allocation Manager,
- Datentransfer mit Globus GridFTP und
- Data Movement Service anhand von Globus Online.

Im Folgenden wird Globus Online näher betrachtet, welches Datentransfer für Grid über die Kommandozeile, aber auch über ein Webinterface anbietet und somit eine Schnittstelle zu den Webanwendungen darstellt, die FIM einsetzen. Allgemein erzeugt der Nutzer bei der Authentifizierung einen Proxy, der in seinem Namen und mit seinen Rechten handelt [Rei08]. Die Authentifizierung geschieht hier über eine PKI Infrastruktur mit X.509 Zertifikaten und Proxies. Ein *grid-mapfile* dient zur Autorisierung, indem mittels der Textdatei der `Distinguished Name (DN)`, beispielsweise `/C=DE/O=GridGermany/OU=LeibnizRechenzentrum/CN=Daniela Poehn di34koj`, des Zertifikats auf eine Kennung der lokalen Ressource abgebildet wird. Unterschiedliche Systeme können durch die dezentrale Struktur für den gleichen `DN` unterschiedliche Benutzerkonten vergeben. Jeder Nutzer benötigt dadurch ein Zertifikat, welches von einer Certificate Authority (CA), beispielsweise dem DFN-Verein, oder einer untergeordneten Registration Authority (RA), wie dem LRZ, erstellt werden muss. Kurzlebige Zertifikate können direkt beim DFN-Verein über eine Webanwendung angefordert werden. Für langfristige Zertifikate muss sich der Antragsteller zudem ausweisen. Das LRZ verknüpft anschließend über das vorhandene ID-Portal, einem Selfservice Portal für das Identity Management am LRZ, den `DN` mit dem entsprechenden Benutzerkonto. Zudem muss der Nutzer sein Zertifikat in den Browser importieren. Für Globus müssen Zertifikat (`usercert.pem`) und Schlüssel (`userkey.pem`) im lokalen Globus-Verzeichnis hinterlegt werden.

Das oben beschriebene Konzept kann auch zur Delegation von Rechten in entfernen Domänen oder Diensten angewandt werden, vgl. Helmut Reiser [Rei08]. Der Nutzer erzeugt für Proxy Credentials lokal ein Schlüsselpaar, mit dem er wiederum ein Proxy Zertifikat mit beschränkter Gültigkeit erzeugt. Das Proxy Zertifikat signiert der Nutzer mit seinem privaten Schlüssel und überträgt es, zusammen mit einem temporären privaten Schlüssel, zum Proxy in der Gastdomäne. Aufgrund der potentiell unsicheren Umgebung ist die Gültigkeit dieser Zertifikate sehr kurz. Mit diesem Prinzip lassen sich Ressource-Proxies erzeugen, die beispielsweise im Auftrag des Nutzers weitere Dienste aufrufen. Falls der Nutzer kein Grid-Zertifikat nach der oben beschriebenen Vorgehensweise erhält, kann das Tool LRZ MyProxy ein Zertifikat ausstellen, welches nur für LRZ-Ressourcen und das EGCF Testbed gilt. MyProxy ist ein Standard-Tool von Globus zum Delegieren von Rechten und zusätzlich zum Verwalten von Zertifikaten über ein Online Repository, in das Nutzer ihre Zertifikate ablegen

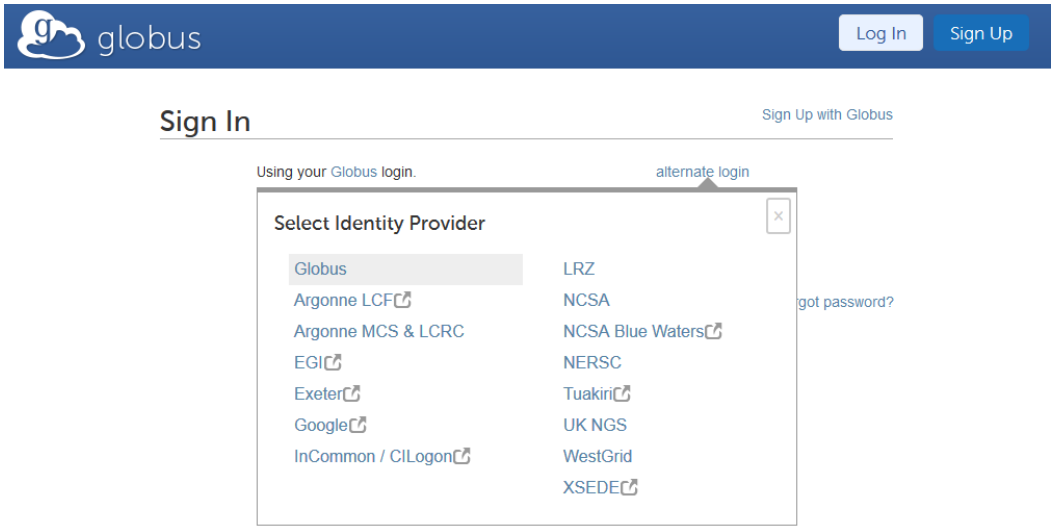

Abbildung 2.21.: Auswahl des IdPs bei Globus Online

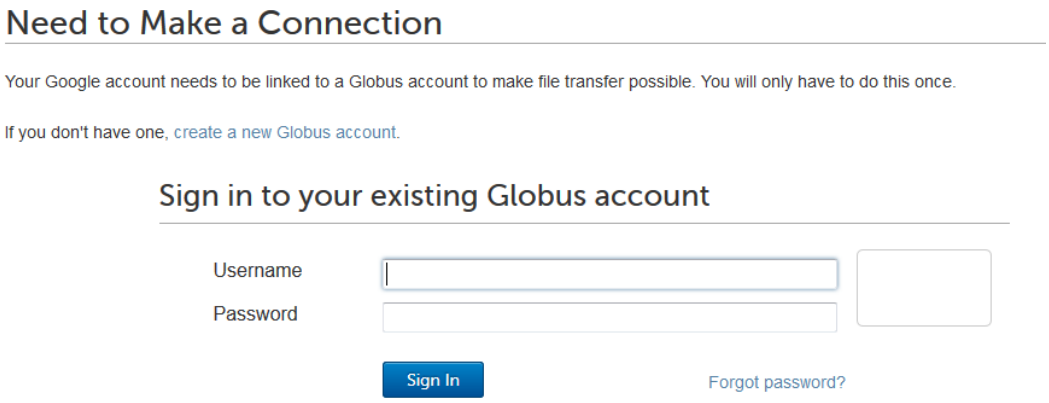

Abbildung 2.22.: Verknüpfung des Google Benutzerkontos mit dem Globus Online Benutzerkonto

können. Geschützt werden die Zertifikate durch ein selbst gewähltes MyProxy Passwort.

Workflows

Da im Grid Computing keine Föderationen, wie in Abschnitt 2.2, existieren, sind Mechanismen der Inter-Föderation eduGAIN nicht einsetzbar. Dafür ermöglichen verschiedene virtuelle Organisationen u. a. die Authentifizierung und Autorisierung innerhalb ihrer Gruppierung. Folglich unterscheidet sich der Workflow im Grid Computing stark vom Workflow in FIM und Inter-FIM.

1. Schritt: Der Nutzer erstellt ein Benutzerkonto bei Globus Online.

2. Schritt: Der Nutzer meldet sich bei Globus Online an. Dazu hat der Benutzer vom LRZ die Möglichkeit sein European Grid Infrastructure (EGI)-Benutzerkonto, sein LRZ MyProxy-Benutzerkonto oder sein privates Benutzerkonto bei Google aus einer Liste von IdPs zu verwenden, wie in Abbildung 2.21 zu sehen. Wählt der LRZ-Mitarbeiter das Leibniz-Rechenzentrum aus, so wird, wie aus Firefox Plugin SAML Tracer ausgelesen, über Hypertext Transfer Protocol (HTTP) POST eine Anfrage an LRZ MyProxy geschickt, um eine Authentifizierungsbestätigung zu erhalten (vgl. Listing 2.2).

```
1  POST https://www.globus.org/service/graph/authenticate_myproxy HTTP/1.1
2  Host: www.globus.org
3  Accept: application/json, text/javascript, */*; q=0.01
4  ...
5  Authorization: Globus-Goauthtoken null
6  Content-Type: application/json; charset=UTF-8
7  X-Requested-With: XMLHttpRequest
8  Referer: https://www.globus.org/SignIn
9  Content-Length: 72
10 Cookie: AWSELB=5
       B393DA91098F42A3300A299C6CB223940E92B238CE6971655035B54E15FB44385947E5
        4E42A3C99DD5FC7EBB1B8B9289A23E10B9510EFCE88AB1B7418D2C5C2FF0AD8A6ED;
        has_js=1; globus_auth_default=%7B%22provider%22%3A%22myproxy.lrz.de
       %22%7D
```

Listing 2.2: Anfrage an den LRZ MyProxy zur Authentifizierung

Als Parameter wird über Hypertext Transfer Protocol Secure (HTTPS) die LRZ Secure Identity Management (SIM) Kennung, bestehend aus Benutzername und Passwort, sowie die Server-Uniform Resource Locator (URL) übergeben (vgl. Listing 2.3).

```
1 POST
2 {"username":"di34koj","password":"DPIhU!2014","server":"myproxy.lrz.de"};
```

Listing 2.3: Parameter der Anfrage an den LRZ MyProxy

Die Authentifizierung geschieht bei EGI und Google ebenfalls über OAuth2. Für EGI benötigt der Proxy zudem die VO Daten von Virtual Organisation Management System (VOMS), einer Datenbank im Grid Computing, welche die Rollen und Berechtigungen

speichert.

3. Schritt: Der Nutzer muss sein Globus Online Benutzerkonto mit dem Benutzerkonto verbinden, mit dem er sich angemeldet hat, wie in Abbildung 2.22 dargestellt.

4. Schritt: Der Nutzer wählt den gewünschten Endpunkt aus.

5. Schritt: Der Benutzer authentifiziert sich bei MyProxy, welches in einem Popup-Fenster erscheint. Als MyProxy Server muss `myproxy.lrz.de` ausgewählt werden. Benutzername und Passwort entsprechen denen, die zum Hochladen des Proxys verwendet wurden.

6. Schritt: Der Nutzer initiiert die Transferanfrage.

7. Schritt: Globus Online verschiebt die Dateien.

8. Schritt: Globus Online benachrichtigt den Benutzer über den erfolgreichen Transfer.

Datenschutz und Trust

Alle Nutzer und Ressourcen haben im Grid Computing einen eindeutigen Namen. International sorgt International Grid Trust Federation (IGTF) für Vertrauen zwischen den teilnehmenden Organisationen, die Mindestanforderungen erfüllen müssen, um IGTF beitreten zu können. Zusätzlich werden in Globus Online unterschiedliche Level of Assurance gesetzt, je nachdem welcher Identity Provider verwendet wird. So hat Google ein niedrigeres Level als beispielsweise das Leibniz-Rechenzentrum. Gleichzeitig werden verschiedene Benutzerkonten durch den Benutzer miteinander verbunden, wodurch Globus Online über die verschiedenen Identitäten erfährt und Benutzerinformationen miteinander verknüpfen kann.

Basismodell und Dienstsicht

Während die Community von CLARIN nur SPs enthält, sind sowohl Identity Provider als auch Service Provider bei Grid Mitglieder (vgl. 2.23):

- Unabhängig von Dienstnehmer und Dienstleister befindet sich das übergeordnete Organ von Grid, hier als Grid Community Forum bezeichnet.

In der Dienstsicht für Grid werden die Verträge sichtbar, die vor der Dienstnutzung ausgehandelt werden 2.24:

- Sowohl IdP als auch SP schließen mit dem übergeordneten Organ einen *Vertrag* ab.
- Meist gibt es zudem noch einen *Vertrag* zwischen Identity Provider und Service Provider.

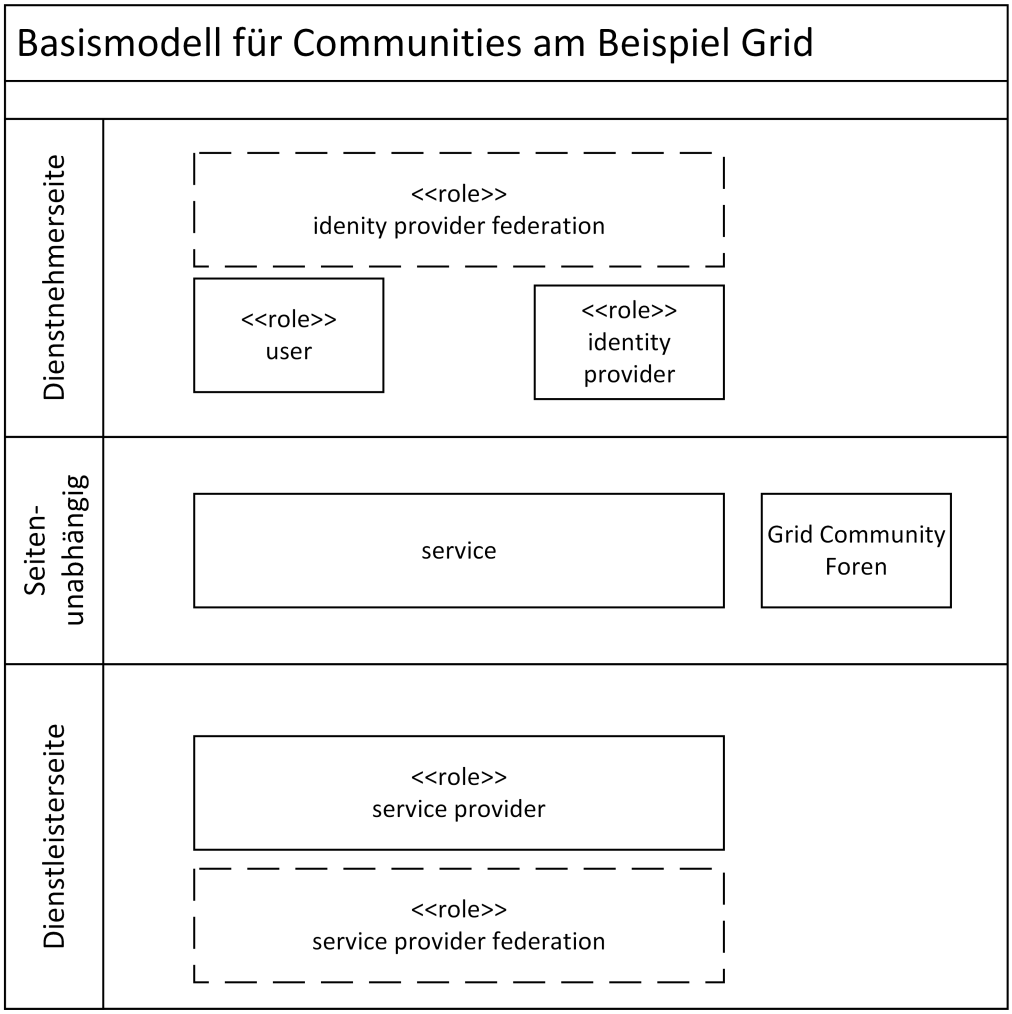

Abbildung 2.23.: Basismodell der Community Grid

- Nationale Föderationen und Attribute Authorities sind im Grid-Umfeld unbedeutend und daher nicht in der Abbildung vorhanden.

Defizite

Aus der Anwendung Globus Online lassen sich folgende Defizite ableiten:

- Das LRZ ist die einzige Heimatorganisation, mit der eine Authentifizierung und Autorisierung bei Globus Online möglich ist. Folglich müssen Mitarbeiter anderer Heimatorganisationen neben dem Benutzerkonto ihrer Heimatorganisationen weitere Benutzerkonten besitzen, was gleichzeitig zu mehreren Passwörtern führt.

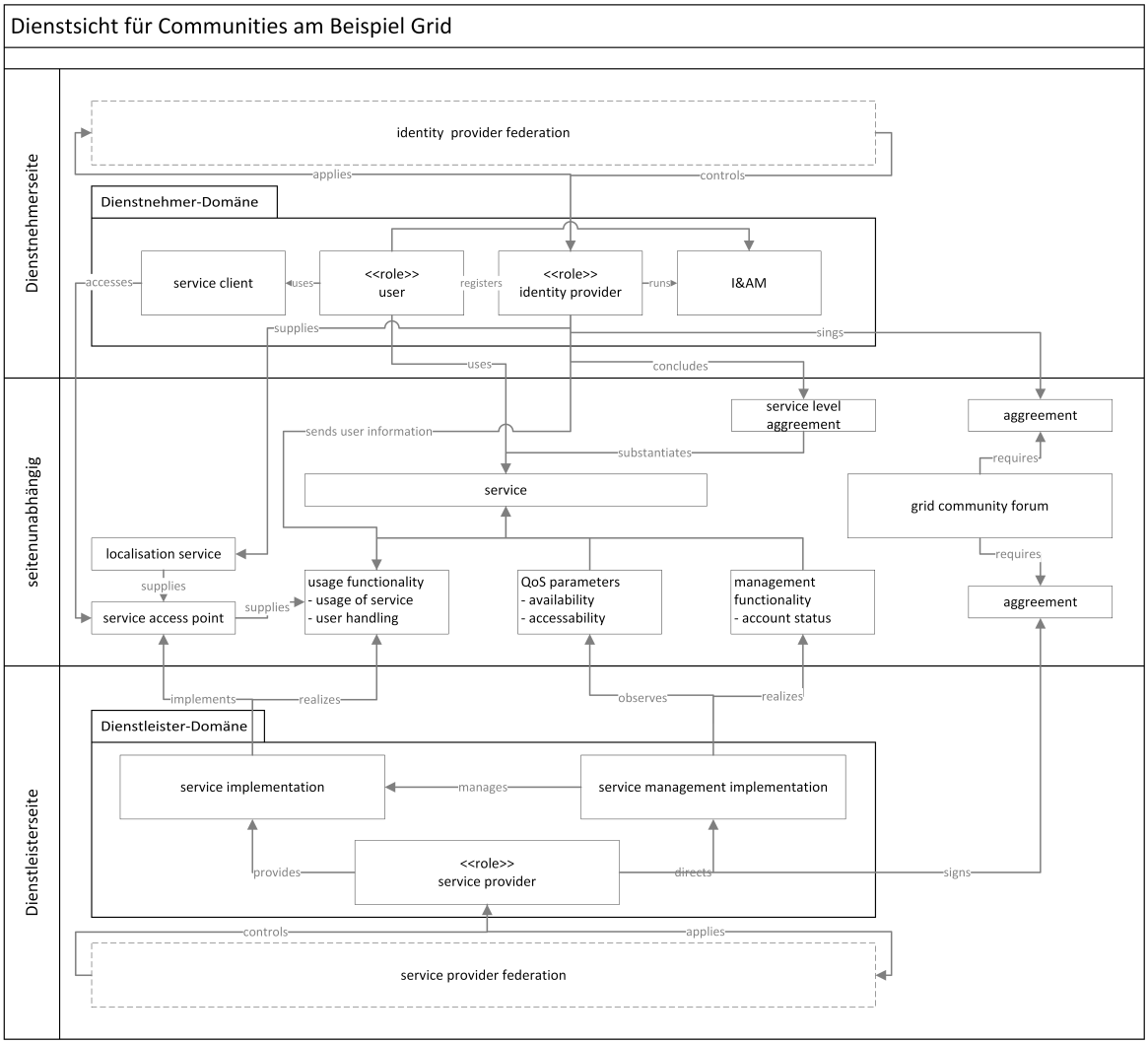

Abbildung 2.24.: Dienstsicht der Community Grid

- Account linking ist bei der ersten Verwendung einer Kennung möglich, jedoch gibt es nur wenige Möglichkeiten die dadurch aggregierten Benutzerinformationen anzuschauen.

- Die interne Verwendung von LoA durch Globus Online ist für den Benutzer nicht transparent.

- Die Authentifizierung ist alleine durch OAuth2 und Zertifikaten über MyProxy möglich. SAML kann nicht proprietär verwendet werden.

- Die Benutzerfreundlichkeit ist vor allem bei der ersten Verwendung gering. Der Nutzer muss mehrere Benutzerkonten haben und deren Passwörter wissen. Zudem sind die Fehlermeldungen bei der Authentifizierung bei Globus Online und beim MyProxy Popup-Fenster wenig aussagekräftig, wodurch die Fehlersuche erschwert wird.

Ziele

Um die Defizite insbesondere bei der Authentifizierung bei Globus Online zu beheben, wird eine Anbindung an eduGAIN und an weiteren Organisationen in R&E angestrebt. Dadurch würden sich folgende Verbesserungen ergeben:

- Durch die Einbindung der Inter-Föderation eduGAIN und weiteren Föderationen können Nutzer die Benutzerkonten der Heimatorganisationen verwenden, um auf den Dienst Globus Online zugreifen zu können.

- Gleichzeitig wird die Reichweite erhöht, was dazu führen kann, dass die Mitgliederzahl steigt.

- Um dies zu erreichen, müssen Authentifizierung und Autorisierung stärker unterschieden werden. Die Authentifizierung muss somit über die Heimatorganisation möglich sein, während ein Zertifikat, MyProxy oder eine Attribute Authority im Grid Umfeld die Autorisierung der Nutzer vornimmt.

- Damit Nutzer sich über ihre Heimatorganisation anmelden können, ist die Protokollunabhängigkeit unabdingbar. Dazu müssen Proxies die Nachrichten in das gewünschte Format umwandeln oder entsprechende Implementierungen für die Unabhängigkeit sorgen.

- Die Anwendung, insbesondere der statische Lokalisierungsdienst, muss sich dynamisch an die Anzahl der teilnehmenden Organisationen anpassen und die Möglichkeit bieten neue Organisationen hinzuzufügen.

- Das Vertrauen in den Identity Provider muss sich in einem für den Nutzer und die Heimatorganisation transparenten Level of Assurance ausdrücken lassen.

- Der Nutzer muss die Möglichkeit haben seine beim Dienst hinterlegten persönlichen Informationen abzufragen und die Erlaubnis für die Verwendung bestimmter Daten zu widerrufen.

- Die Verwendung der Anwendung, insbesondere der Authentifizierung und Autorisierung, muss benutzerfreundlich sein.

2.4.4. Anforderungen

Aus den beiden Szenarien der Community CLARIN und aus dem Grid-Umfeld ergeben sich folgende Anforderungen.

Funktionale Anforderungen

Die folgenden funktionalen Anforderungen ergeben sich aus der Integration der Virtuellen Organisationen der Communities in Federated Identity Management:

- Die zu übertragenden Daten müssen semantisch zwischen Authentifizierungs- und Autorisierungsbestätigungen sowie allgemeinen Attributsauskünften unterschieden werden, um Attribute Authorities einbinden zu können [FA-Datenkategorisierung].

- Dem Service Provider muss angezeigt werden, welche Datenqualität der Identity Provider liefern kann. Zugleich muss die Einteilung in eine bestimmte Klasse transparent erfolgen [FA-LoA].

- Nutzer ohne Heimatorganisation müssen die Dienste trotzdem nutzen können. Eine entsprechende Schnittstelle zu einem Dienst für heimatlose Nutzer oder die direkte Integration eines solchen Dienstes muss gegeben sein [FA-Homeless].

- Die Integration in die lokale Umgebung der IdPs und SPs muss ohne erheblichen Aufwand geschehen [FA-Integration].

- Die Lösung muss alle benötigten Identity Provider und Service Provider umfassen, um die Parallelität von Föderationen mit schlechter administrierbaren Ad-hoc-Lösungen zu vermeiden [FA-Reichweite].

- Die Anzahl der benötigten Verträge muss reduziert bzw. dem Nutzerkreis angepasst werden, damit keine unnötigen oder redundanten Verträge geschlossen werden [FA-SLA].

- Die Architektur der Föderationen muss sich dynamisch an geänderte Anforderungen anpassen können, um eine FIM-Architektur für Projekte zu ermöglichen, die nur für eine bestimmte Dauer bestehen [FA-Dynamik].

- Der Nutzer muss die Möglichkeit haben seine Heimatorganisation bei einem Lokalisierungsdienst auszuwählen [FA-Lokalisierung].

- Die Anforderung [FA-Interaktion] wird um die Möglichkeit erweitert die persönlichen hinterlegten Daten beim Service Provider abzufragen und gegebene Erlaubnisse zu widerrufen.

Nichtfunktionale technische Anforderungen

Die folgenden Anforderungen beschreiben Anforderungen an das zu entwickelnde System, welche technische Aspekte betreffen, die jedoch nicht direkt mit der Funktionalität zusammen hängen.

- Das System muss unterschiedliche Protokolle, z. B. SAML und OpenID Connect, akzeptieren. Dies ist insbesondere zu Schnittstellen im Grid-Umfeld (X.509 Zertifikate) und bei Einbindung von heimatlosen Nutzern (OpenID Connect) von Interesse [NFA-Protokollunabhängigkeit].

- Das System muss skalierbar für eine beliebige Anzahl an Teilnehmern und dynamischen Föderationen sein. Die Anzahl an Benutzern, Service Providern, Identity Providern und Föderationen muss sich dynamisch verändern können [NFA-Skalierbarkeit].

Sicherheitsanforderungen

Die folgenden Anforderungen zielen auf die Sicherheit und die Integration in bestehende Sicherheitsstrukturen ab:

- Benutzerinformationen dürfen nur an Service Provider weiter gegeben werden, zu denen ein Vertrauensverhältnis besteht und die genügend vertrauenswürdig sind [SEC-LoT].

- Benutzer dürfen einen Dienst nur dann nutzen, wenn ihr Identity Provider den Qualitätsbestimmungen des SPs entspricht. Somit ist die Anforderung [FA-LoA] auch in der Sicherheit notwendig.

- Die Integration muss auch aus Sicht der Sicherheit betrachtet werden. Folglich ist [FA-Integration] auch in der Sicherheit gültig.

Organisatorische Anforderungen

Die organisatorischen Anforderungen stellen ein weiteres wichtiges Element dar:

- Die Anforderung [FA-Automatisierung] hat ferner Auswirkungen auf die Organisation, die beachtet werden müssen.
- Zusätzlich gelten [SEC-LoT] und [SEC-LoA] ebenfalls als organisatorische Anforderung.
- Die Datenqualität und andere Gütemerkmale müssen, ergänzend zu [FA-SLA], passend spezifiziert werden können. Die Einhaltung der SLAs muss automatisch überprüft werden können [ORG-SLA].

Datenschutzrechtliche Anforderungen

Durch die Weitergabe und den Bezug personenbezogener Daten sind die datenschutzrechtlichen Anforderungen ebenso zu betrachten:

- Die Datenschutzrichtlinie der EU muss überprüft werden [DSA-CoCo].
- Die Datenschutzrichtlinien und Datenschutzgesetze müssen eingehalten werden können [DSA-Datenschutz].
- Die Anforderung [SEC-LoT] gilt ebenfalls für den Datenschutz.

Die genannten Anforderungen beziehen sich auf die Sicht von IdP und SP, die in Communities operieren. Die weiteren Szenarien betrachten Aspekte der Wirtschaft und die Benutzersicht.

2.5. Identity Management in der Wirtschaft

In diesem Abschnitt wird aufgezeigt, dass dynamische virtuelle Föderationen auch in der Wirtschaft benötigt werden.

2.5.1. Motivation

Durch die verstärkte Zusammenarbeit, die auch Sektor-übergreifend sein kann, kommen die bisherigen föderierten Lösungen an ihre Grenzen. Damit das in dieser Arbeit zu erstellende Konzept sich nicht nur auf R&E beschränkt, sondern universal eingesetzt werden kann, wird im Folgenden ein fiktives Szenario aus der Wirtschaft betrachtet.

2.5.2. Szenario 4: Sektorübergreifendes Identitätsmanagement mit Automobilherstellern

Während sich in R&E nationale Föderationen, wie die DFN-AAI, bildeten, wurden in der Wirtschaft Sektor-spezifische Föderationen etabliert. Ein Beispiel hierfür ist Odette SESAM für die Automobilbranche. In einem gemeinsamen, fiktiven Projekt sollen zwei Partner aus der Automobilbranche, aus dem Hochschulumfeld sowie eine Ausgründung einer Universität kooperieren.

Ausgangssituation

Die beiden Teilnehmer aus dem Hochschulumfeld sind Mitglied in der DFN-AAI und beruhen somit auf SAML sowie den Schemata `dfnEduPerson`, `eduPerson` und `SCHAC`. Die Ausgründung der Universität hat sich dahingegen für OpenID Connect entschieden. Sie ist noch kein Teilnehmer einer Föderation und hat somit ihr eigenes Namensschema. Weitere zwei Teilnehmer stammen aus der Automobilbranche, wobei andere Organisationen noch in das Projekt hinzu stoßen können. In der Automobilbranche existiert die Föderation Odette SESAM [Ode09], die 2009 durch eine Working Group spezifiziert wurde und ebenfalls auf SAML beruht. Teilnehmer dieser Föderation sind beispielsweise die Bayerische Motoren Werke (BMW) Group, Bosch Gesellschaft mit beschränkter Haftung (GmbH), Daimler Aktiengesellschaft (AG) und Zahnradfabrik (ZF) Friedrichshafen AG.

SESAM definiert bestimmte Voraussetzungen und beschreibt Empfehlungen für die Integration von IdPs und SPs. Im Folgenden werden verwendete Profile und Bindings betrachtet:

- SAML V2.0 Web Browser SSO Profile wird empfohlen,
- HTTP POST Binding wird ebenfalls empfohlen,
- HTTP Redirect Binding ist optional und
- das vom IdP initiierte Browser/POST Profile kann eingesetzt werden, wenn Assertions aus SAML V2.0 verwendet werden.

Assertions müssen ein `Subject` Kindelement enthalten, welches ein `NameId` und genau ein `SubjectConfirmation` Element mit einem Method-Attribut, gesetzt auf `urn:oasis:names:tc:SAML:2.0:cm:bearer`, enthält. Auch wenn keine weiteren Voraussetzungen explizit genannt sind, soll ein `Conditions` Element ebenfalls enthalten sein.

Ebenso wie für Bindings und Protokoll gibt es vorgeschriebene Attribute, die vorhanden sein müssen:

- `Universally Unique Identifier (UUID)`: Nach [`Request for Comments (RFC)4122`],

identifiziert durch die URN `urn:odette:sesam:uuid`.

- `Assurance Level`: Nach NIST-800-63, identifiziert durch `urn:odette:sesam:assuranceLevel`.
- `E-Mail-Adresse`: Nach [RFC822], identifiziert durch `urn:odette:sesam:mail`.

Zusätzlich werden bestimmte Attribute empfohlen:

- `Salutation`: D. h. Anrede, identifiziert durch `urn:odette:sesam:salutation`.
- `Given Name`: Identifiziert durch `urn:odette:sesam:givenName`.
- `Surname`: Identifiziert durch `urn:odette:sesam:surname`.
- `Display Name`: Identifiziert durch `urn:odette:sesam:displayName`.
- `Preferred Language`: Identifiziert durch `urn:odette:sesam:preferredLanguage`.
- `Telephone Number`: Identifiziert durch `urn:odette:sesam:telephoneNumber`.

Odette empfiehlt weiterhin die Verwendung von TLS bzw. Secure Sockets Layer (SSL), Zertifikaten und signierten Nachrichten. Dies veranschaulicht, dass die Föderationen begrenzt sind auf Sektoren. Jede Föderation hat ihr eigenes Schema und ihre eigenen Voraussetzungen, was die Zusammenarbeit verkompliziert. Nachdem die Ausgründung auf OpenID Connect setzt, ist eine Teilnahme in der DFN-AAI nicht möglich. Ebenso werden die Voraussetzungen für Odette SESAM nicht erfüllt. Folglich müssen die Projektteilnehmer eine eigene Projektföderation gründen oder die Benutzerdaten duplizieren.

Workflows

Somit ergibt sich folgender fiktiver Workflow:

1. Schritt: Die Teilnehmer stellen fest, dass sie keine der vorhandenen Föderationen nutzen können. Sie besprechen die Problematik mit ihren Administratoren, die einen manuellen Austausch empfehlen.

2. Schritt: Die IdPs und SPs aus Hochschulen und Automobilbranche tauschen folglich ihre Metadaten manuell aus.

3. Schritt: Nachdem OpenID Connect nicht unterstützt wird, wird für die Ausgründung ein Homeless IdP aufgesetzt, indem alle Benutzerinformationen der im Projekt involvierten Mitarbeiter kopiert werden.

4. Schritt: Die Metadaten des Homeless IdP werden ebenfalls manuell bei den SPs eingefügt.

5. Schritt: Die eingebetteten Lokalisierungsdienste der Service Provider werden so angepasst, dass die Teilnehmer leicht über den Projektnamen ihre IdPs finden können. Auch der Homeless IdP wird vom Namen so modifiziert, dass der Name der Ausgründung zu sehen ist.

6. Schritt: Damit die Teilnehmer die Dienste auch nutzen können, werden im nächsten Schritt die Konfigurationen so angepasst, dass die technischen Voraussetzungen als gleichwertig gelten. Hierzu muss auch ein schriftlicher Vertrag aufgesetzt werden.

7. Schritt: Damit die Benutzerinformationen verstanden werden, muss jeder IdP passende Konvertierungsregeln manuell einfügen.

7b. Schritt: Falls dies nicht ausreicht, muss eine SAML-SAML-Bridge aufgesetzt werden, die die Kommunikation normalisiert und für die jeweiligen Partner verständlich macht, falls zu unterschiedliche Bindings, Profiles und weitere Spezialisierungen verwendet werden.

8. Schritt: Nun können die Nutzer benötigte Dienste nutzen.

Dies zeigt, dass viele manuelle Schritte durch die unterschiedlichen technischen Voraussetzungen notwendig sind.

Basismodell und Dienstsicht

Das Basismodell von Wirtschaftskooperationen ist ähnlich dem von Inter-FIM, wie am Beispiel von DFN-AAI und Odette SESAM dargestellt (vgl. Abbildung 2.25):

- Dienstnehmerseite und Dienstleisterseite sind äquivalent zu denen in Inter-FIM.
- Die dienstunabhängige Seite enthält neben dem Dienst ein *homeless IdP*, der es ermöglicht zusätzliche IdPs einzubinden, beispielsweise weil sie ein anderes Protokoll benutzen. Unter Umständen ist selbst für dasselbe Protokoll eine Art Bridge notwendig, um die unterschiedlichen Profiles, Bindings und sonstige Besonderheiten auszugleichen.

Die Sektoren sind weder im Basismodell noch in der Dienstsicht gut darstellbar. In der Dienstsicht zeigen sich die Sektoren insbesondere durch den manuellen Austausch und die Verträge, wie in Abbildung 2.26 zu sehen. Gegebenenfalls wird zudem eine Bridge notwendig, wie eben beschrieben. Dies hat Auswirkungen auf die Dienstsicht. Die Unterschiede zu Inter-FIM sind die Folgenden:

- *identity provider* und *service provider* unterschreiben ein *agreement*.

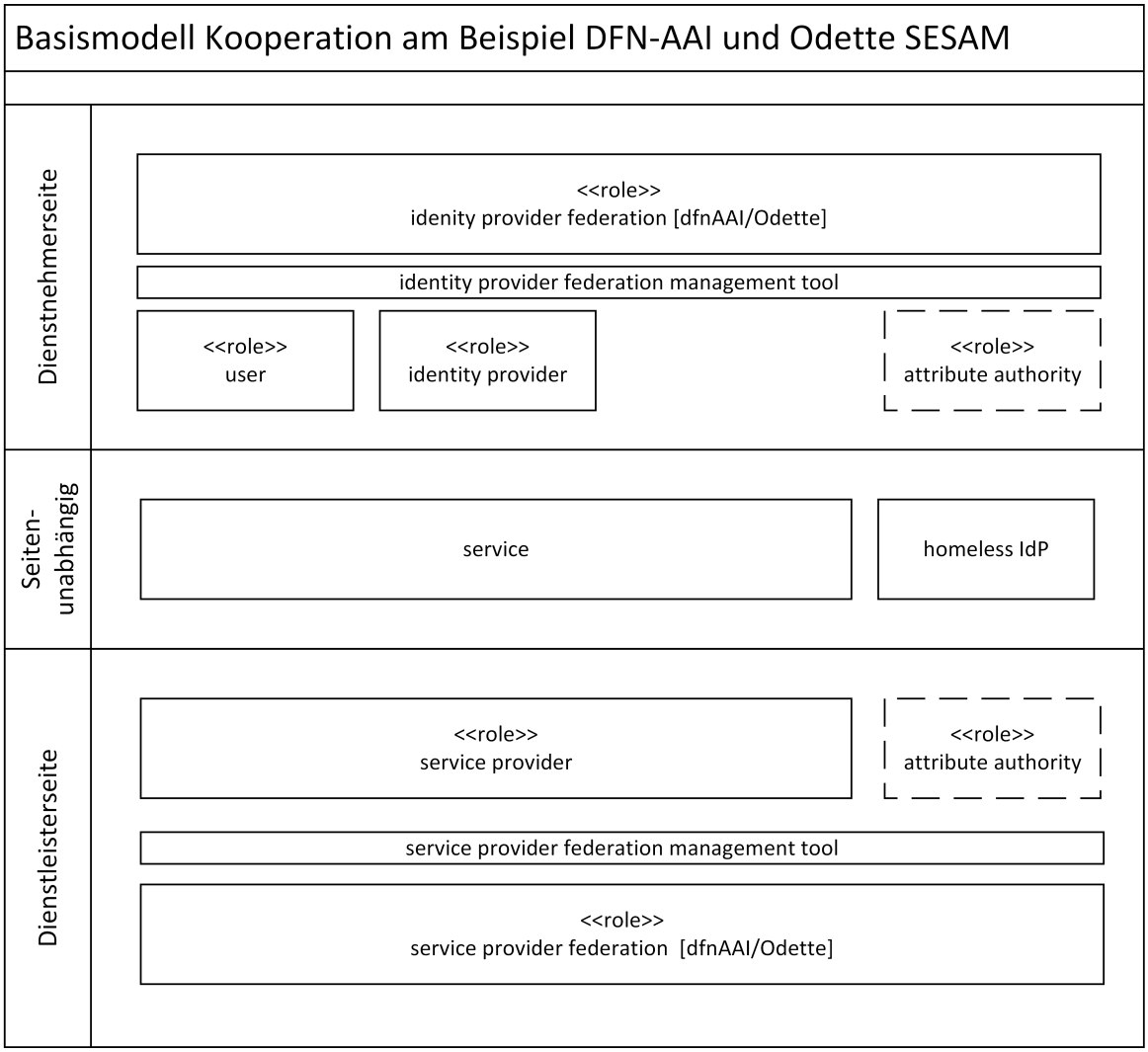

Abbildung 2.25.: Basismodell für Odette SESAM

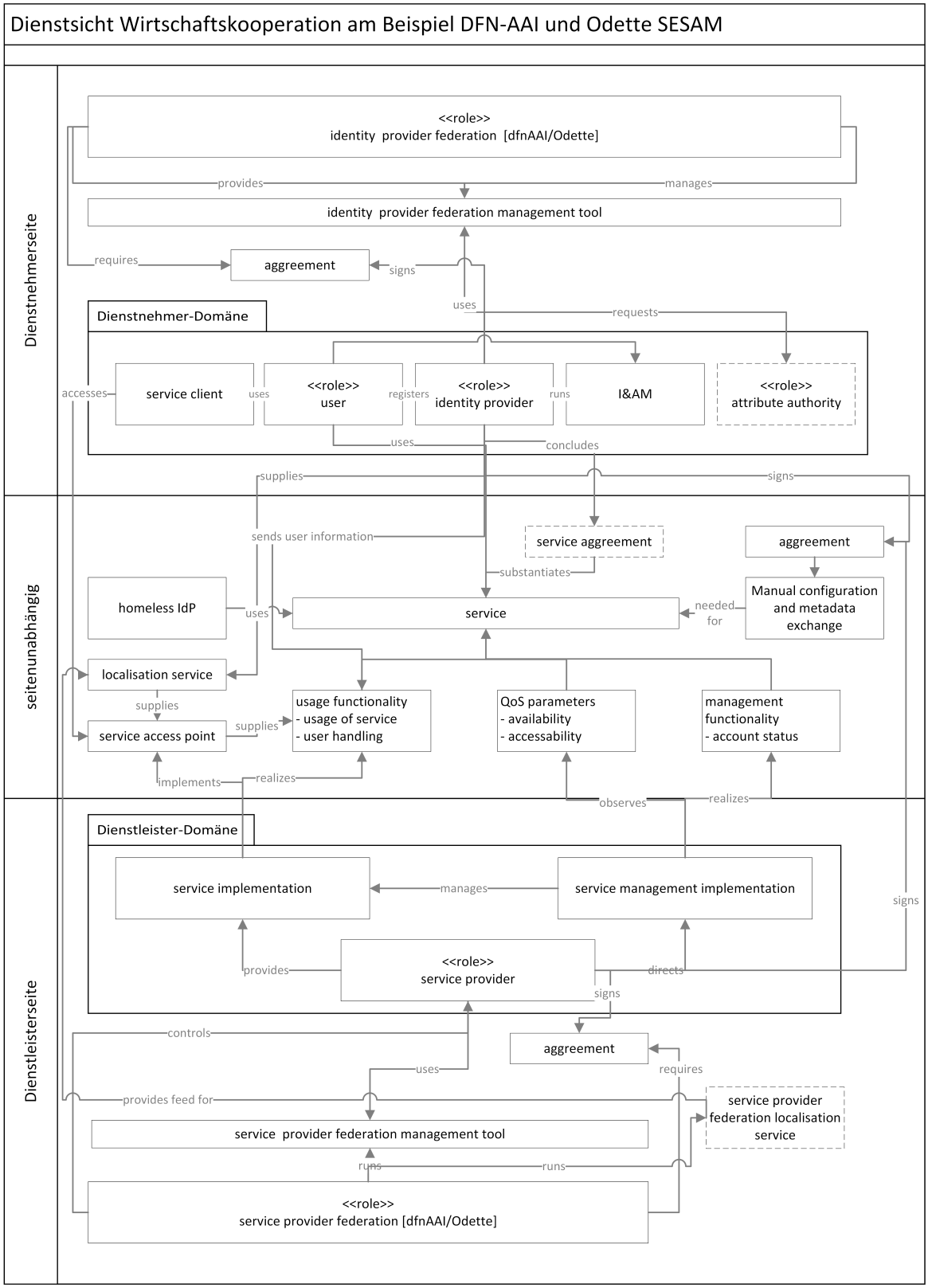

Abbildung 2.26.: Dienstsicht für Odette SESAM

- Bedingt durch das agreement werden *manuell* Metadaten ausgetauscht und die Konfiguration ausgetauscht.
- Falls dies nicht ausreicht, muss zudem mindestens eine Bridge installiert werden, der die Kommunikation normalisiert.
- Erst dann ist es möglich, einen Dienst zu nutzen.
- Für Teilnehmer aus anderen Protokollen wird ein *homeless IdP* installiert.

Datenschutz und Trust

Innerhalb der einzelnen Föderationen ist ein gewisses Vertrauen vorhanden. Odette SESAM hat strenge Aufnahmekriterien, wodurch hier ein genügend großer Schutz vorhanden ist. Die Teilnehmer der DFN-AAI vertrauen sich in einem gewissen Grad, der stark von dem einzelnen Teilnehmer abhängt. Beispielsweise verlangt die TUM für jeden Dienst einen schriftlichen Vertrag, während LRZ und LMU diesbezüglich mehr Vertrauen in die Föderation setzen. Föderationsübergreifend ist hier erst einmal kein Vertrauen vorhanden. Technisch wird das durch den manuellen Metadatenaustausch hergestellt. Hierfür müssen die beteiligten Administratoren dem zustimmen. Datenschutz wird durch Attributfilter, Consent und dem deutschen Bundesdatenschutzgesetz eingehalten.

Defizite

Aus diesem Sektor-übergreifenden Szenario lassen sich die folgenden Defizite ableiten:

- Die Projektföderation muss manuell durch händischen Metadatenaustausch und Anpassung der Konfiguration geschehen.
- Sektor-übergreifende Föderationen benötigen mehr Anpassungen als Sektor-interne Kooperationen.
- Außerhalb der Automobilbranche fehlen teilweise Angabe zu Assurance bzw. sind nicht in der benötigten Form vorhanden (vgl. NIST zu Verlässlichkeitsklassen).
- Die aktuellen Föderationen mitsamt ihrer technischen Infrastruktur sind nicht protokollunabhängig.
- Es geschieht mehrfache Arbeit durch das Erstellen von Konvertierungsregeln durch jeden IdP.

Ziele

Aus den genannten Defiziten ergeben sich diese Ziele:

- Automatische Erstellung von Föderationen und Austausch von Metadaten.
- Sektorübergreifende Föderationen bzw. einfaches Mapping durch einen zentralen Dienst.
- Assurance Mapping bzw. ein generisches Trust Management, welches möglichst viele Verlässlichkeitsklassen einschließt.
- Protokollübergreifende Föderationen.
- Einfache Konvertierung von Benutzerinformationen.
- Um den Anforderungen von Odette SESAM einzuhalten, sollen diese Föderationen möglichst sicher sein.
- Bei einem zentralen Managementtool zur Registrierung von Entitäten soll es die Möglichkeit geben Föderationen zu verwalten. Diese sollen über einen definierbaren Aufnahmeprozess verfügen, um Parallellösungen zu vermeiden.

2.5.3. Anforderungen

Aus dem fiktiven Sektor-übergreifenden Szenario ergeben sich die nachfolgenden Anforderungen.

Funktionale Anforderungen

Es lassen sich die folgenden zusätzlichen funktionalen Anforderungen ableiten bzw. erweitern:

- Dem SP muss angezeigt werden, welche Datenqualität der IdP liefern kann. Zugleich muss die Einteilung in eine bestimmte Klasse transparent erfolgen. Diese Anforderung wird über ein Mapping bzw. ein generisches Trust Management erweitert [FA-LoA].
- Der Vertrauensaufbau soll on demand und automatisch geschehen können, um dynamisch auf geänderte Anforderungen der Nutzer reagieren zu können. Dies soll auch Sektor-übergreifend möglich sein. Somit wird diese Anforderung konkretisiert [FA-Automatisierung].
- Die Bildung und Verwaltung mehrerer Föderationen muss unterstützt werden. Diese Föderationen sollen Länder- und Sektor-unabhängig sein. Folglich wird diese Anforde-

rung ebenfalls erweitert [FA-Föderation].

- Die Benutzerinformationen müssen über nationale Grenzen hinweg versendet werden können. Als Grenze gelten auch Föderationen, wodurch diese Anforderung erweitert wird [FA-Grenzüberschreitend].

Nichtfunktionale technische Anforderungen

Die Anforderung [NFA-Koexistenz] gilt auch für Föderationen, die nicht aus einem Sektor stammen.

Sicherheitsanforderungen

Die folgenden beiden Anforderungen werden durch das Szenario aus der Wirtschaft erweitert:

- Benutzer dürfen einen Dienst nur dann nutzen, wenn ihr IdP den Qualitätsbestimmungen des SPs entspricht. Um dies zu überprüfen soll ein generisches Schema oder ein Mapping möglich sein [SEC-LoA].

- Die Sicherheit jeder Komponente sowie der gesamten Föderationen muss betrachtet werden. Um [SEC-Multilateral] für einzelne Föderationen zu gewährleisten, sollen Föderationsverwaltungen Anforderungen stellen können.

Organisatorische Anforderungen

Die folgenden organisatorischen Anforderungen werden durch das Szenario aufgestellt:

- Die Organisationen müssen sich in virtuellen Föderationen registrieren können [ORG-Registrierung].

- Die Verlässlichkeitsklasse betrifft interne Abläufe und die Konfiguration. Diese sollen durch Hilfsmittel, wie beispielsweise durch ein zentrales Tool, erleichtert werden [ORG-LoA].

- Die Lösung soll Schnittstellen zu den organisationsinternen Supportprozessen, wie dem Service Desk und dem Change Management aufweisen. Damit Föderationen diesbezüglich auch unterstützt werden, soll die Anforderung [ORG-Supportprozesse] auch für organisationsübergreifende Supportprozesse gelten.

2.6. User Centric Identity Management

In diesem Abschnitt wird auf UCIM eingegangen. Diese auf den Nutzer zugeschnittenen Konzepte beruhen auf den Schutzbedürfnissen der Benutzer, die sich in vier Kategorien einteilen lassen können [BGS05]:

Not concerned: Personen, die keinen Wert auf den Umgang mit ihren Daten legen.

Identity concerned: Personen, die zwar zur Personalisierung Informationen an Dienste übermitteln, jedoch versuchen keine Informationen weiterzugeben, die auf eine reale, identifizierbare Person schließen lassen.

Profile averse: Personen, die unter ihrem Realnamen agieren, jedoch keine profilbezogenen Daten herausgeben.

Privacy Fundamentalists: Personen, die möglichst wenige Informationen über sich preisgeben wollen und vorsichtshalber auf die Nutzung entsprechender Dienste verzichten.

Auf Grund von Defiziten in FIM-Lösungen, insbesondere bei der Funktionalität für Benutzer, wurde UCIM entwickelt und zuletzt im Bereich des UMA wieder aufgegriffen.

2.6.1. Motivation

Zur Verwaltung der vielen digitalen Identitäten kamen zunächst Passwort-Management-Tools auf den Markt, mit denen Benutzer ihre Passwörter lokal speichern können. Diese Funktionalität wurde zunehmend in Webbrowser integriert. Unabhängig davon wurde Privacy Enhancing Technology (PET) eingeführt, durch welche Benutzer ihre Daten selbst verwalten können. Eine Weiterentwicklung davon stellt UMA dar, welches im folgenden Abschnitt näher erläutert wird. Benutzer sind zunehmend im privaten Umfeld mit Entscheidungen über die Weitergabe ihrer personenbezogenen Daten im Web 2.0 durch die Fokussierung auf den Nutzer [GHN09] konfrontiert. Soziale Netzwerke, aber auch bei der Installation von Apps auf Smartphones werden Nutzer nach ihrer Zustimmung gefragt. Dabei sehen sie, welche Berechtigungen welche Anwendungen benötigen und können anschließend pro Anwendung entscheiden, ob sie dies akzeptieren oder nicht. Gleichzeitig fehlt die globale Sicht auf die Verteilung ihrer persönlichen Daten.

2.6.2. Aktuelle Entwicklungen

User-Managed Access [Kan15] ist ein von der Kantara-Initative entwickeltes, OAuth-basierendes Protokoll, welches dem Benutzer einen zentralen, einheitlichen Kontrollpunkt zur Autorisierung bereit stellt, über den er selbst bestimmen kann, wer wann Zugriff auf die

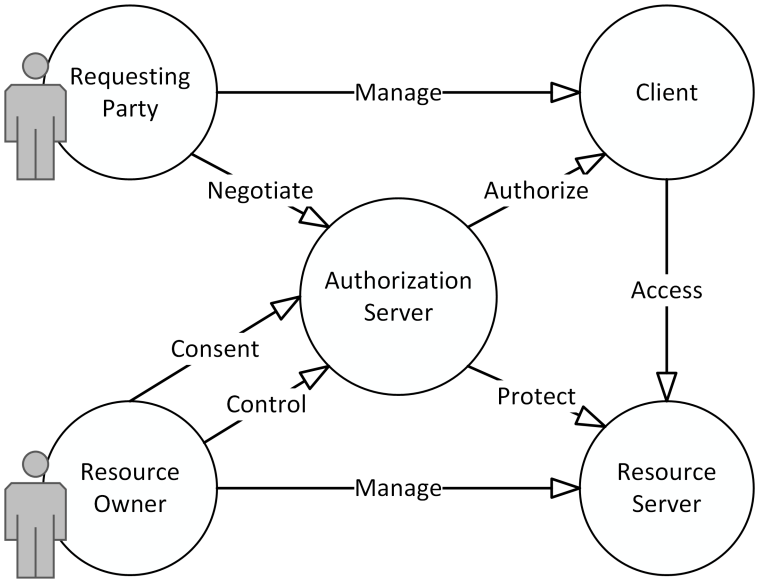

Abbildung 2.27.: UMA – Rollen und Funktionalitäten [Kan15]

persönlichen Daten, Inhalte und Dienste bekommt. Die Architektur von UMA sieht, wie in Abbildung 2.27 dargestellt, folgende Rollen vor:

- Die *Requesting Party* startet den UMA-Workflow.
- Die Requesting Party bedient einen *Client*, der Zugriff auf einen Dienst haben möchte.
- Dieser Dienst ist beim *Resource Server* zu finden, der von einem *Resource Owner*, d. h. Benutzer, betrieben wird.
- Der Resource Owner, d. h. der Nutzer, kontrolliert den *Authorization Server*, der die Autorisierung für den Dienst übernimmt. Der Authorization Server kommuniziert mit der Requesting Party über den Client, damit dieser autorisiert wird.

Dadurch können unterschiedliche Funktionalitäten geboten werden:

- Dedizierter Dienst zur Autorisierung von Zugriffen (*Access*) vom Client auf persönliche Daten und für den Datenaustausch (*Resource Server*).
- Schnittstelle von der autorativen Quelle zu unterschiedlichen Anwendungen, wodurch die manuelle Eingabe von persönlichen Daten entfallen soll.
- Überblick des Resource Owners über die Freigaben, sowie die Möglichkeit auf dem Authorization Server Rechte zu ändern und Zugriffsrechte zu entziehen.

Das Design von UMA benötigt grundsätzlich keine Objektidentifizierung, jedoch werden diese häufig in Access Authorization Policies, wie Access Control List (ACL), eingesetzt. Die Option wird daher aktuell (Stand Januar 2016) in OpenID Connect integriert.

2.6.3. Szenario 5: UMA

Das folgende fiktive Beispiel illustriert die Möglichkeiten von UCIM am Beispiel von UMA. Daraus werden Anforderungen abgeleitet, die auch für dynamische virtuelle Föderationen bezüglich der Selbstbestimmung des Nutzers relevant sind.

Ausgangssituation

Ein Benutzer, Alice, möchte seine persönlichen Daten in Webanwendungen besser verwalten, um ihnen zum Beispiel einmaligen Zugriff zu persönlichen Daten zu gewähren und Sets mit bestimmten Daten anzulegen. Die Daten von Alice sind auf dem Resource Server, wie in Abbildung 2.27 zu sehen, gespeichert. Um die persönlichen Daten für Webanwendungen verwenden zu können, müssen sich der Resource Server und der Authorization Server kennen. In den meisten Fällen werden sie auf einem reellen Server betrieben. Alice fand einen Provider, der Authorization Server und Resource Server in einem Dienst betreibt und dem sie vertraut. Dort hat sie ihre persönlichen Daten eingegeben, die sie freigeben möchte. Parallel dazu hat sie zwei Identity Provider eingetragen, die über eine Schnittstelle bei Bedarf angefragt werden können. Alice möchte online einen neuen Arbeitsspeicher für ihr Notebook kaufen und vertraut dem Online-Shop NewHardware4U auf Grund von guter Reputation und früheren Erfahrungen. Diese Requesting Party NewHardware4U agiert beim Einkauf im Auftrag von Alice, um ihr beim Versand des eingekauften Produktes von England nach Deutschland zu helfen.

Workflows

Im Gegensatz zu FIM-Workflows unter SAML betreibt der Nutzer einen eigenen IdP in Form eines Resource Servers oder eine Schnittstelle zu einem, beispielsweise kommerziellen, IdP. Somit hat der Benutzer die Kontrolle über den Versand von persönlichen Daten, wie im Ablaufdiagramm von Abbildung 2.28 zu sehen.

1. Schritt: Alice entscheidet sich für einen Authorization und Resource Server.

2. Schritt: Authorization Server und Resource Server werden miteinander bekannt gemacht. Beide Server sind unter der Kontrolle von Alice.

3. Schritt: Der Resource Server registriert sich beim Authorization Server. Nachdem Alice

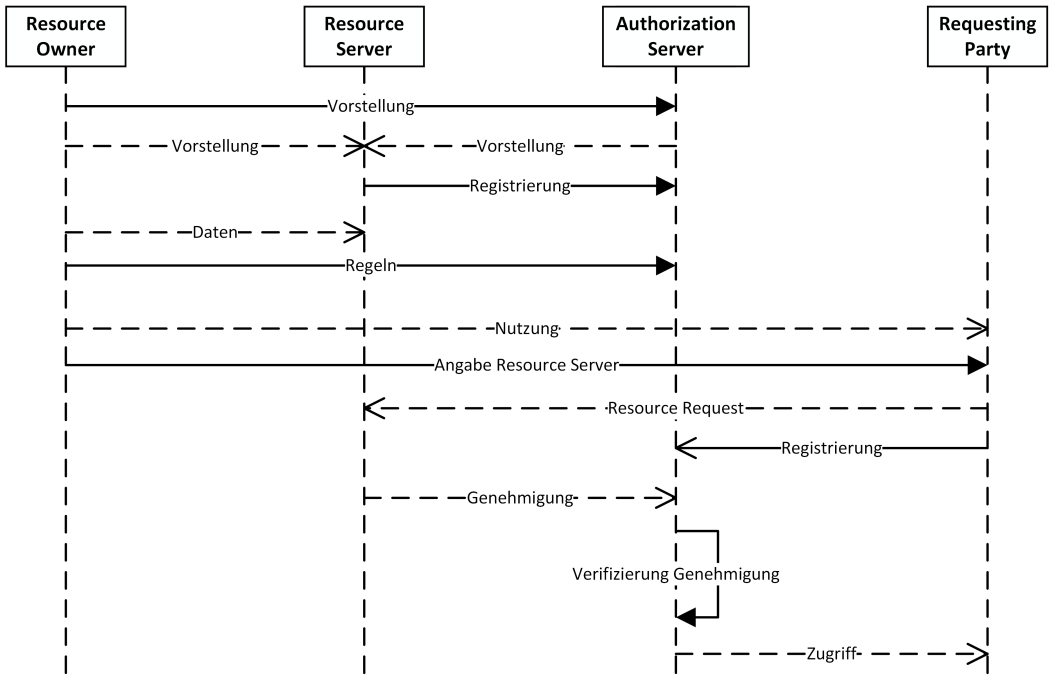

Abbildung 2.28.: Ablaufdiagramm UMA

den Dienst eines Providers nutzt, sind Schritt 2 und 3 bereits vom Provider vorgenommen worden.

4. Schritt: Alice speichert persönliche Daten im Resource Server. Zusätzlich fügt sie weitere IdPs hinzu, wo ebenfalls persönliche Daten hinterlegt sind.

5. Schritt: Alice gibt grundsätzliche Regeln im Authorization Server an, nach denen Dienste persönliche Daten von ihr erhalten.

6. Schritt: Alice möchte den Dienst des Online-Shops NewHardware4U nutzen.

7. Schritt: Alice gibt beim Bestellvorgang ihren Resource Server an.

8. Schritt: NewHardware4U fragt bei ihrem Resource Server nach ihren persönlichen Daten, wie Name und Adresse.

9. Schritt: NewHardware4U wird beim Authorization Server registriert.

10. Schritt: Die Genehmigung des Resource Servers für den Zugriff des Online-Shops NewHardware4U auf persönliche Daten wird ebenfalls registriert, bevor sie durch den Authorization Server verifiziert wird.

11. Schritt: Nachdem Alice bereits Regeln für Online-Shops spezifiziert hat, bekommt New-Hardware4U die Anschrift und Rechnungsdaten von ihrem IdP Adressverwaltung.

12. Schritt: Alice kann den Dienst wie gewohnt nutzten.

Datenschutz und Trust

Das Trust-Modell von UMA besteht aus drei Phasen:

Resource Registration: Aufbau des Vertrauensverhältnisses zwischen Authorization Server und Resource Owner sowie zwischen Resource Owner und Resource Server.

Resource Server Introduction: Aufbau des Vertrauensverhältnisses zwischen Resource Server und Authorization Manager mit dem Resource Owner als Broker, damit der Resource Server die Autorisierungsentscheidung an den Authorization Manager delegieren kann.

Data Sharing Constellations: Anfrage eines Dritten (*Requesting Party*) durch Delegation.

Dem Benutzer, aber auch den Entitäten, wird die Entscheidung überlassen, welche Entität vertrauenswürdig ist und welche Informationen ihr anvertraut werden können. Dazu gibt es die drei Säulen Access, Authorization und Consent, d. h. Zugriff, Autorisierung und Zustimmung. Nach aktuellem Stand (Januar 2016) wird dem Nutzer die Entscheidung über das Vertrauen überlassen.

Basismodell und Dienstsicht

Die Bezeichnungen der Dienstnehmer- und Dienstleisterseite unterscheidet sich bei User-Managed Access von Federated Identity Management bei Verwendung von SAML, wie in Abbildung 2.29 zu sehen:

- Die Rolle User wird bei UMA als *resource owner* bezeichnet.
- Auf Dienstnehmerseite befinden sich zudem *resource server* und *authorization server*.
- Die Dienstleisterseite besteht aus einer *requesting Party*, die Benutzerinformationen anfordert.
- Föderationen sind nicht vorhanden.

Dies hat Auswirkungen auf die Dienstsicht, wie in Abbildung 2.30 zu sehen. Es lässt sich folgendermaßen zusammenfassen:

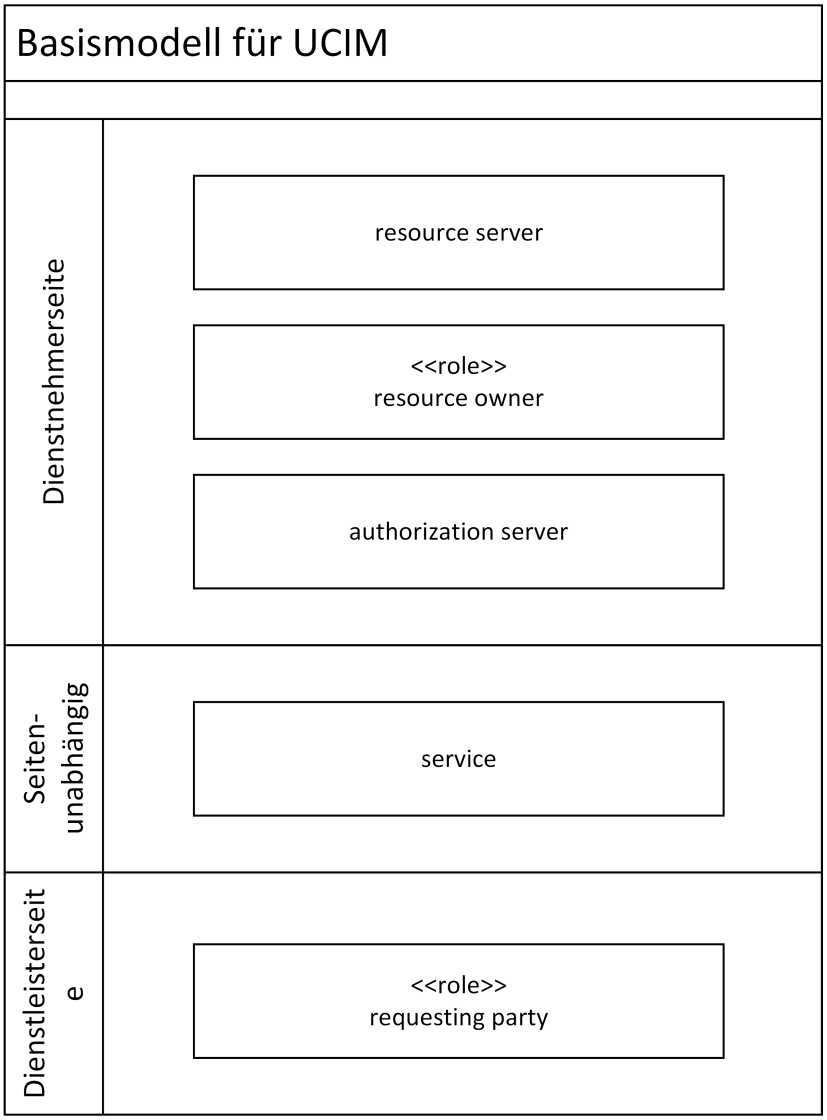

Abbildung 2.29.: Basismodell für UCIM am Beispiel UMA

- Der *resource owner* kontrolliert den *authorization server*.
- Der authorization server schützt den *resource server* mit den dort enthaltenen Benutzerinformationen.
- Der resource server wird vom resource owner verwaltet.
- Damit die *requesting party* Zugriff auf die Benutzerinformationen erhält, muss der resource owner seinen Consent geben.

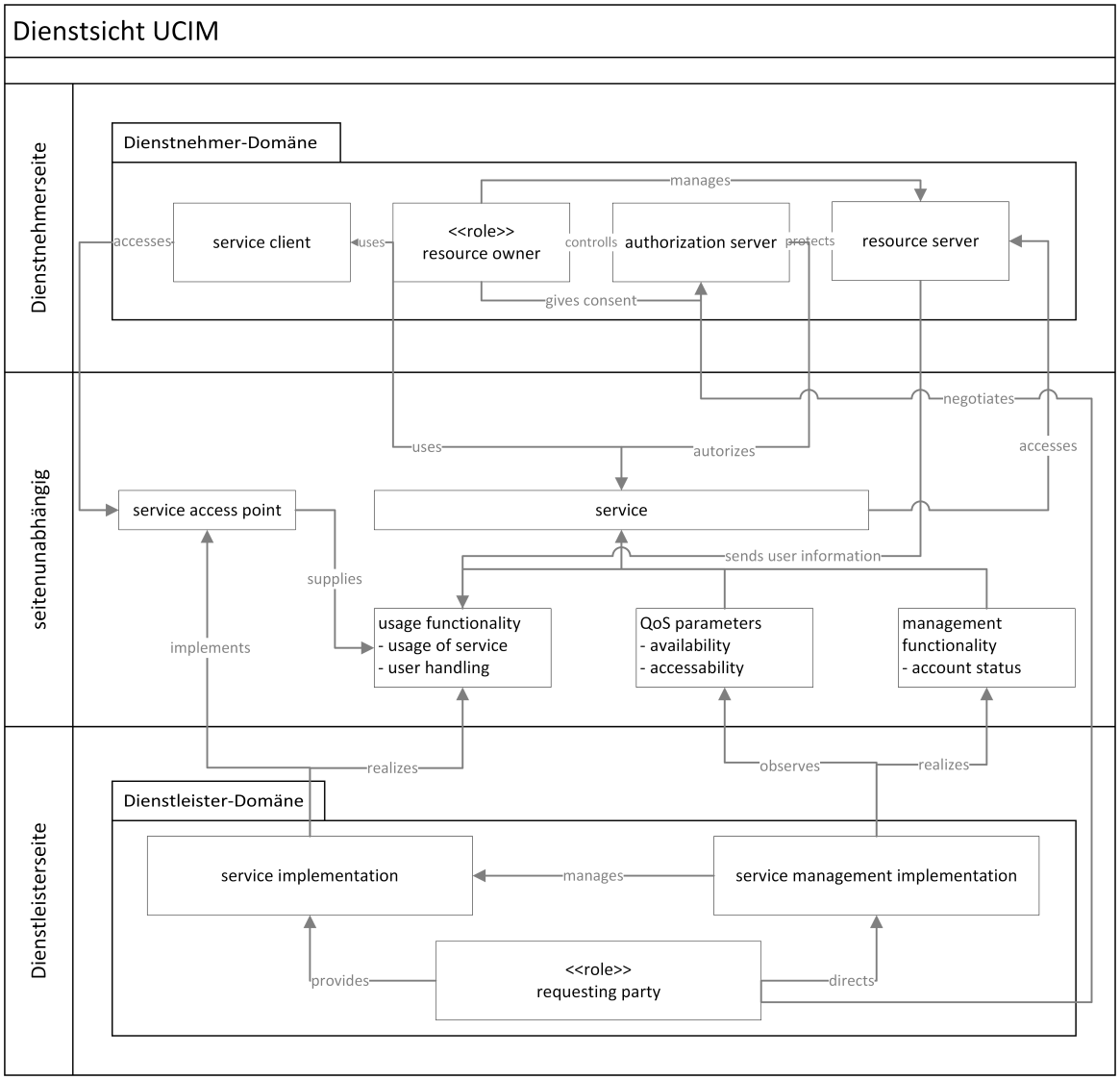

Abbildung 2.30.: Dienstsicht für UCIM am Beispiel UMA

- Ein Lokalisierungsdienst im üblichen Sinn existiert nicht.

Defizite

Aus der Architektur des UMAs ergeben sich folgende Defizite, insbesondere, wenn UMA im R&E-Umfeld einsetzt wird:

- Nachdem der Benutzer selbst die Weitergabe seiner Daten einschränkt, ist es möglich, dass ein Dienst nicht mehr funktioniert, weil er nicht alle benötigten Attribute erhält oder der Nutzer einem unzuverlässigen Dienst alle persönlichen Daten gibt.

- Der Dienst erhält Daten von unterschiedlicher Qualität. Informationen, die der Benutzer selbst einträgt (*self-asserted*), müssen nicht stimmen, während angebundene IdPs aus dem R&E-Umfeld an Mindestanforderungen ihrer Föderationen gebunden sind, die jedoch eine unterschiedliche Ausprägung haben.

- Diese Information über diesen Kontext der Attribute fehlt bei der Übertragung zum Service Provider.

Ziele

Die Ziele, die in virtuellen Föderationen bedacht werden können, konzentrieren sich vor allem auf den Nutzer, der durch UMA mehr Kontrolle über seine Daten erlangt.

- Über eine Webanwendung erhält der Benutzer eine Entscheidungshilfe, welche Regeln für welche Art von Diensten sinnvoll sind. Die Webanwendung erklärt, welche Daten für die Verwendung eines Dienstes unbedingt notwendig und welche Daten für andere Funktionalitäten, wie Profilbildung, interessant sind. Zusätzlich wird der Verwendungszweck bei optionalen Attributen genannt. Ferner bekommt der Nutzer einen Hinweis darüber, wie vertrauenswürdig der Service Provider ist.

- Der Benutzer erhält mehr Kontrolle über seine Daten. Gleichzeitig bekommt er einen Überblick darüber, welcher Dienst welche Daten von ihm erhalten hat.

- Der Benutzer kann zwischen verschiedenen IdPs und somit unterschiedlichen Benutzerkonten wählen.

- Der Kontext der Attribute wird mit den Attributen selbst versendet, damit der SP einschätzen kann, ob die gelieferten Daten die benötigte Qualität aufweisen.

2.6.4. Anforderungen

Aus dem UCIM-Szenario lassen sich weitere, auf den Benutzer fokussierte, Anforderungen ableiten.

Funktionale Anforderungen

Die funktionalen Anforderungen werden durch das oben genannte Szenario aus dem Bereich des User Centric Identity Management wie folgt ergänzt:

- Der Nutzer kann in einem erweiterten Lokalisierungsdienst zwischen verschiedenen Identitäten wählen [FA-Identitätswahl].
- Der Nutzer hat durch eine Webanwendung die Auswahl, welche Attribute er welchem Dienst zur Verfügung stellt [FA-Attributswahl].
- Der Service Provider erhält den Kontext der Attribute [FA-Kontext].
- Der Benutzer bekommt eine Entscheidungshilfe, welche Attribute von welcher Art von Dienst benötigt werden und kann daraufhin [FA-Attributswahl] anpassen [FA-Entscheidungshilfe].
- Der Nutzer hat die Möglichkeit eigene Attribute, beispielsweise für die Profilbildung, zu speichern [FA-SelfAsserted].
- Der Benutzer bekommt einen Überblick über die bisher verwendeten Dienste und die herausgegebenen Daten [FA-Monitoring].
- Die Vertrauenswürdigkeit (Level of Trust (LoT)) von SPs wird überprüft und geeignet dargestellt. Daher gilt [SEC-LoT] ebenfalls als funktionale Anforderung.

Sicherheitsanforderungen

Die funktionale Anforderung [FA-Kontext] gilt ebenfalls für die Sicherheit.

Datenschutzrechtliche Anforderungen

Die Einhaltung von Datenschutzrichtlinien ist bei UCIM, wie auch beim Teilbereich UMA, ein explizit formuliertes Ziel und steht stärker im Vordergrund als bei den anderen Varianten des FIMs. Dadurch erleichtern [FA-Monitoring] und [FA-Attributswahl] die Umsetzung der informationellen Selbstbestimmung [DSA-Selbstbestimmung].

2.7. Ergänzungen und Gewichtung

Im nachfolgenden Abschnitt werden die auf Basis der Szenarien ermittelten Anforderungen um einige weitere ergänzt. Alle Anforderungen werden anschließend zusammengefasst und gewichtet, um eine Bewertung der aktuellen Lösungen und Ansätze vornehmen zu können. Abschließend werden die Anforderungen in einer Übersichtstabelle mit ihrer Gewichtung dargestellt.

2.7.1. Ergänzende Anforderungen

Folgende Anforderungen sind für die Bewertung existierender Lösungsansätze relevant, aber nicht FIM-spezifisch oder nicht speziell einem Szenario zuzuordnen:

- Die Spezifikation des Föderationskonzeptes soll offen gelegt werden [NFA-Dokumentation].
- Die Lösung muss auf offenen Standards basieren und unter den entsprechenden Lizenzen veröffentlicht werden, um nachhaltig zu sein und eine entsprechende Akzeptanz zu bekommen [NFA-OpenSource].
- Die Sicherheit jeder Komponente sowie der gesamten Föderationen muss betrachtet werden [SEC-Multilateral].

Damit möglichst alle benötigten Attribute unabhängig vom Schema der Inter-Föderation oder Föderation versendet und interpretiert werden können, vgl. Anforderung [FA-Schema], soll diese Anforderung genauer aufgeschlüsselt werden. Zum einen sollen Regeln nicht mehrmals erstellt werden, wie im Szenario 4 zu sehen, zum anderen sollen verschiedene Implementierungen und Protokolle bedacht werden. Auch die Qualität der Regeln ist wichtig, wenn diese automatisch eingebaut werden. Somit lässt sich die Anforderung [FA-Schema] in die folgenden Anforderungen untergliedern:

- Um die Effizienz zu steigern, soll es möglich sein die Regeln wieder zu verwenden [Konv-Wiederverwendbarkeit].
- Damit ein möglichst geringer Aufwand für die Automatisierung erreicht wird, soll sich das Werkzeug automatisch in den Workflow integrieren lassen [Konv-Automatisierung].
- Gleichzeitig soll eine hohe Abdeckung vorhanden sein, d. h., es sollen alle möglichen Schemata unterstützt werden. Darunter fallen auch Schemata von Inter-Föderationen, Communities und Projekten [Konv-Abdeckung].
- Eine hohe Abdeckung ist auch bezüglich der unterschiedlichen Implementierungen und möglichst auch Protokolle wichtig. So soll die Konvertierung mit allen möglichen

SAML-Implementierungen nutzbar sein. Dies kann auch Anpassungen oder Plugins für die SAML-Implementierungen bedeuten [Konv-Implementierungsunabhängigkeit].

- Das Werkzeug muss unterschiedliche Konvertierungen ermöglichen. Diese sind insbesondere Umbenennungen, Merging/Splitting sowie syntaktische Umformungen, wie bei Datumsformaten [Konv-Konvertierungen].

- Das Werkzeug soll modulare Konvertierungen erlauben, so dass verschiedene Regeln hintereinander durchgeführt werden können [Konv-Modularität].

- Die Regeln zur Konvertierung sollen aus einer oder mehreren vertrauenswürdigen Quellen stammen oder nach einer Qualitätskontrolle freigegeben werden, so dass sich alle Beteiligten darauf verlassen können [Konv-Qualität].

Ebenso, wie die Anforderungen zu den Konvertierungsregeln, können die Anforderungen zu LoA und LoT aufgegliedert werden:

- Die Einteilung in LoA bzw. LoT soll soweit sicher sein, dass sich alle Beteiligten darauf verlassen können. Um gegebenenfalls eine höhere Sicherheit zu gewährleisten, sollen Audits ermöglicht werden [LoA/LoT-Auditing].

- Damit ein möglichst geringer Aufwand für die Automatisierung erreicht wird, soll das Werkzeug automatisch LoA bzw. LoT vergleichen können [LoA/LoT-Automatisierung].

- Die Anforderung für LoT gilt ebenfalls für den Datenschutz. Das Schema soll den Datenschutz unterstützen [LoT-Datenschutz].

- Eine Entität soll einem oder mehreren LoA/LoT eingeordnet werden können [LoA/LoT-Einordnung].

- Die einzelnen Anforderungen für LoA und LoT sollen auf eine Art gewichtet werden können. Dies kann dadurch geschehen, dass die Klassifikation eine unterschiedlich gute Abdeckung von Teilbereichen erlaubt. Im Gegenzug kann die gegenüber liegende Organisation einen oder mehrere Teilbereiche als wichtiger empfinden [LoA/LoT-Gewichtung].

- Eine hohe Abdeckung ist bezüglich der unterschiedlichen Implementierungen wichtig, um LoA bzw LoT angeben und konfigurieren zu können [LoA/LoT-Implementierungsunabhängigkeit].

- Eine Organisation muss parallel mehrere Klassifikationen verwenden können [LoA-Koexistenz].

- Der Administrator muss die Möglichkeit haben den LoA/LoT zu konfigurieren [LoA/-LoT-Konfiguration].

- Um mehrere LoA zu unterstützen und um die Umstellung zu vereinfachen, soll ein Vergleich verschiedener LoA ermöglicht werden [LoA-Mapping].
- Um eine einheitliche Lösung zu erhalten, sollen auch unterschiedliche Protokolle unterstützt werden [LoA/LoT-Protokollunabhängigkeit].
- Die Implementierung und die Inanspruchnahme der Schemata bzw. des Werkzeugs muss in akzeptabler Zeit möglich sein. Ebenso muss der Abgleich in akzeptabler Zeit möglich sein [LoA/LoT-Realisierbarkeit].
- Die Organisationen müssen das verwendete Schema registrieren können. Zudem muss die zu erstellende Klassifikation ebenfalls registriert werden können [LoA/LoT-Registrierung].
- Eine Entität hat die Möglichkeit den LoA/LoT selbst zu bestimmen [LoA/LoT-Self-Asserted].

2.7.2. Abhängigkeiten

Zwischen einzelnen Anforderungen bestehen Abhängigkeiten, die in Abbildung 2.31 visualisiert sind. Diese können von oben bis unten folgendermaßen beschrieben werden:

- ARPs gelten sowohl als Sicherheitsanforderung [SEC-ARPs] als auch als Datenschutzanforderung [DSA-ARPs].
- Automatisierung gilt für die Kategorien *FA*, *SEC* und *ORG*.
- Die Reichweite [FA-Reichweite] hängt mit der entsprechenden Bildung von Föderationen [FA-Föderationen] zusammen. Diese Anforderung gilt ebenfalls für die Sicherheit [SEC-Föderationen]. Gleichzeitig hängt [FA-Föderation] stark mit [NFA-Koexistenz] zusammen, da sowohl passende Föderationen gebildet werden sollen als auch mehrere Föderation möglich sein sollen.
- Im Bereich des Datenschutzes existieren die Anforderungen [DSA-CoCo] und [DSA-Datenschutz]. Während [DSA-CoCo] eine szenarienspezifische Anforderung darstellt, ist [DSA-Datenschutz] allgemein gültig und kann dadurch eine höhere Priorität erhalten.
- Der Kontext gilt sowohl eine funktionale Anforderung [FA-Kontext] als auch eine Sicherheitsanforderung [SEC-Kontext].
- Die Initiierung ist in den Kategorien *FA*, *SEC* und *DSA* gültig.
- Die Integration hat Auswirkungen auf *FA* und *SEC*.

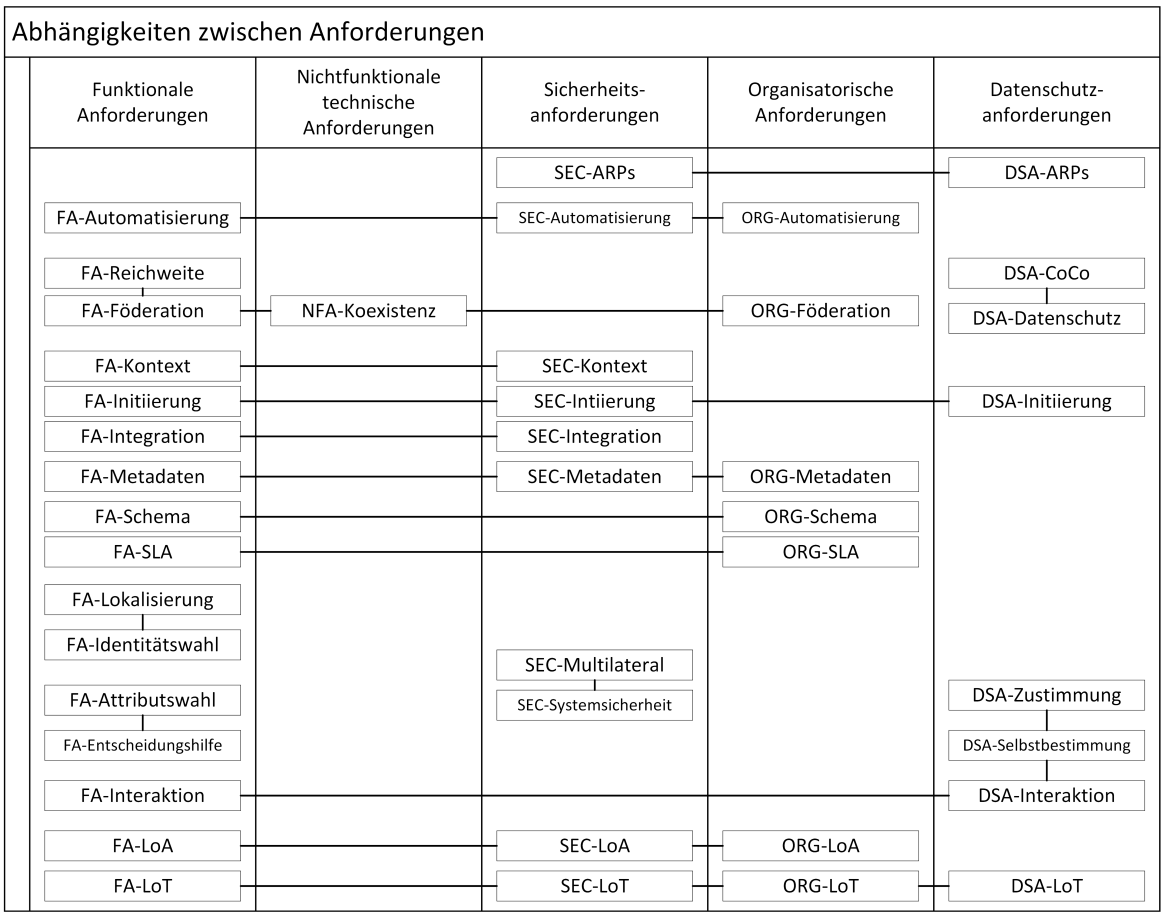

Abbildung 2.31.: Abhängigkeiten zwischen Anforderungen

- Die Anforderungen bezüglich Metadaten erstrecken sich auf *FA*, *SEC* und *ORG*.
- Das Schema ist sowohl eine funktionale Anforderung als auch eine organisatorische Anforderung.
- SLA gilt als funktionale Anforderung, aber auch als organisatorische Anforderung.
- Die Lokalisierung [FA-Lokalisierung] hängt eng mit der Identitätswahl [FA-Identitätswahl] zusammen. Während die Lokalisierung allgemein gültig ist und für jedes Protokoll verwendet werden kann, ist die Identitätswahl spezifisch für AccountChooser. Folglich kann [FA-Lokalisierung] eine höhere Priorisierung erhalten.
- Ebenso sind die Anforderungen [SEC-Multilateral] und [SEC-Systemsicherheit] eng miteinander verknüpft. Während die Systemsicherheit die Integration eines Systems in die lokale Sicherheitsumgebung betrachtet, gilt es bei der multilateralen Systemsicherheit alle Komponenten und Systeme mit einzubeziehen.
- Die Anforderungen [FA-Attributswahl] und [FA-Entscheidungshilfe] sind spezifisch für das User Centric Identity Management Szenario. Wenn der Nutzer die Möglichkeit hat seine Attribute frei zu wählen, ist eine Entscheidungshilfe sinnvoll, um ihn bei der Entscheidung zu unterstützen.
- Die Interaktion des Nutzers hat die Aspekte *FA* und *DSA*. Aus dem Gesichtspunkt des Datenschutzes ist die Zustimmung des Nutzers [DSA-Zustimmung] essentiell. Diese hilft die Selbstbestimmung des Nutzers [DSA-Selbstbestimmung] zu erreichen. Hierfür wird die Interaktion des Nutzers benötigt.
- Der Level of Assurance gilt als funktionale Anforderung, als Sicherheitsanforderung sowie als organisatorische Anforderung.
- Der Level of Trust gilt zudem als datenschutzrechtliche Anforderung.

2.7.3. Gewichtung der Anforderungen

Im Folgenden werden alle Anforderungen zusammengefasst und die im Rahmen dieser Arbeit vorgenommene Gewichtung der ermittelten Anforderungen kurz begründet. Die Gewichtung basiert auf den in den Szenarien vorgestellten Notwendigkeiten. Der daraus resultierende Anforderungskatalog kann durch seine gesamtheitliche Betrachtung als Basis für die Auswahl von FIM-Produkten und Lösungen zu bestimmten Aspekten von FIM dienen. Die verwendete Gewichtung sieht wie folgt aus:

Priorität 1 – muss: In diese Kategorie fallen MUSS-Kriterien. Wenn ein Konzept diese essentielle Anforderung nicht erfüllt, gilt es als nicht geeignet für den praktischen Einsatz.

Priorität 2 – soll: Diese SOLL-Anforderungen sind ebenfalls wichtige Indikatoren bei der Wahl eines Ansatzes, da die Eignung durch Nichterfüllen einer wichtigen Anforderung beeinträchtigt wird. Ein Konzept, welches mehrere wichtige Anforderungen nicht erfüllt, wird als praktisch nicht einsetzbar bewertet.

Priorität 3 – kann: Das Nichterfüllen von KANN-Anforderung führt zu szenarienspezifischen Einschränkungen, die toleriert werden können.

Funktionale Anforderungen

Die funktionalen Anforderungen werden wie folgt bewertet:

[FA-Aktualisierung]: Priorität 2

- Beschreibung: Aktualisierte Daten sollen an die entsprechenden Entitäten weiter geleitet werden können, damit diese Entitäten nicht mit veralteten Daten, beispielsweise Freigaben, arbeiten.
- Begründung: Falls veraltete Daten an einen Service Provider gelangen, kann es sein, dass ein Benutzer unerlaubt die Berechtigung erlangt einen Dienst zu nutzen. Gleichzeitig kann es sein, dass er einen Dienst nicht nutzen kann, den er benötigt. Daher ist diese Anforderung wichtig, jedoch nicht essentiell.

[FA-Attributswahl]: Priorität 2

- Beschreibung: Der Nutzer hat durch eine Webanwendung die Auswahl, welche Attribute er welchem Dienst zur Verfügung stellt, äquivalent zu UMA.
- Begründung: Durch die Attributswahl erhält der Nutzer eine größere Selbstbestimmung über seine Daten und kann individuell auf Basis der Freigaben durch seine Heimatorganisation die Auswahl einschränken oder ggf. erweitern.

[FA-Automatisierung]: Priorität 2

- Beschreibung: Der Vertrauensaufbau soll on demand und automatisch geschehen können, um dynamisch auf geänderte Anforderungen der Nutzer reagieren zu können. Dies soll für verschiedene Föderationen, Sektoren und Protokolle gelten.
- Begründung: Ein bisheriges Problem ist die Wartezeit für den Benutzer. Durch eine Automatisierung kann der Aufwand pro Organisation verringert und die Zeit, bis ein Dienst vollständig verwendet werden kann, reduziert werden.

[FA-Datenkategorisierung]: Priorität 1

- Beschreibung: Die zu übertragenden Daten müssen semantisch zwischen Authentifizierungs- und Autorisierungsbestätigungen sowie allgemeinen Attributsauskünften unterschieden werden können, um u. a. AAs einbinden zu können.

- Begründung: Diese Kategorisierung ist wichtig für die Realisierung von FIM-Lösungen. Die Unterscheidung zwischen Autorisierung, Authentifizierung und Attributen hilft entscheidend bei der Verarbeitung der darin enthaltenen Informationen. So können beispielsweise Authentifizierung und Attribute gemeinsam geschickt werden oder Attribute später nachgefordert werden. Zudem unterstützt die Kategorisierung die Einbindung von AAs.

[FA-Dynamik]: Priorität 2

- Beschreibung: Die Architektur der Föderationen soll sich dynamisch an geänderte Anforderungen anpassen können, um Federated Identity Management in dynamischen Umgebungen einsetzen zu können. Als zusätzlicher Aspekt soll die Dynamik der Metadaten betrachtet werden, um deren Skalierbarkeit zu verbessern.

- Begründung: Der erste Aspekt der Anforderung betrifft nur dynamische Umgebungen, wie Projekte. Jedoch ist es dort wichtig, dass Organisationen ohne Probleme hinzukommen und weggehen können. Der zweite Aspekt betrifft die Skalierbarkeit der Metadaten, was insbesondere für große Umgebungen, wie Inter-Föderationen, wichtig ist.

[FA-Entscheidungshilfe]: Priorität 3

- Beschreibung: Der Benutzer bekommt eine Entscheidungshilfe, welche Attribute von welcher Art von Dienst benötigt werden und kann daraufhin seine [FA-Attributswahl] anpassen.

- Begründung: Eine Entscheidungshilfe für den Benutzer ist wünschenswert, jedoch ist diese Hilfe auf optionale Attribute beschränkt, um weiterhin die reibungslose Nutzung eines Dienstes zu gewährleisten. Eine Unterstützung für den Nutzer ist zwar sinnvoll, jedoch können Informationen durch IdP oder SP auch die Bedeutung eines Attributs erklären.

[FA-Fehlermanagement]: Priorität 2

- Beschreibung: Die Behandlung von Fehlern soll unterstützt werden.

- Begründung: Eine aussagekräftige Fehlermeldung ist wichtig, damit Fehler entsprechend beseitigt werden können. Dies ist häufig noch nicht gegeben. Zusätzlich sind Schnittstellen zu organisationsinternen Ticket-Systemen sinnvoll, um eine nahtlose Weiterleitung von Problemen zu gewährleisten.

[FA-Föderation]: Priorität 1

- Beschreibung: Die Bildung und Verwaltung mehrerer Föderationen, egal ob national oder international, in einem oder in mehreren Sektoren, muss unterstützt werden.

- Begründung: Diese Anforderung ist essentiell, damit die Lösung verwendet werden kann. Es existieren bereits weltweit mehrere Föderationen gleichzeitig, die sich nicht zu einer großen, gemeinsamen Föderation vereinen lassen. Um nicht die Parallelität verschiedener Ansätze zu begünstigen und folglich den Aufwand für Administratoren zu erhöhen, muss die Architektur diesen Umstand berücksichtigen.

[FA-Grenzüberschreitend]: Priorität 1

- Beschreibung: Die Benutzerinformationen müssen über nationale und Sektor-Grenzen hinweg versendet werden können.

- Begründung: Damit internationale Kollaborationen auf FIM aufbauen können, muss diese Anforderung erfüllt werden.

[FA-Homeless]: Priorität 3

- Beschreibung: Nutzer ohne Heimatorganisation können die Dienste trotzdem nutzen. Eine entsprechende Schnittstelle zu einem Dienst für heimatlose Nutzer oder die direkte Integration eines solchen Dienstes muss gegeben sein.

- Begründung: Um Nutzern außerhalb von teilnehmenden Föderationen und Organisationen an einer Kollaboration teilnehmen zu lassen, ist dieser Dienst wünschenswert. Sinnvoller wäre jedoch eine Integration ohne zusätzlichen Aufwand für den Nutzer, wie beispielsweise eine passende Reichweite [FA-Reichweite].

[FA-Identitätswahl]: Priorität 3

- Beschreibung: Der Nutzer kann in einem erweiterten Lokalisierungsdienst zwischen verschiedenen Identitäten wählen.

- Begründung: Diese wünschenswerte Anforderung hilft dem Nutzer zwischen verschiedenen Identitäten zu wählen. Dies ist jedoch nicht unbedingt notwendig, da der Nutzer auch über die jeweilige Heimatorganisation seine Identität auswählen kann. Die Identitätswahl ist eine szenarienspezifische Anforderung und daher von geringerer Priorität wie die allgemeine Anforderung [FA-Lokalisierung].

[FA-Initiierung]: Priorität 2

- Beschreibung: Der Nutzer soll den Austausch von Daten und somit Aufbau des Vertrauensverhältnisses initiieren können, um u. a. die Wartezeit zu verringern.

- Begründung: Durch die Initiierung des Vertrauensverhältnisses durch den Nutzer kann wiederum die Wartezeit verringert werden. Gleichzeitig wird ggf. der Aufwand für die jeweiligen Heimatorganisationen gering gehalten.

[FA-Integration]: Priorität 1

- Beschreibung: Die Integration in die lokale Umgebung der IdPs und SPs muss ohne erheblichen Aufwand geschehen.

- Begründung: Diese Anforderung ist essentiell für die erfolgreiche Verwendung der Lösung. Hierbei müssen der Aufwand pro Organisation minimiert und aktuelle Produkte möglichst weiter verwendet werden können.

[FA-Interaktion]: Priorität 1

- Beschreibung: Der Nutzer muss die Möglichkeit haben eine Auswahl an zu versendenden Attributen zu treffen. Dazu reicht Zustimmung bzw. Ablehnung.

- Begründung: Diese Form der Interaktion ist essentiell, um die Anforderungen des Datenschutzes zu erfüllen. Ferner hilft dies bei der Akzeptanz der Lösung durch die Nutzer. Die Attributswahl wird in [FA-Attributswahl] beschrieben.

[FA-Konfiguration]: Priorität 1

- Beschreibung: Der Administrator muss die Möglichkeit haben die Integration der Metadaten und den Vertrauensaustausch zu konfigurieren.

- Begründung: Diese Anforderung ist essentiell, um die Lösung den Anforderungen jeder Organisation möglichst anpassen zu können.

[FA-Konnektor]: Priorität 2

- Beschreibung: Es sollen geeignete Schnittstellen zu den internen Systemen bestehen.

- Begründung: Dies ist wichtig für eine nahtlose Integration. Ansonsten müssen eigene Konnektoren erstellt werden, was zu einem erhöhten Aufwand und einer schlechteren Realisierbarkeit führt.

[FA-Kontext]: Priorität 3

- Beschreibung: Der Service Provider erhält den Kontext der Attribute.

- Begründung: Diese Information ist sinnvoll, um das Vertrauen in die Datenqualität besser einschätzen zu können. Solange Attribute nicht aus verschiedenen Quellen aggregiert versendet werden, ist der zuständige IdP bereits in den Metadaten

genannt. Daher ist diese Funktion nur empfehlenswert, jedoch nicht erforderlich.

[FA-Langlebigkeit]: Priorität 1

- Beschreibung: Die Architektur der Föderationen muss eine möglichst langfristige und dauerhafte Lösung ermöglichen.
- Begründung: Eine langfristige Planungsmöglichkeit ist notwendig, damit die Lösung akzeptiert wird.

[FA-LoA]: Priorität 2

- Beschreibung: Dem SP soll angezeigt werden, welche Datenqualität der IdP liefern kann. Zugleich soll die Einteilung in eine bestimmte Klasse transparent erfolgen. Falls unterschiedliche Klassifikationen verwendet werden, sollen diese verglichen werden können.
- Begründung: Eine Einteilung in Verlässlichkeitsklassen ermöglicht es den SPs eine einfachere Auswahl der Identity Provider zu treffen. Da die Verlässlichkeit bereits geeignet angezeigt werden kann durch die beiden möglichen Protokolle SAML und OpenID und da SPs bei der Auswahl von IdPs Unterstützung erhalten sollen, ist diese Anforderung wichtig.

[FA-Lokalisierung]: Priorität 1

- Beschreibung: Der Nutzer muss die Möglichkeit haben seine Heimatorganisation bei einem Lokalisierungsdienst auszuwählen.
- Begründung: Damit Service Provider wissen, von welchem Identity Provider der Nutzer kommt und von welchem sie Attribute anfragen müssen, ist die Lokalisierung des Nutzers durch einen Lokalisierungsdienst zwingend notwendig.

[FA-LoT]: Priorität 2

- Beschreibung: Die Vertrauenswürdigkeit von SPs soll überprüft und geeignet dargestellt werden.
- Begründung: Die Information über die Vertrauenswürdigkeit von Service Providern erleichtert IdPs bei der Entscheidung über die Akzeptanz des jeweiligen SPs. Dies wiederum beschleunigt die Verwendbarkeit eines Dienstes durch die Nutzer und unterstützt den dynamischen Aufbau von Vertrauen, wodurch diese Anforderung umgesetzt werden soll.

[FA-Metadaten]: Priorität 2

- Beschreibung: Die Metadaten enthalten Informationen über die Organisationen

und sollen passend generiert werden.

- Begründung: Metadaten sind wichtig, um die Kommunikationsendpunkte zu identifizieren. Da generische Metadaten zwar praktisch, aber nicht zwingend notwendig sind, wird diese Anforderung mit Priorität 2 bewertet.

[FA-Monitoring]: Priorität 2

- Beschreibung: Der Benutzer bekommt einen Überblick über die bisher verwendeten Dienste und die herausgegebenen Daten.

- Begründung: Diese Anforderung bietet zusätzliche Funktionalität bezüglich des Datenschutzes, was wünschenswert ist.

[FA-Pull&Push]: Priorität 1

- Beschreibung: Der SP muss den Abruf von Daten vom IdP initiieren können.

- Begründung: Damit der Datenabruf durch Service Provider initiiert werden kann und keine zusätzliche Wartezeit entsteht, ist diese Funktionalität essentiell.

[FA-Realisierbarkeit]: Priorität 1

- Beschreibung: Die Implementierung und die Teilnahme an Föderationen müssen in akzeptabler Zeit möglich sein.

- Begründung: Dies ist wichtig, da ansonsten die Lösung nicht eingesetzt werden kann.

[FA-Reichweite]: Priorität 2

- Beschreibung: Die Lösung soll alle benötigten IdPs und SPs umfassen.

- Begründung: Damit jeder Nutzer mitmachen kann und um die Parallelität von Föderationen mit schlechter administrierbaren Ad-hoc-Lösungen zu vermeiden, ist die Umsetzung dieser Anforderung wichtig.

[FA-Rollen]: Priorität 2

- Beschreibung: Organisationen sollen parallel in mehreren Rollen agieren können.

- Begründung: Organisationen können parallel als IdP und SP agieren. Damit der Aufwand pro Organisation gering gehalten wird, ist es sinnvoll, wenn die Lösung verschiedene Rollen parallel unterstützt.

[FA-Schema]: Priorität 2

- Beschreibung: Es sollen alle benötigten Attribute unabhängig vom Schema der Inter-Föderation oder Föderation versendet und interpretiert werden können.

- Begründung: Diese Anforderung ist wichtig, damit Attribute unabhängig vom Schema versendet und interpretiert werden können. Dies hat zur Folge, dass Dienste in vollem Umfang genutzt werden können und dass die Wartezeit für Nutzer reduziert wird.

[FA-SelfAsserted]: Priorität 3

- Beschreibung: Der Nutzer hat die Möglichkeit eigene Attribute, beispielsweise für die Profilbildung, zu speichern.

- Begründung: Da die allermeisten Attribute in den Heimatorganisationen der Nutzer vorliegen, ist diese Anforderung zwar wünschenswert, aber nicht notwendig.

[FA-SLA]: Priorität 3

- Beschreibung: Die Anzahl der benötigten Verträge kann reduziert bzw. dem Nutzerkreis angepasst werden.

- Begründung: Damit keine unnötigen oder redundanten Verträge geschlossen werden und somit der Aufwand pro Organisation minimiert wird, ist die Reduzierung der Verträge durch eine zusätzliche Funktion sinnvoll, aber nicht notwendig.

Nichtfunktionale technische Anforderungen

Die nichtfunktionalen technischen Anforderungen werden wie folgt gewichtet.

[NFA-Dokumentation]: Priorität 1

- Beschreibung: Die Spezifikation des Föderationskonzeptes muss offen gelegt werden.

- Begründung: Die Dokumentation der Lösung ist essentiell, damit sie geeignet implementiert, erweitert und aktualisiert werden kann. Zusätzlich ist die Dokumentation wichtig für den Betrieb und die Migration.

[NFA-Implementierungsunabhängigkeit]: Priorität 2

- Beschreibung: Das System soll unterschiedliche Implementierungen akzeptieren.

- Begründung: Die Umsetzung der Anforderung ist wichtig für die Akzeptanz und Verwendung, u. a. damit ein möglichst großer Nutzerkreis abgedeckt werden kann.

[NFA-Koexistenz]: Priorität 1

- Beschreibung: Es muss möglich sein, dass eine Entität mehreren Föderationen zugehört.
- Begründung: Damit den aktuellen Geschäftsbeziehungen und Projekten von IdPs und SPs Rechnung getragen wird, ist diese nichtfunktionale technische Anforderung essentiell, da ansonsten, u. a. bedingt durch den entstehenden Mehraufwand, die Akzeptanz der Lösung fehlt.

[NFA-OpenSource]: Priorität 1

- Beschreibung: Die Lösung muss auf offenen Standards basieren und unter den entsprechenden Lizenzen veröffentlicht werden.
- Begründung: Offene Standards sind elementar, damit die Lösung geeignet erweitert und aktualisiert werden kann. Zusätzlich helfen offene Standards für die Nachhaltigkeit und die Erreichung der notwendigen Akzeptanz.

[NFA-Performanz]: Priorität 2

- Beschreibung: Das System soll performant sein, beispielsweise auf Anfragen in akzeptabler Zeit reagieren. Akzeptabel sind dabei Zeiten im unteren Sekundenbereich.
- Begründung: Die Performanz des Systems ist wichtig für die Akzeptanz durch Nutzer. Wenn das System nicht in akzeptabler Zeit reagiert, ist es möglich, dass Nutzer entsprechen erreichbare Dienste nicht verwenden wollen oder Umwege finden.

[NFA-Portabilität]: Priorität 3

- Beschreibung: Das System soll unabhängig von der Hardware und dem Betriebssystem verwendet werden können.
- Begründung: Die Portabilität wirkt sich positiv auf die Integration aus, ist jedoch nicht notwendig für den professionellen Betrieb.

[NFA-Protokollunabhängigkeit]: Priorität 2

- Beschreibung: Das System soll unterschiedliche Protokolle akzeptieren.
- Begründung: Die Akzeptanz verschiedener Protokolle ist wichtig für die Akzeptanz durch Organisationen und für die weitere Verwendung, insbesondere im Grid-Bereich. Durch fehlende Protokollunabhängigkeit kann ein Mehraufwand auf Seiten der Organisationen entstehen, um geeignete Lösungen zu finden.

[NFA-Skalierbarkeit]: Priorität 1

- Beschreibung: Das System muss skalierbar für eine beliebige Anzahl an Teilnehmern und dynamischen Föderationen sein. Skalierbarkeit hat hierbei unterschiedliche Bedeutungen, die allesamt erfüllt werden sollen. Zum einen soll das System für eine beliebige Anzahl an Teilnehmern und Föderationen skalieren. Zum anderen soll die Skalierbarkeit bezüglich des Metadatenaustausches verbessert werden. Hierbei werden die Anzahl an manuellen Schritten und die Größe des Metadatensatzes beachtet.

- Begründung: Die Anzahl an Benutzern, Service Providern, Identity Providern und Föderationen muss sich dynamisch verändern können. Diese Anforderung ist essentiell, da Federated Identity Management wegen Skalierbarkeitsproblemen etabliert wurde und die Skalierbarkeit insbesondere bei Projekten weiterhin ein Problem darstellt.

[NFA-Usability]: Priorität 2

- Beschreibung: Die Benutzeroberflächen sollen intuitiv zu bedienen und selbsterklärend sein.

- Begründung: Die Akzeptanz durch Nutzer hängt wesentlich von der Benutzbarkeit der Oberflächen ab. Daher ist diese Anforderung wichtig.

Sicherheitsanforderungen

Die Sicherheitsanforderungen werden wie folgt gewichtet:

[SEC-ARPs]: Priorität 1

- Beschreibung: Die Abfrage von Benutzerinformationen muss auf ausgewählte Attribute eingeschränkt werden können.

- Begründung: Die Einschränkung ist essentiell, um den Datenschutz zu erfüllen. Beispielsweise ermöglicht dies keine rein internen Informationen an externe SPs zu liefern.

[SEC-Auditing]: Priorität 2

- Beschreibung: Das System und seine Teilkomponenten sollen auditierbar sein.

- Begründung: Die Nachvollziehbarkeit von Änderungen ist wichtig für das Sicherheitskonzept von Organisationen.

[SEC-Authentifizierung]: Priorität 1

- Beschreibung: Benutzer müssen sich vor der Nutzung eines Dienstes authentifizieren können.
- Begründung: Diese Anforderung ist essentiell, damit nur tatsächliche Nutzer einen Dienst verwenden können.

[SEC-Automatisierung]: Priorität 2

- Beschreibung: Bei der Automatisierung des Datenaustausches soll die Sicherheit gewährleistet werden.
- Begründung: Die Verringerung von Wartezeit für Nutzer und der verminderte Aufwand auf Seiten der IdPs müssen sicher sein. Gleichzeitig darf diese Funktionalität keine Beeinträchtigung auf die Systemsicherheit haben.

[SEC-Datenübertragung]: Priorität 1

- Beschreibung: Die Übertragung der Benutzerinformationen muss sicher sein.
- Begründung: Eine sichere Übertragung ist essentiell, um die enthaltenen Daten zu schützen. Dies beinhaltet, dass die Gegenseite identifiziert und authentifiziert werden muss. Zusätzlich muss die Integrität, beispielsweise durch kryptographische Prüfsummen, und die Verschlüsselung der Daten gewährleistet werden.

[SEC-Initiierung]: Priorität 2

- Beschreibung: Die Initiierung des Vertrauensaufbaus soll sicher sein.
- Begründung: Durch die Initiierung des Vertrauensaufbaus durch den Nutzer wird die Entscheidungsgewalt vom Identity Provider zum Nutzer verlagert. Obwohl diese Kontrollinstanz automatisiert wird (vgl. [SEC-Automatisierung]), muss der Vertrauensaufbau für den IdP sicher sein.

[SEC-Integration]: Priorität 2

- Beschreibung: Die Integration soll aus Sicht der Sicherheit betrachtet werden.
- Begründung: Die Einführung und der Betrieb einer neuen Lösung sind nur dann sinnvoll, wenn die Sicherheit bei der Integration sichergestellt ist.

[SEC-Kontext]: Priorität 3

- Beschreibung: Der Service Provider erhält den Kontext der Attribute.
- Begründung: Der Kontext der Authentifizierung und Autorisierung hilft bei der

Entscheidung des SPs, ob er einen Nutzer zulässt oder nicht. Daher ist die Anforderung wünschenswert, jedoch kann diese Information auch anderweitig abgefragt werden.

[SEC-LoA]: Priorität 2

- Beschreibung: Benutzer dürfen einen Dienst nur dann nutzen, wenn ihr IdP den Qualitätsbestimmungen des SPs entspricht.
- Begründung: Die Vertraulichkeit ist für Service Provider wichtig, um zu entscheiden, ob ein Identity Provider den Anforderungen entspricht.

[SEC-LoT]: Priorität 2

- Beschreibung: Benutzerinformationen dürfen nur an Service Provider weiter gegeben werden, zu denen ein Vertrauensverhältnis besteht und die genügend vertrauenswürdig sind.
- Begründung: Diese Vertrauensinformation ist für Identity Provider wichtig, um zu entscheiden, ob ein Service Provider sicher genug ist, um die Benutzerinformationen zu erhalten.

[SEC-Metadaten]: Priorität 3

- Beschreibung: Die Verwaltung und Aktualisierung sicherheitsrelevanter Konfigurationsparameter, wie Metadaten, kann weitgehend automatisiert werden, so dass diese Metadaten nur noch an einer zentralen Stelle gepflegt werden müssen.
- Begründung: Die zentrale und automatisierte Verwaltung von Metadaten hilft den Aufwand für Organisationen zu minimieren. Da Metadaten meist öffentlich zugänglich sind, ist die Sicherheitsanforderung diesbezüglich verhältnismäßig gering.

[SEC-Multilateral]: Priorität 1

- Beschreibung: Die Sicherheit jeder Komponente sowie der gesamten Föderationen muss betrachtet werden.
- Begründung: Diese Anforderung ist essentiell, damit die Sicherheit des gesamten Systems sowie jeder einzelnen Komponente gewährleistet ist.

[SEC-Systemsicherheit]: Priorität 1

- Beschreibung: Das System muss sich nahtlos in bestehende Netzwerk- und Systemsicherheitsprozesse integrieren lassen.

- Begründung: Diese Anforderung ist, wie [SEC-Multilateral], grundlegend, um die Sicherheit des gesamten Systems herzustellen.

Organisatorische Anforderungen

Die organisatorischen Anforderungen werden wie folgt gewichtet.

[ORG-Automatisierung]: Priorität 2

- Beschreibung: Die Automatisierung der Verbindungserstellung hat Auswirkungen auf die Organisation, die beachtet werden sollen.
- Begründung: Diese Anforderung ist wichtig, damit die organisationsinternen Prozesse an die Automatisierung der Workflows angepasst werden kann. Dabei sollen die Umstellungen möglichst gering sein. Ferner soll die Lösung möglichst leicht an die internen Anforderungen angepasst werden können.

[ORG-Föderation]: Priorität 2

- Beschreibung: Die Bildung und Verwaltung nach verschiedenen Föderationsmodellen soll passend durch die Lösung unterstützt werden.
- Begründung: Organisationen nehmen in verschiedenen Föderationen teil, die unterschiedlichen Modellen entsprechen. Damit der Aufwand in den Organisationen gering gehalten wird, ist es wichtig, dass die Lösung verschiedene Föderationsmodelle unterstützt.

[ORG-Konfiguration]: Priorität 2

- Beschreibung: Auf organisatorischer Ebene betrifft die Konfiguration die internen Abläufe.
- Begründung: Bei der Konfiguration ist wichtig, dass die organisatorischen Anforderungen abgebildet werden können, damit die Lösung akzeptiert wird.

[ORG-LoA]: Priorität 2

- Beschreibung: Die Verlässlichkeitsklasse betrifft interne Abläufe und die Konfiguration.
- Begründung: Der LoA soll für SPs entsprechend den internen Anforderungen konfiguriert und angepasst werden. Gleichzeitig ist es wichtig, dass der Level of Assurance der IdPs ohne erheblichen Mehraufwand ermittelt werden kann.

[ORG-LoT]: Priorität 2

- Beschreibung: Das Vertrauen in den SP betrifft interne Abläufe sowie die Konfiguration.

- Begründung: Der LoT soll für Identity Provider entsprechend den internen Anforderungen konfiguriert und angepasst werden. Gleichzeitig ist es wichtig, dass der Level of Trust der SPs ohne erheblichen Mehraufwand ermittelt werden kann.

[ORG-Metadaten]: Priorität 3

- Beschreibung: Die Metadaten können passend mit den Informationen über die Organisationen generiert werden.

- Begründung: Die Automatisierung der Metadatengenerierung ist sinnvoll, damit die Organisationsinformationen ohne Aufwand in die Metadaten eingefügt werden können. Jedoch können Metadaten auch manuell editiert werden, während Änderungen selten vorkommen, wodurch der Aufwand nicht stark ins Gewicht fällt.

[ORG-Migration]: Priorität 2

- Beschreibung: Für die Migration sollen geeignete Lösungen bereitgestellt werden.

- Begründung: Migrationskonzepte sind wichtig, da die Migration auf neue Lösung stark davon abhängen.

[ORG-Realisierbarkeit]: Priorität 1

- Beschreibung: Der Aufwand für Realisierung und Betrieb muss angemessen sein.

- Begründung: Diese Anforderung ist essentiell, da komplexe Lösungen und lange Wartezeiten für die Praxis irrelevant sind.

[ORG-Registrierung]: Priorität 1

- Beschreibung: Die Organisationen müssen sich in der Lösung registrieren können. Hierfür sollen Föderationen auch bestimmte Anforderungen für die Aufnahme aufstellen können.

- Begründung: Die Registrierung ist essentiell, damit Organisationen an den Föderationen teilnehmen können und damit die Integration bestehender Abläufe reibungslos verläuft.

[ORG-Schema]: Priorität 2

- Beschreibung: Ein organisationsübergreifendes Modell zum Datenaustausch soll verwendet werden.

- Begründung: Damit organisatorische Anforderungen an die FIM-Lösung umgesetzt werden können, ist ein organisationsübergreifendes Modell bzw. eine Transformation von Daten notwendig. Folglich muss die FIM-Lösung mit dem internen Schema und dem ggf. verwendeten externen Schema umgehen können. Eine Nichteinhaltung dieser Anforderung hat einen Mehraufwand insbesondere bei IdPs zur Folge, was die Akzeptanz der Lösung mindert.

[ORG-SLA]: Priorität 3

- Beschreibung: Die Datenqualität und andere Gütemerkmale können passend spezifiziert werden. Die Einhaltung der SLAs kann automatisch überprüft werden.

- Begründung: Nachdem wenige SLAs erstellt, diese aber trotzdem abgebildet werden müssen, ist diese Anforderung sinnvoll. Nachdem die automatische Überprüfung der Einhaltung nicht notwendig ist, wird diese Anforderung mit Priorität 3 bewertet.

[ORG-Supportprozesse]: Priorität 2

- Beschreibung: Die Lösung soll Schnittstellen zu den organisationsinternen und organisationsübergreifenden Supportprozessen, wie dem Service Desk und dem Change Management, aufweisen.

- Begründung: Diese Anforderung ist wichtig, da das Nichtvorhandensein von diesen Schnittstellen große Auswirkungen auf innerorganisatorische Abläufe und Betrieb hat und zu Mehraufwand führt.

[ORG-Validierung]: Priorität 1

- Beschreibung: Registrierte Organisationen müssen validiert und überprüft werden können. Dies kann beispielsweise über eine Instanz der Föderation geschehen.

- Begründung: Eine Kontrollinstanz ist notwendig, damit nur valide Entitäten Daten abrufen und Dienste nutzen können. Ansonsten kann es zu negativen Folgen für den Datenschutz kommen.

Datenschutzrechtliche Anforderungen

Die datenschutzrechtlichen Anforderungen werden wie folgt gewichtet.

[DSA-ARPs]: Priorität 1

- Beschreibung: Die Daten, die an einen SP gesendet werden, müssen vorher gefiltert werden. Benutzer müssen dabei kontrollieren können, welche Daten an einen

Service Provider gesendet werden.

- Begründung: Die Kontrolle des Nutzers über seine Daten ist eine grundlegende Anforderung, die erfüllt werden muss.

[DSA-CoCo]: Priorität 3

- Beschreibung: Die Datenschutzrichtlinie der EU kann überprüft werden.

- Begründung: Diese Anforderung ist eine szenarienspezifische Anforderung. Wenn [DSA-Datenschutz] eingehalten und dies visualisiert ist, wird die Anforderung [DSA-CoCo] ebenfalls erfüllt. Daher erhält sie eine geringere Priorität.

[DSA-Datenschutz]: Priorität 1

- Beschreibung: Die Datenschutzrichtlinien und Datenschutzgesetze müssen eingehalten werden können.

- Begründung: Diese Anforderung ist ebenfalls grundlegend und muss daher eingehalten werden. Dazu ist es wichtig anzuzeigen, auf welchen Datenschutz sich die Organisation bezieht und wie sich dieser zu den Anforderungen an den Datenschutz des Geschäftspartners verhält.

[DSA-Initiierung]: Priorität 3

- Beschreibung: Da durch die Initiierung und Automatisierung des Vertrauensaufbaus durch den Nutzer Benutzerinformationen an den gewählten Service Provider nach [DSA-Zustimmung] geschickt werden, ist die Initiierung ebenfalls datenschutzrechtlich relevant. Diese kann dem Benutzer geeignet dargestellt werden.

- Begründung: Die Visualisierung des Vertrauensaufbaus für den Benutzer ist wichtig, jedoch gibt es weitere, wichtigere Mechanismen zur Sicherstellung des Datenschutzes, wie [DSA-ARPs], die auch ohne dieser Funktion greifen.

[DSA-Interaktion]: Priorität 2

- Beschreibung: Die Interaktion des Benutzers soll möglich sein.

- Begründung: Die Interaktion des Benutzers ist wichtig für die Akzeptanz der Lösung.

[DSA-LoT]: Priorität 2

- Beschreibung: Durch bzw. trotz der Funktion Level of Trust soll der Datenschutz eingehalten werden.

- Begründung: Die Bestimmung des Level of Trust bei Service Providern ist ein Hilfsmittel, um die Vertrauenswürdigkeit des jeweiligen SPs zu ermitteln, jedoch nicht zwingend notwendig, um den Datenschutz zu erfüllen.

[DSA-Selbstbestimmung]: Priorität 1

- Beschreibung: Die Einhaltung von Datenschutzrichtlinien ist bei UCIM ein explizit formuliertes Ziel und steht stärker im Vordergrund als bei den anderen Varianten des FIMs. Dadurch erleichtern [FA-Monitoring] und [FA-Attributswahl] die Umsetzung der informellen Selbstbestimmung.

- Begründung: Die informelle Selbstbestimmung ist essentiell, damit der Nutzer Federated Identity Management akzeptiert. Zudem erleichtert es die datenschutzrechtlichen Bestimmungen einzuhalten.

[DSA-Zustimmung]: Priorität 2

- Beschreibung: Es ist erforderlich, dass die Benutzer über die Weitergabe ihrer Daten an Dritte informiert werden und dieser zustimmen.

- Begründung: Sofern dies nicht im Lokalisierungsdienst integriert ist, muss dies bei der ersten Nutzung eines SPs nachgeholt werden.

2.8. Anforderungskatalog

Die nachfolgende Tabelle fasst alle Anforderungen und ihre Prioritäten zusammen. Sie dient als Anforderungskatalog in den weiteren Kapiteln.

Somit gibt es 25 essentielle und 35 wichtige Anforderungen, die für die Auswahl von FIM-Ansätzen besonders relevant sind.

Anforderung	Priorität	Anforderung	Priorität
Funktionale Anforderungen			
[FA-Aktualisierung]	2	[FA-Konnektor]	2
[FA-Attributswahl]	2	[FA-Kontext]	3
[FA-Automatisierung]	2	[FA-Langlebigkeit]	1
[FA-Datenkategorisierung]	1	[FA-LoA]	2
[FA-Dynamik]	2	[FA-Lokalisierung]	1
[FA-Entscheidungshilfe]	3	[FA-LoT]	2
[FA-Fehlermanagement]	2	[FA-Metadaten]	2
[FA-Föderation]	1	[FA-Monitoring]	2
[FA-Grenzüberschreitend]	1	[FA-Pull&Push]	1
[FA-Homeless]	3	[FA-Realisierbarkeit]	1
[FA-Identitätswahl]	3	[FA-Reichweite]	2
[FA-Initiierung]	2	[FA-Rollen]	2
[FA-Integration]	1	[FA-Schema]	2
[FA-Interaktion]	1	[FA-SelfAsserted]	3
[FA-Konfiguration]	1	[FA-SLA]	3
Nichtfunktionale technische Anforderungen			
[NFA-Dokumentation]	1	[NFA-Portabilität]	3
[NFA-Implementierungsunabhängigkeit]	2	[NFA-Protokollunabhängigkeit]	2
[NFA-Koexistenz]	1	[NFA-Skalierbarkeit]	1
[NFA-OpenSource]	1	[NFA-Usability]	2
[NFA-Performanz]	2		
Sicherheitsanforderungen			
[SEC-ARPs]	1	[SEC-Kontext]	3
[SEC-Auditing]	2	[SEC-LoA]	2
[SEC-Authentifizierung]	1	[SEC-LoT]	2
[SEC-Automatisierung]	2	[SEC-Metadaten]	3
[SEC-Datenübertragung]	1	[SEC-Multilateral]	1
[SEC-Initiierung]	2	[SEC-Systemsicherheit]	1
[SEC-Integration]	2		
Organisatorische Anforderungen			
[ORG-Automatisierung]	2	[ORG-Realisierbarkeit]	1
[ORG-Föderation]	2	[ORG-Registrierung]	1
[ORG-Konfiguration]	2	[ORG-Schema]	2
[ORG-LoA]	2	[ORG-SLA]	3
[ORG-LoT]	2	[ORG-Supportprozesse]	2
[ORG-Metadaten]	3	[ORG-Validierung]	1
[ORG-Migration]	2		
Datenschutzanforderungen			
[DSA-ARPs]	1	[DSA-Interaktion]	2
[DSA-CoCo]	3	[DSA-LoT]	2
[DSA-Datenschutz]	1	[DSA-Selbstbestimmung]	1
[DSA-Initiierung]	3	[DSA-Zustimmung]	2

Tabelle 2.3.: Anforderungen

Kapitel 3

Status Quo

Inhalt dieses Kapitels

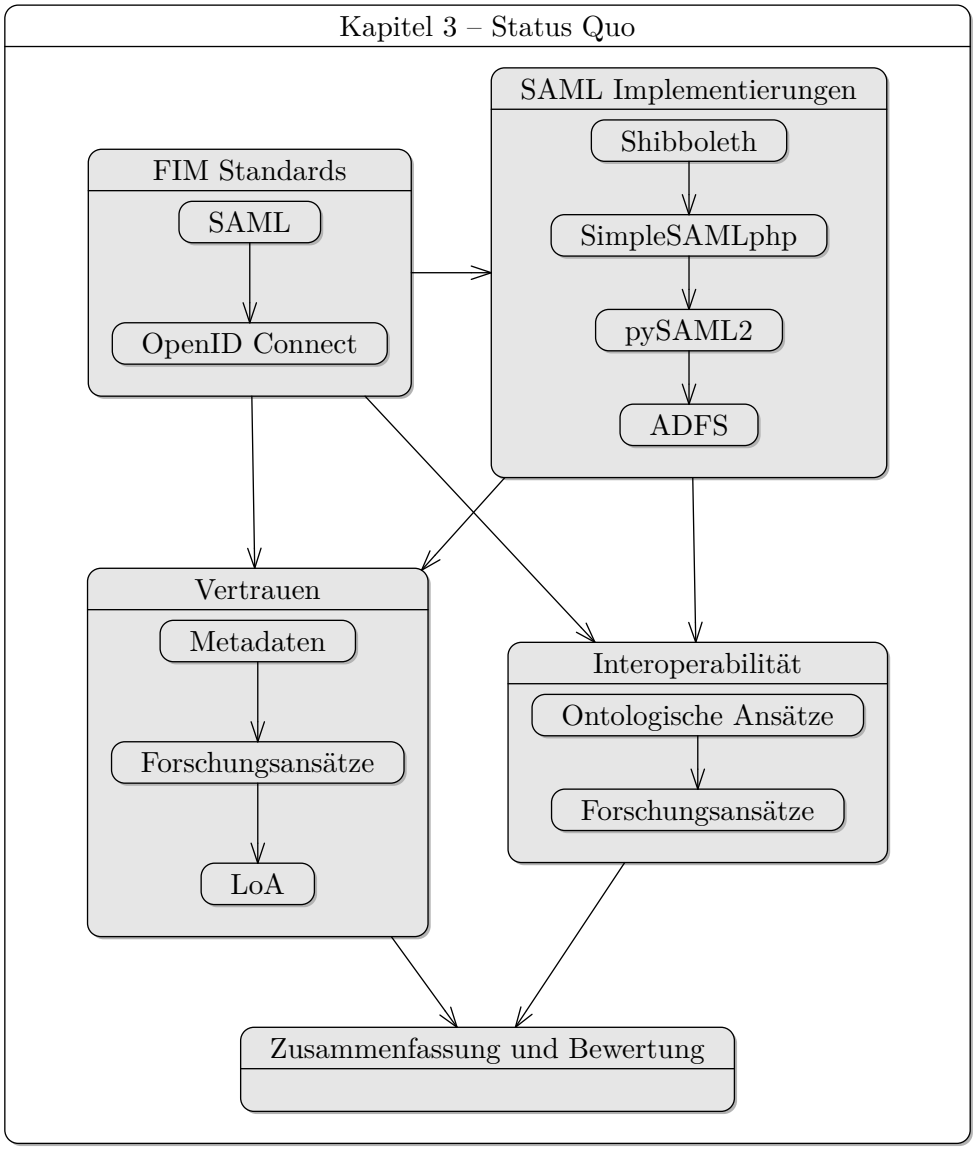

Abbildung 3.1.: Vorgehensmodell in diesem Kapitel

In diesem Kapitel werden, wie in Abbildung 3.1 dargestellt, existierende *Konzepte* und *Lösungen* aus Forschung und Industrie vorgestellt. Ihre Stärken und Defizite werden anhand des *Anforderungskatalogs* aufgezeigt und darauf basierend wird ihr potentieller Beitrag zum Lösungsansatz beurteilt. Zunächst werden in den Abschnitten 3.1 und 3.2 die FIM-Standards

und ausgewählte Implementierungen bewertet. Dabei werden unterschiedliche Aspekte näher betrachtet:

- *Aufbau*: Der Aufbau des Protokolls bzw. der Implementierung. Bei der Implementierung wird insbesondere der technische Aufbau betrachtet.

- *Lokalisierungsdienst*: Der Lokalisierungsdienst ist für die Umsetzung der essentiellen Anforderung [FA-Lokalisierung] zuständig. Zusätzlich kann er die Erfüllung weiterer Anforderungen, wie beispielsweise [FA-Grenzüberschreitend], [FA-Identitätswahl], [FA-Initiierung], [FA-Reichweite], [SEC-Initiierung] und [DSA-Initiierung], unterstützen.

- *Attribute Handling*: Attribute Handling ermöglicht das Filtern von Benutzerinformationen, was zur Erfüllung der essentiellen Anforderung [SEC-ARPs] beiträgt. Zusätzlich ist dieser Aspekt bei der Umwandlung von Attributen in das vom SP gewünschte Format, [FA-Schema], wichtig. Die Konfigurationsmöglichkeiten zeigen in einem beispielhaften Bereich, inwiefern [FA-Konfiguration] beachtet wird. Folglich ist Attribute Handling auch ein bedeutender Faktor für die Umsetzung von [FA-Integration], [SEC-Integration] und [ORG-Konfiguration].

- *User Consent*: Die Zustimmung des Nutzers zur Übertragung der Daten an einen SP ist grundlegend für die Erfüllung der Anforderungen an den Datenschutz, vgl. Anforderungen [FA-Interaktion], [DSA-Datenschutz] und [DSA-Zustimmung]. Zusätzlich bietet dieses Tool zur expliziten Zustimmung des Nutzers die Möglichkeit die Anforderungen [FA-Attributswahl], [FA-Entscheidungshilfe], [FA-SelfAsserted], [DSA-Interaktion] und [DSA-Selbstbestimmung] zu realisieren. Ferner ist die Anforderung [NFA-Usability] in diesem Zusammenhang wichtig.

- *Verlässlichkeitsklassen*: Verlässlichkeitsklasse bzw. deren Vergleichbarkeit ist ein Defizit in aktuellen Föderationen und Inter-Föderationen, vgl. wichtige Anforderungen [FA-LoA] und [ORG-LoA]. Der Faktor Vertrauen bildet eine Basis für die Automatisierung des Vertrauensaufbaus, der sich in den Anforderungen [FA-Automatisierung], [SEC-Automatisierung], [ORG-Automatisierung] und [SEC-Initiierung] widerspiegelt. Ein weiterer Aspekt stellt das Vertrauen eines IdPs in einen SP dar, wie anhand der Anforderungen [FA-LoT], [ORG-LoT] und [DSA-LoT] zu sehen.

Anschließend wird in Abschnitt 3.3 auf die *Metadaten-Verwaltung* selbst sowie die *Verteilung* der Daten eingegangen, welche die Anforderungen [FA-Metadaten], [FA-Reichweite], [NFA-Koexistenz], [SEC-Metadaten], [ORG-Föderation], [ORG-Metadaten], [ORG-Registrierung] und [ORG-Validierung] direkt umfasst. So müssen die Metadaten gebildet und die Organisation registriert werden. Die Reichweite, Koexistenz und die Bildung mehrerer Föderationen hängen eng mit den jeweiligen Metadaten-Verwaltungen in den einzelnen Föderationen zusammen. Ferner spielt die Metadaten-Verwaltung für die Anforderungen [FA-Automatisierung], [FA-Dynamik], [FA-Föderation] sowie [SEC-Automatisierung] eine Rolle.

Damit eng verknüpft sind die Forschungsansätze zum Thema *Vertrauensaufbau* in Ab-

schnitt 3.4 und *Level of Assurance* in Abschnitt 3.6, vergleiche Anforderungen [FA-LoA] und [ORG-LoA] sowie [FA-LoT], [ORG-LoT] bzw. [DSA-LoT]. Neben der Möglichkeit die Verlässlichkeitsklasse im Protokoll einzubinden, wie in den Abschnitten 3.1 und 3.2 beschrieben, gibt es verschiedene Lösungen in der Praxis sowie Normen, in denen Verlässlichkeitsklassen definiert sind, die in dem Abschnitt 3.6 näher betrachtet werden.

Auf Grund der Defizite in aktuellen Föderationen und Inter-Föderationen werden in Abschnitt 3.5 unterschiedliche Forschungsansätze für die *Interoperabilität* in Föderationen und Inter-Föderationen betrachtet und mit den Anforderungen verglichen. Abschließend erfolgt in Abschnitt 3.7 eine Zusammenfassung und Bewertung.

3.1. FIM-Standards

In diesem Abschnitt werden mit *SAML* und *OAuth* bzw. *OpenID Connect* zwei FIM-Industriestandards vorgestellt und anhand des in Kapitel 2 definierten Kriterienkatalogs bewertet. Während in R&E tendenziell SAML eingesetzt wird, herrscht in anderen Webanwendungen OpenID Connect vor, welches auf OAuth basiert. Zunächst wird der Aufbau der jeweiligen Protokolle beschrieben. Anschließend wird auf organisatorische Aspekte eingegangen, bevor die Gesichtspunkte Lokalisierungsdienst und Level of Assurance aufgezeigt werden. Diese beiden Aspekte werden gesondert analysiert, da sie für die Architektur von besonderer Bedeutung sind. Der Lokalisierungsdienst ist eine Trusted Third Party, durch die SPs und IdPs miteinander verknüpft werden. Dies geschieht in vorhandenen Föderationen statisch und stellt ein Defizit dar, vgl. [FA-Automatisierung]. Ferner ist die Verlässlichkeitsklasse bzw. deren Vergleichbarkeit ein Defizit in aktuellen Föderationen und Inter-Föderationen, vgl. wichtige Anforderung [FA-LoA], wodurch sie in diesem Kapitel näher betrachtet wird. Abschließend werden die Protokolle anhand des im letzten Kapitel erarbeiteten Kriterienkatalogs bewertet.

3.1.1. Security Assertion Markup Language

OASIS ist ein internationales gemeinnütziges Konsortium mit über 5.000 Mitgliedern, welches Standards in beispielsweise den Bereichen Sicherheit, Internet of Things, Cloud Computing und Energie entwickelt und umsetzt. *SAML* wird vom Security Service Technical Committee (SSTC) von OASIS entwickelt und ist ein XML-basiertes Framework für FIM. SAML V1.0 wurde im November 2002 ein OASIS Standard. Die aktuelle Version ist V2.0. SAML ist ein allgemeines Framework, um Informationen über Identitäten, Attribute und Berechtigungen zu kommunizieren, wodurch SAML auf verschiedenen Arten eingesetzt werden kann. Im Bereich R&E sind die Implementierungen Shibboleth und SimpleSAMLphp weit verbreitet, die daher im nächsten Abschnitt näher dargestellt werden. PySAML2 wird zukünftig in der Distribution Ubuntu nativ vorhanden sein, wodurch diese Implementierung ebenfalls beschrieben wird. Zusätzlich wird die kommerzielle Implementierung Active

Directory Federation Services (ADFS) betrachtet.

Aufbau

SAML [SAML2Core] [CKPM05] ist anhand von verschiedenen Standards zu Assertions, Protocols, Bindings und Profiles definiert. Zusätzlich werden die Begriffe Metadaten und Authentication Context verwendet, was insbesondere bei Föderationen wichtig ist. Die Begriffe sind wie folgt beschrieben:

Assertions: Gebündelte Informationen, *Statements*, welche von einer Entität gesendet werden. Diese dienen entweder der Authentifizierung (*Authentication*), enthalten Attributsinformationen oder überliefern die Autorisierungsentscheidung (*Authorization Decision*). Die Struktur einer *Assertion* ist generisch mit allgemeinen Informationen über die sendende Entität und der *Assertion* an sich.

Protocols: Ablauf von Nachrichten, d. h. *Requests* und *Responses*, zwischen Entitäten, beispielsweise für die Authentifizierung oder zum Single Logout.

Bindings: Verknüpfung von SAML Nachrichten zu anderen Übertragungsprotokollen. So beschreibt das *SAML SOAP Binding* wie SAML Nachrichten über Simple Object Access Protocol (SOAP) Nachrichten kommuniziert werden können. *Bindings* werden in [SAML2Bind] [CHK+05] näher beschrieben.

Profiles: Spezifikation von SAML für bestimmte Anwendungsfälle durch Bedingungen oder bzw. und Erweiterungen, z. B. das *Web Browser SSO Profile* für Nutzung von SSO durch Webbrowser. Zusätzlich gibt es *Attribute Profiles*, welche Regeln zur Interpretation von Attributen in *SAML Attribute Assertions* definieren. *Profiles* sind in [SAML2Prof] [HCH+05] näher spezifiziert.

Metadata: Informationen über eine Entität, welche gleichzeitig ein Kommunikationsendpunkt ist [SAML2Meta] [CMPM05].

Authentication Context: Kontext der Authentifizierung [SAMLAC] [KCM+05].

Die Elemente werden im Folgenden näher erläutert.

Assertions

Assertions werden in SAML verwendet, um Aussagen über Subjekte, wie z. B. Benutzer, zu treffen. Laut des SAML Standards [SAML2Core] [CKPM05] kann der Aufbau solcher Assertions drei verschiedene Arten von Statements enthalten, wie bereits im Kapitel 2 beschrieben:

- Das *Authentication Statement* bestätigt die Authentifizierung eines Subjekts, beispielsweise eines Nutzers. Zusätzlich können hier Angaben über den Authentication Context gemacht werden, d. h. wie, wann und von wem das Subjekt authentifiziert wurde.

- Das *Attribute Statement* informiert über die Attribute eines Subjekts, z. B. den Namen des Nutzers.

- Die *Authorization Decision* erteilt die Aussage, ob ein Subjekt autorisiert ist auf eine Ressource zuzugreifen.

Assertions können, wie im Beispiel 3.1 zu sehen, auch verschlüsselt werden, wenn die Nachricht beispielsweise über eine potentiell unsichere Zwischeninstanz übertragen wird.

Der Aussteller dieser Assertion ist in `Issuer` in Zeilen 7 bis 9 angegeben. Das Ziel der Assertion steht in `Destination`, vergleiche Zeile 2. Zusätzlich werden eine Referenz (Zeile 4), eine ID (Zeile 3) sowie eine Signatur (Zeilen 10 bis 22) beschrieben. Das Element `EncryptedAssertion` in Zeile 26 zeigt an, dass die Assertion verschlüsselt ist. Die Informationen werden in `EncryptedData`, Zeilen 27 bis 50, entsprechend der `XML Encryption Syntax and Processing specification` [`XMLEnc`] verschlüsselt. Zusätzlich wird der Schlüssel zur Entschlüsselung in `EncryptedKey`, Zeilen 34 bis 43, angegeben.

```
<saml2p:Response xmlns:saml2p="urn:oasis:names:tc:SAML:2.0:protocol"
    Destination="https://gigamove.rz.rwth-aachen.de/Shibboleth.sso/SAML2/POST"
    ID="_e98b64a44f617abca5297bf577d048f6"
    InResponseTo="_8198ea107d9681e30d962a3631b92e83"
    IssueInstant="2014-09-08T07:25:19.947Z"
    Version="2.0" >
    <saml2:Issuer xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion"
        Format="urn:oasis:names:tc:SAML:2.0:nameid-format:entity"
        >https://idp.lrz.de/idp/shibboleth</saml2:Issuer>
     <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
          ...
         <ds:SignatureValue>PuO7YHWJzJ8UPvHa2wZ0wRCfkrh8LS2wWaqeqyaFc
            ...
        NBPiCL+atswVv1PoojuBeeUDNNs1BbI1IPNDYmqg==</ds:SignatureValue>
        <ds:KeyInfo>
            <ds:X509Data>
              <ds:X509Certificate>MIIFsDCCBJigAwIBAgIEDyCYejANBgkqhkiG9w0BAQU
               ...
              vuutJ8gG4MRSDt0kVopfGsEIT142DrBwPmGvPGmU=</ds:X509Certificate>
            </ds:X509Data>
        </ds:KeyInfo>
    </ds:Signature>
    <saml2p:Status>
        <saml2p:StatusCode Value="urn:oasis:names:tc:SAML:2.0:status:Success"/>
    </saml2p:Status>
    <saml2:EncryptedAssertion xmlns:saml2="urn:oasis:names:tc:SAML:2.0
    :assertion">
        <xenc:EncryptedData xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
            Id="_3007edbe737f4afa9069c7b45ab8b315"
            Type="http://www.w3.org/2001/04/xmlenc#Element" >
            <xenc:EncryptionMethod
                Algorithm="http://www.w3.org/2001/04/xmlenc#aes128-cbc"
                xmlns:xenc="http://www.w3.org/2001/04/xmlenc#" />
                <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
                    <xenc:EncryptedKey Id="_3feed951a03572a8abbf942cc797de29"
                        xmlns:xenc="http://www.w3.org/2001/04/xmlenc#" >
                        ...
                        <xenc:CipherData
                            xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
                            <xenc:CipherValue>EeHyhhO6zXMpNiMCCbA+KIB734PAyOamb
                            ...
                            GzY4+OHQT5d2oENWDhNZ2Qc0ShV2o+cAw==</xenc:CipherValue>
                        </xenc:CipherData>
                    </xenc:EncryptedKey>
                </ds:KeyInfo>
                <xenc:CipherData xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
                    <xenc:CipherValue>6T/lOBDxISClez+dBKRhQbLNl1BVtOil/8/
                    ...
                    QWB1hnoKvhUcigNbyo3D9ycaN6o=</xenc:CipherValue>
                </xenc:CipherData>
        </xenc:EncryptedData>
    </saml2:EncryptedAssertion>
</saml2p:Response>
```

Listing 3.1: Beispiel einer Assertion

Protocols

Durch Protocols werden in [SAML2Core] [CKPM05] die Abläufe von Nachrichten, d. h. von Requests und Responses, festgelegt. Hierfür gibt es folgende Aktionen:

- Authentifizierung,
- Versenden einer oder mehrerer Assertions,
- Empfang von durch Artefakte benötigte Nachrichten,
- Registrierung oder Entfernen eines Name Identifiers sowie
- Single-Logout, d. h. gleichzeitiger Logout bei allen Diensten, an denen das Subjekt angemeldet ist.

Zur Authentifizierung wird ein *AuthnRequest* an den jeweiligen Identity Provider geschickt, wie in Listing 3.2 zu sehen. Dieser beschreibt den Zweck der Authentifizierung, beispielsweise durch `Issuer` in Zeilen 9 bis 11. Zusätzlich kann der IdP durch die Angabe `AllowCreate` im Element `NameIDPolicy` (vgl. Zeile 12) eine neue ID für den Nutzer erstellen, falls er dies zur Erfüllung der Authentifizierung benötigt.

```
1  <samlp:AuthnRequest xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
2          AssertionConsumerServiceURL="https://gigamove.rz.rwth-aachen.de/
       Shibboleth.sso/SAML2/POST"
3          Destination="https://idp.lrz.de/idp/profile/SAML2/Redirect/SSO"
4          ID="_8198ea107d9681e30d962a3631b92e83"
5          IssueInstant="2014-09-08T07:25:05Z"
6          ProtocolBinding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
7          Version="2.0"
8          >
9      <saml:Issuer xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">
10         https://gigamove.rz.rwth-aachen.de/shibboleth
11     </saml:Issuer>
12     <samlp:NameIDPolicy AllowCreate="1" />
13 </samlp:AuthnRequest>
```

Listing 3.2: Beispiel eines Authentication Requests

Die Response des IdPs kann eine oder mehrere Assertions enthalten.

Bindings

SAML *Bindings* geben an, welches Transport-Protokoll für einen Nachrichtenaustausch verwendet werden soll. Das Transport-Protokoll ist dabei unabhängig von SAML. Im Standard [SAML2Bind] [CHK+05] werden die folgenden zur Verfügung stehenden Bindings definiert:

- *HTTP Redirect* verwendet URL-Parameter zum Übermitteln von Nachrichten. Zwar gibt der HTTP-Standard [RFC2616] keine maximale Länge einer URL an, jedoch begrenzen Browser und Server-Software die Länge, wodurch in SAML nur kurze Nachrichten mit einem Binding übertragen werden sollen.

- *HTTP POST* übermittelt Daten, die Base64 verschlüsselt sein müssen.

- *HTTP Artifact* übermittelt Referenzen auf SAML-Nachrichten. Diese Methode kann angewandt werden, wenn der Browser des Nutzers nur HTTP Redirect unterstützt, aber trotzdem lange Nachrichten übertragen werden sollen.

- *SAML SOAP* dient zur Übertragung von SOAP-Nachrichten, welche auf XML basieren.

- *Reverse SOAP* bietet die Möglichkeit in einem SAML Request anzuzeigen, dass der Sender SOAP-Nachrichten empfangen kann.

- *SAML URI* gibt eine URL zu einem SAML-Dokument an, welches per HTTP aufgerufen werden kann.

Am häufigsten werden, insbesondere bei Shibboleth und SimpleSAMLphp, die Bindings HTTP Redirect und HTTP POST verwendet.

Profiles

SAML *Profiles* beschreiben, wie Assertions mit anderen Frameworks und Transport-Protokollen verwendet werden können. Zusätzlich kann durch ein Profile die Anwendung von SAML-Funktionen eingeschränkt oder näher beschrieben werden. Grundlegende Profiles sind im SAML Standard [`SAML2Prof`] [HCH+05] definiert, wie beispielsweise:

- *Web Browser SSO Profile*, welches den Prozess der Authentifizierung über das SAML Authentication Request Protocol mit Hilfe von HTTP Redirect, HTTP POST und HTTP Artifact Bindings beschreibt. Dieses Profile wird bei den meisten Authentifizierungen verwendet.

- *Enhanced Client or Proxy (ECP) Profile* verwendet SOAP und Reverse SOAP Bindings zur Authentifizierung einen Clients.

- Das *Single Logout Profile* ermöglicht es Identity Providern die Session eines bestimmten Benutzers bei mehreren Service Providern gleichzeitig zu beenden.

- Durch das *Name Identifier Management Profile* können Name Identifier verwaltet werden. Name Identifier sind eindeutige Zeichenketten, die zwischen IdP und SP verwendet werden, um einen Benutzer zu identifizieren.

- Das *Artifact Resolution Profile* beschreibt das Vorgehen, ein durch ein HTTP Artifact Binding übermitteltes Artefakt abzufragen.
- Das *Assertion Query/Request Profile* charakterisiert die Abfrage von Attributen über synchrone Bindings, wie zum Beispiel das SOAP Binding.
- Das *Name Identifier Mapping Profile* wird eingesetzt, um zwei Name Identifier eines Benutzers zu verknüpfen.
- Das *SAML Attribute Profile* spezifiziert die Namen von häufig benötigten SAML-Attributen.

Die *Identity Provider Discovery Profiles* und *Identity Assurance Profiles* werden im Folgenden näher beschrieben, da sie für die Architektur von besonderer Bedeutung sind.

Metadaten

Metadaten [`SAML2Meta`] [CMPM05] spezifizieren eine standardisierte Methode, um die für die verwendeten Profiles benötigten Informationen zwischen den beteiligten Kommunikationsendpunkten, d. h. IdP und SP, auszutauschen. Diese Informationen können beispielsweise `Identifier`, unterstützte Methoden für Bindings, Zertifikate und Schlüssel sein. Eine Metadaten-Datei beginnt mit einem der Elemente `<EntityDescriptor>` oder `<EntitiesDescriptors>`, abhängig davon, wie viele Objekte beschrieben werden. Ein `<EntityDescriptor>` muss eine EntityID enthalten, die die Entität eindeutig identifiziert. Innerhalb des Elternelements können unterschiedliche Rollen, `RoleDescriptor`, angegeben werden, wie `SSODescriptor` und `IDPSSODescriptor`. Das Element `<SingleSignOnService>` zeigt an, welcher Endpunkt für die Authentifizierung verwendet werden kann. Jeder IdP muss mindestens ein solches Element spezifizieren, wie auch das LRZ im Beispiel 3.3 ab Zeile 10. Als Erweiterung, Zeile 11 ff, werden hier weitere Informationen, wie der `DisplayName` und die `Description` angegeben. Die verwendeten Schlüssel werden ab Zeile 23 beschrieben, um die Authentizität der Identität des IdPs überprüfen zu können. Ferner werden verwendete Bindings in Zeilen 39 bis 43 beschrieben. Zusätzlich können weitere Informationen als Elemente, wie eine Beschreibung der Organisation (Zeile 46 bis 56), hinzugefügt werden.

```
<EntityDescriptor entityID="https://idp.lrz.de/idp/shibboleth">
  <Extensions>
    <mdrpi:RegistrationInfo registrationAuthority="https://www.aai.dfn.de"
     registrationInstant="2009-05-27T12:36:25Z">
     ...
      <mdrpi:RegistrationPolicy xml:lang="de">https://www.aai.dfn.de/teilnahme/
      </mdrpi:RegistrationPolicy>
    </mdrpi:RegistrationInfo>
  </Extensions>
  <IDPSSODescriptor protocolSupportEnumeration="urn:mace:shibboleth:1.0␣
    urn:oasis:names:tc:SAML:1.1:protocol␣urn:oasis:names:tc:SAML:2.0:protocol">
    <Extensions>
      <saml1md:Scope regexp="false">lrz.de</saml1md:Scope>
      <mdui:UIInfo>
        <mdui:DisplayName xml:lang="de">
                  Leibniz-Rechenzentrum (LRZ)
        </mdui:DisplayName>
        <mdui:Description xml:lang="de">Der LRZ Identity Provider bedient
          Benutzer und Mitarbeiter des Leibniz-Rechenzentrums, insb. auch
          Supercomputing-Kunden</mdui:Description>
         ...
      </mdui:UIInfo>
    </Extensions>
    <KeyDescriptor use="encryption">
      <ds:KeyInfo>
        <ds:KeyName>lrzidp.lrz.de</ds:KeyName>
        <ds:X509Data>
          <ds:X509SubjectName>
            CN=lrzidp.lrz.de,O=Leibniz-Rechenzentrum,L=Muenchen,ST=Bayern,C=DE
          </ds:X509SubjectName>
          <ds:X509Certificate>
            ...
          </ds:X509Certificate>
        </ds:X509Data>
      </ds:KeyInfo>
    </KeyDescriptor>
    <KeyDescriptor use="signing">
    ...
    </KeyDescriptor>
    <ArtifactResolutionService Binding="urn:oasis:names:tc:SAML:1.0:bindings:SOAP-
    binding" Location="https://idp.lrz.de:8443/idp/profile/SAML1/SOAP/
    ArtifactResolution" index="1"/>
    ...
    <SingleSignOnService Binding="urn:mace:shibboleth:1.0:profiles:AuthnRequest"
    Location="https://idp.lrz.de/idp/profile/Shibboleth/SSO"/>
    <SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
    Location="https://idp.lrz.de/idp/profile/SAML2/POST/SSO"/>
    <SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"
     Location="https://idp.lrz.de/idp/profile/SAML2/Redirect/SSO"/>
  </IDPSSODescriptor>
  ...
  <Organization>
    <OrganizationName xml:lang="de">e38</OrganizationName>
    <OrganizationDisplayName xml:lang="de">Leibniz-Rechenzentrum der Bayerischen
    Akademie der Wissenschaften</OrganizationDisplayName>
    <OrganizationURL xml:lang="de">http://www.lrz.de</OrganizationURL>
    ...
  </Organization>
  <ContactPerson contactType="administrative">
    <GivenName>Ralf</GivenName>
    <SurName>Ebner</SurName>
    <EmailAddress>mailto:shibboleth@lrz.de</EmailAddress>
  </ContactPerson>
  ...
</EntityDescriptor>
```

Listing 3.3: Beispiel von Metadaten am LRZ

Organisatorische Aspekte

In SAML (vgl. SAML Technical Overview [RHPM08]) existiert eine Asserting Party, die eine SAML Assertion erstellt, und eine Relying Party, die die SAML Assertion verwendet. Durch das *Web Browser SSO Profile* [SAML2Prof] [HCH+05] wurden zusätzlich die Rollen Identity Provider, Service Provider und Attribute Authority eingeführt, wie in Kapitel 2 beschrieben. Metadaten und die darin enthaltenen Informationen, wie verwendete Server und das Kommunikationsprotokoll, bilden die Basis zur Kommunikation zwischen IdP und SP. IdP und SP gehören somit einer gemeinsamen Föderation an. Bei der Kommunikation zwischen Identity Provider und Service Provider kann durch Verschlüsselung, zum Beispiel durch die Verwendung von SSL 3.0, und signierte Nachrichten die Sicherheit erhöht werden. Zur Sicherung der Privatsphäre ist beispielsweise die Anwendung des *Authentication Contexts* möglich, ebenso kann der *User Consent* abgefragt werden.

Discovery Service

Mit Hilfe des *Identity Provider Discovery Profiles* ist ein Service Provider in der Lage den Identity Provider eines Nutzers für das *Web Browser SSO Profile* zu bestimmen. Der Name des IdPs wird dabei, beim *Identity Provider Discovery Profile* innerhalb der SAML V2.0 Profiles specification [SAML2Prof] [HCH+05], in einem Browser-Cookie in einer gemeinsamen Domain, *Common Domain Cookie* genannt, gespeichert. Eine 2008 veröffentlichte Erweiterung dieses Profiles mit dem Namen *Identity Provider Discovery Service Protocol and Profile* [SAMLIdPDisc] [LCWC08], spezifiziert ein generisches Browser-basiertes Protokoll für einen zentralen Lokalisierungsdienst. Im Gegensatz zum ersten Profile sieht es keine gemeinsame Domain mehr vor, da diese zwischen unterschiedlichen Organisationen schwer zu verwalten ist. Stattdessen wird ein Parameter in der Antwortnachricht des Discovery Services eingeführt. Zusätzlich kann die Information in einem HTTP Cookie gespeichert werden. Der generelle Ablauf enthält drei Schritte:

Schritt 1: Der Service Provider leitet den User Agent, beispielsweise einen Browser, mit bestimmten Parametern weiter zum Lokalisierungsdienst.

Schritt 2: Der Discovery Service interagiert mit dem Nutzer über den User Agent, damit ein oder mehrere passende IdPs ausgewählt werden.

Schritt 3: Der Discovery Service leitet den User Agent zum SP weiter mit dem ausgewählten IdP.

Als Parameter muss die `entityID`, d. h. die einmalige ID des SPs, angegeben werden. Weitere Parameter sind optional und werden im R&E-Umfeld selten verwendet. Um Angriffe durch bösartige Entitäten zu erschweren, können zusätzlich Metadaten eingesetzt werden, um die Anzahl der erlaubten Entitäten zu beschränken. Das bedeutet, dass nur Entitäten den Lokalisierungsdienst nutzen können, deren Metadaten bekannt sind.

SAML V2.0 Identity Assurance Profiles

Das von OASIS spezifizierte *SAML V2.0 Identity Assurance Profiles* [SAMLIAP] [KHM+10] beschreibt die Verwendung des *SAML Authentication Contexts* in Authentifizierungsnachrichten für die Angabe des LoAs. Zusätzlich wird ein Attribute definiert, welches in die SAML Metadaten hinzugefügt werden kann, um die LoA Zertifikate des IdPs aufzuzählen. Der *SAML Authentication Context* benötigt ein XML Schema, welches bestimmte Kriterien für eine gegebene *Authentication Context Class* definiert. Es wird empfohlen jedes Level durch eine eigene Klasse darzustellen, wobei jedes LoA durch folgende Elemente repräsentiert wird:

- URI, welche die *Authentication Context Class* definiert,
- Body des Schemas, welcher auf eine Referenz zur externen Dokumentation der LoA verweist.

Somit muss für ein neues Schema folgendes definiert werden:

- Für jedes LoA wird eine eigene URI benötigt.
- Jedes LoA benötigt ein passendes Dokument mit einer dazugehörigen URL.
- Zudem muss für jedes LoA ein XML-Schema definiert werden, welches
  - die *Base Authentication Context Types Schema* neu definiert,
  - die URI aus Schritt 1 als Schema `targetNamespace` setzt,
  - die `AuthnContextDeclarationBaseType` so beschränkt, dass nur ein einziges `GoverningAgreements` Element zugelassen ist und
  - die `governingAgreementRef` auf die URL aus Schritt 2 festlegt.

Das Schema für eine niedrige Verlässlichkeitsklasse LoA 1 für das LRZ kann wie das nachfolgende Beispiel 3.4 aussehen.

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <xs:schema targetNamespace="http://loa.lrz.de/assurance/loa1"
3   xmlns:xs="http://www.w3.org/2001/XMLSchema"
4   xmlns="http://loa.lrz.de/assurance/loa1"
5   finalDefault="extension" blockDefault="substitution"
6   version="2.0">
7    <xs:redefine schemaLocation="saml-schema-authn-context-types-2.0.xsd">
8      <xs:annotation>
9        <xs:documentation>
10         Class identifier:
11         http://loa.lrz.de/assurance/loa1
12         Definiert LoA1
13       </xs:documentation>
14     </xs:annotation>
15
16     <xs:complexType name="AuthnContextDeclarationBaseType">
17       <xs:complexContent>
18         <xs:restriction base="AuthnContextDeclarationBaseType">
19           <xs:sequence>
20             <xs:element ref="GoverningAgreements"/>
21           </xs:sequence>
22           <xs:attribute name="ID" type="xs:ID" use="optional"/>
23         </xs:restriction>
24       </xs:complexContent>
25     </xs:complexType>
26     <xs:complexType name="GoverningAgreementRefType">
27       <xs:complexContent>
28         <xs:restriction base="GoverningAgreementRefType">
29           <xs:attribute name="governingAgreementRef" type="xs:anyURI"
30            fixed="http://loa.lrz.de/assurance.pdf#section1"
31            use="required"/>
32         </xs:restriction>
33       </xs:complexContent>
34     </xs:complexType>
35
36   </xs:redefine>
37 </xs:schema>
```

Listing 3.4: Schema für LoA am LRZ

In Zeile 2 wird die URI für das LoA 1 als `targetNamespace` gesetzt. In der folgenden Zeile 3 wird das Schema neu definiert. Die Dokumentation findet sich in den Zeilen 9 bis 13. Das für die LoA benötigte Dokument findet sich in Zeile 30 wieder, wobei hier ein Portable Document Format (PDF) für alle Verlässlichkeitsklassen mit je einem Abschnitt pro LoA als `governingAgreeementRef` festgelegt ist. Zusätzlich enthält die `AuthnContextDeclaration BaseType` nur ein `GoverningAgreements` Element in Zeile 16. Für die SAML Metadaten wurde ferner ein Attribut definiert, welches dafür verwendet werden soll den Level of Assurance auszudrücken. Im Elternelement `<md:EntityDescriptor>` wird das XML-Element `<saml:Attribute>` genutzt, um die LoA zu beschreiben, wie aus dem nachfolgenden Beispiel 3.5 ersichtlich.

```
1  <md:EntityDescriptor
2    xmlns="urn:oasis:names:tc:SAML:2.0:metadata"
3    xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
4    xmlns:attr="urn:oasis:names:tc:SAML:metadata:attribute"
5    xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
6    entityID="https://www.lrz.de/SAML">
7     <md:Extensions>
8       <attr:EntityAttributes>
9         <saml:Attribute
10          NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
11          Name="urn:oasis:names:tc:SAML:attribute:assurance-certification">
12            <saml:AttributeValue>
13               http://loa.lrz.de/assurance/loa1
14            </saml:AttributeValue>
15              </saml:Attribute>
16      </attr:EntityAttributes>
17    </md:Extensions>
18     ...
19  </md:EntityDescriptor>
```

Listing 3.5: XML-Element in SAML Metadaten für LoA

Der `EntityDescriptor` in Zeile 1 enthält die EntityID des LRZs. Als Extension wird ab Zeile 7 ein Attribute für die LoA definiert, welches den Wert `http://loa.lrz.de/assurance/loa1` (Zeile 13) hat.

Bewertung

Bei der Beurteilung von SAML ist zu berücksichtigen, dass dieser Standard sehr allgemein gefasst wurde und daher keine Bewertung zu organisatorischen Anforderungen und Datenschutzanforderungen möglich ist. Zusätzlich gibt es keine Beschreibung der Integration ([FA-Integration]) in die bestehenden Systeme. Die folgenden Aspekte hängen von der Implementierung ab und werden im SAML Standard nicht oder nur teilweise behandelt:

- Usability ([NFA-Usability]),
- Monitoring ([FA-Monitoring]),
- Lokalisierung ([FA-Lokalisierung]); hier wird ein grundsätzlicher Lokalisierungsdienst beschrieben, jedoch ist die Ausprägung von der Implementierung abhängig
- Fehlermanagement ([FA-Fehlermanagement]), wobei ein grundlegendes Fehlermanagement vorhanden ist
- Automatisierung ([FA-Automatisierung]),
- Systemsicherheit ([SEC-Systemsicherheit]), wodurch [SEC-Multilateral] auch nicht ge-

geben ist und

- Konfiguration ([FA-Konfiguration]).

Die damit zusammenhängenden Anforderungen können folglich ebenso wenig betrachtet werden. Föderationen ([FA-Föderation]), Schema ([FA-Schema]) und Dynamik ([FA-Dynamik]) werden im SAML Standard nicht direkt behandelt und hängen sowohl von der Implementierung als auch von der Organisation ab.

Die übrigen essentiellen und wichtigen Anforderungen werden wie folgt bewertet:

- +: Anforderung vollständig erfüllt.
- o: Anforderung teilweise erfüllt bzw. hängt von der Implementierung ab.
- -: Anforderung nicht erfüllt.

Daraus ergibt sich folgendes Bild (vgl. Tabelle 3.1).

Anforderung	Priorität	Bewertung	Anforderung	Priorität	Bewertung
Funktionale Anforderungen					
[FA-Aktualisierung]	2	+	[FA-Metadaten]	2	o
[FA-Datenkategorisierung]	1	+	[FA-Pull&Push]	1	+
[FA-Initiierung]	2	-	[FA-Realisierbarkeit]	1	o
[FA-Langlebigkeit]	1	+	[FA-Reichweite]	2	o
[FA-LoA]	2	o	[FA-Rollen]	2	+
[FA-LoT]	2	-			
Nichtfunktionale technische Anforderungen					
[NFA-Dokumentation]	1	+	[NFA-Performanz]	2	o
[NFA-Implementierungsunabhängigkeit]	2	+	[NFA-Protokollunabhängigkeit]	2	-
[NFA-Koexistenz]	1	+	[NFA-Skalierbarkeit]	1	o
[NFA-OpenSource]	1	+			
Sicherheitsanforderungen					
[SEC-Authentifizierung]	1	o	[SEC-LoA]	2	o
[SEC-Datenübertragung]	1	o	[SEC-LoT]	2	-

Tabelle 3.1.: Bewertung von SAML

Bei den wünschenswerten Anforderungen werden lediglich [FA-Kontext] und [NFA-Portabilität] vollständig gewährleistet. SAML bietet somit ein gutes Grundgerüst, was allerdings stark von der Implementierung und der Organisation abhängt.

3.1.2. OAuth und OpenID Connect

Blaine Cook startete 2006 die Entwicklung an OAuth. Im Oktober 2007 wurde die erste Spezifikation von OAuth Core 1.0 veröffentlicht. Seitdem wird OAuth in einer Working Group der IETF weiter entwickelt. OAuth 2.0 ist ein Framework, welches durch `RFC6749` [Har12] sowie `[RFC6750]` [JH12] spezifiziert ist. Das Protokoll erlaubt es Nutzern Dritten, wie Service

Providern und Consumern, Zugriff auf private Ressourcen zu geben. Die standardisierten Nachrichten basieren auf JavaScript Object Notation (JSON) zum Datenaustausch und dem Übertragungsprotokoll HTTP. OAuth stellt somit einen Standard für Entwickler zu Verfügung, damit diese ihre Dienste über eine API an Nutzer bereitstellen, ohne dass diese ihr Passwort offen legen müssen. Im Gegensatz dazu stellt OpenID sicher, dass der Benutzer tatsächlich derjenige ist, den er vorgibt zu sein. OpenID Connect ist die dritte Version von OpenID, die von der OpenID Foundation entwickelt wurde. Im Gegensatz zu den Vorgängerversionen baut OpenID auf OAuth 2.0 auf. Sie fügt somit OAuth 2.0 einen Identity Layer zur Authentifizierung hinzu. Durch diesen ist es Service Providern möglich die Identität des Benutzers über eine Authentifizierung durch einen Authorization Server zu verifizieren und grundsätzliche Benutzerinformationen zu erhalten. OpenID Connect 1.0 besteht aus insgesamt sechs Spezifikationen:

- *OpenID Connect Core* definiert die Hauptfunktionalität, d. h. Authentifizierung auf Basis von OAuth 2.0 und die Verwendung von Behauptungen (*Claims*) zur Kommunikation von Benutzerinformationen.
- *OpenID Connect Discovery* ist optional und beschreibt den Lokalisierungsdienst.
- Die optionale *Dynamic Registration* spezifiziert, wie sich Clients dynamisch beim OpenID Provider (OP) registrieren können.
- *Session Management* ist ebenfalls eine optionale Spezifikation.
- *OAuth 2.0 Multiple Response Types* definiert spezielle OAuth 2.0 Antworttypen (*Response Types*).
- *OAuth 2.0 Form Post Response Mode* ist optional und beschreibt wie *Authorization Response Parameter* über HTTP POST übermittelt werden.

OpenID Connect wird hauptsächlich bei Webanwendungen und teils in der Wirtschaft eingesetzt, zum Beispiel durch Google, Microsoft, Deutsche Telekom und Ping Identity.

Aufbau

Wichtige Elemente in OpenID Connect sind laut OpenID Connect Core [SBJ+14] die Authentifizierung und der Claim, was gleichbedeutend mit einer Information zur Authentifizierung ist. Ein Service Provider kann OpenID Connect verwenden, indem er den Scope-Wert `openid` zur Authentifizierungsanfrage hinzufügt. Dieser Request leitet den Browser durch ein HTTP Redirect zum OP um. Diese bestätigt die erfolgreiche Authentifizierung durch einen Authentifizierungscode. Die Relying Party (RP) wandelt den Authentifizierungscode in ein Token um. Daraufhin sendet der OpenID Provider das `id_token` im Body des Responses, welches Attribute und Informationen über den Authentifizierungsprozess enthält. Die Daten sind in Form eines JSON Web Token (JWT) enthalten und können bei Bedarf digital signiert

und verschlüsselt werden. Die Informationen zur Authentifizierung sehen beispielsweise wie im folgenden Listing 3.6 aus.

```
{
  "iss": "https://accounts.lrz.de",
  "sub": "723423467677",
  "aud": "SDFGHJJKLSDFZTREDFGHJ",
  "exp": 1388700360000,
  "iat": 1388696760000,
  "auth_time": 1388696760000
  "acr": "urn:lrz:de:level-1"
}
```

Listing 3.6: JSON Web Token zur Authentifizierung

Jedes dieser Felder enthält einen Claim zum Benutzer oder zum Authentifizierungsprozess:

iss. Zuerst wird der Issuer (`iss`), also der IdP, genannt.

sub. Das Subject (`sub`) enthält den Benutzer.

aud. Audience (`aud`) identifiziert den legalen Empfänger des Tokens.

iat und exp. Issued At (`iat`) und Expires At (`exp`) geben Auskunft über die Lebenszeit des Tokens.

auth_time. Authentication Time (`auth_time`) gibt den Zeitpunkt an, an dem der Nutzer authentifiziert wurde.

acr. Die Authentication Context Class Reference (`acr`) beschreibt das Verfahren für die Authentifizierung des Benutzers.

nonce. `Nonce` ist ein String um die Client Session mit einem ID Token zu assoziieren.

amr. Authentication Methods References (`amr`) ist ein optionaler Claim zur Identifizierung der verwendeten Authentifizierung, wie beispielsweise Multifaktor-Authentifizierung und Passwort.

azp. Die Authorized party (`azp`) ist ein weiterer optionaler Claim und beschreibt die OAuth 2.0 Client ID der Entität, die den ID Token herausgegeben hat.

Nach der erfolgreichen Auswertung des Web Tokens kann der Benutzer den Dienst nutzen. Falls der OP weitere Benutzerinformationen benötigt, kann er diese bei der Authentifizierungsanfrage anfordern. Dafür hat er folgende vordefinierte Sets von Claims zur Auswahl:

profile: `name`, `family_name`, `given_name`, `middle_name`, `nickname`, `preferred_username`, `profile`, `picture`, `website`, `gender`, `birthdate`, `zoneinfo`, `locale` und `updated_at`.

email: `email`, `email_verified`.

address: `formatted`, `street_address`, `locality`, `region`, `postal_code`, `country`.

phone: `phone_number`, `phone_number_verified`.

Es können auch einzelne Claims, wie `name`, abgefragt werden. Gleichzeitig ist es möglich, dass der OP nicht jeden Wert eines Sets an den Service Provider sendet.

Organisatorische Aspekte

Die Kommunikation in OpenID Connect geschieht zwischen Endnutzer, Relying Party und OpenID Provider, welcher auch als IdP bezeichnet wird. Die Entitäten gehören keinem Zusammenschluss, wie einer Föderation, an. Neben der HTTP Authentifizierung können verschiedene Sicherheitsmaßnahmen, wie Verschlüsselung und Signaturen, verwendet werden.

AccountChooser und WebFinger

OpenID Connect verwendet aktuell WebFinger um den Identity Provider des Benutzers zu Lokalisieren, jedoch wird verstärkt auf den AccountChooser [Bra13] gesetzt, weswegen dieser im Folgenden zunächst beschrieben wird. Das Ziel des *AccountChoosers* ist es dem Benutzer eine Liste seiner Benutzerkonten mittels der jeweils zugehörigen E-Mail-Adressen anzuzeigen, damit dieser den für die Anwendung geeigneten auswählen kann. Dafür sollen Anwendungen, d. h. Relying Parties, die Möglichkeit haben aus dem AccountChooser Daten heraus zu lesen und zusätzliche Benutzerkonten hinzuzufügen. Die Informationen, die im AccountChooser hinterlegt werden, bestehen aus:

- zumindest einer E-Mail-Adresse, möglichst auch aus
- dem Namen des mit der E-Mail-Adresse verknüpften Benutzers und
- der URI, von der ein Foto des Benutzers geladen werden kann.
- Zusätzlich kann der Name des Identity Provider gespeichert und angezeigt werden.

Um den AccountChooser verwenden zu können, muss die Anwendung auf die Javascript-Datei `ac.js` von `accountchooser.com` verweisen, was durch das Element `<script>` in der Hypertext Markup Language (HTML)-Seite der Anwendung geschieht. Wenn der Benutzer sich bei einer Anwendung einloggen will, die AccountChooser verwendet, wird er durch den Browser zur AccountChooser Seite weitergeleitet, die ihm eine Auswahl an bereits gespeicherten Benutzerkonten anbietet, vergleiche Abbildung 3.2. Zusätzlich hat er die Möglichkeit weitere IdPs hinzuzufügen. Wenn der Benutzer ein bereits gespeichertes Benutzerkonto aus-

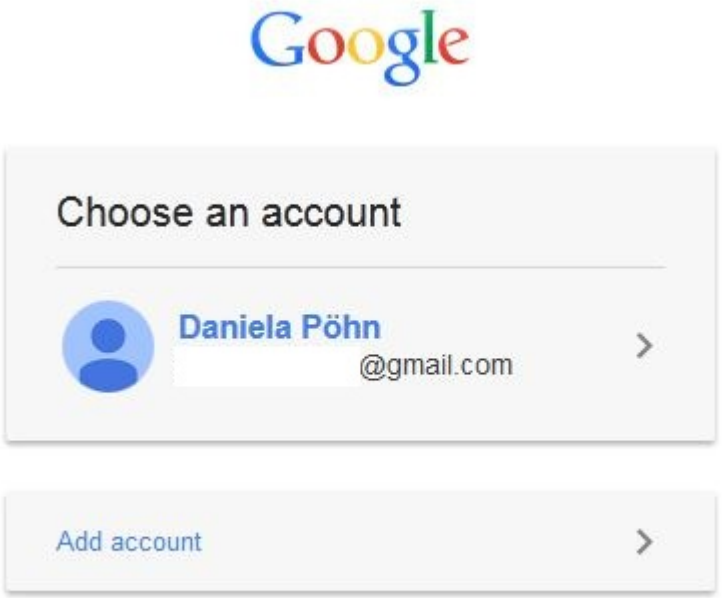

Abbildung 3.2.: AccountChooser bei Google

wählt, muss `ac.js` den Browser zur `userStatusUrl` weiterleiten. Der IdP antwortet mit einem `HTTP Status Code 200`, was `ok` bedeutet, und dem Resultat der Identitätsprüfung im Body, beschrieben durch JSON. Wenn der Nutzer erfolgreich authentifiziert wurde und er der Weitergabe seiner Daten zustimmt (vgl. Abbildung 3.3), soll der IdP zu einer Seite weiterleiten, die einen Link zu `ac.js` sowie das `storeAccount Configuration Argument` mit Informationen über den Benutzer enthält. Daraufhin überprüft `ac.js`, ob das angegebene Benutzerkonto mit einem bereits hinterlegten übereinstimmt und ob aktualisierte Informationen vorhanden sind. Schließlich leitet `ac.js` den Browser zur `homeURL` weiter und der Nutzer kann den Dienst verwenden.

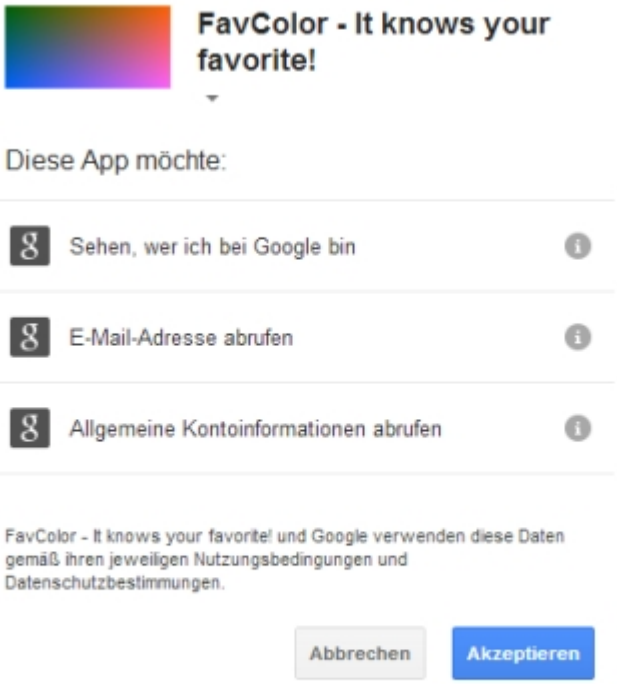

Abbildung 3.3.: User Consent beim AccountChooser

```
1  GET /.well-known/webfinger
2      ?resource=acct%3Adi34koj%40lrz.de
3      &rel=http%3A%2F%2Fopenid.net%2Fspecs%2Fconnect%2F1.0%2Fissuer
4      HTTP/1.1
5    Host: example.com
6
7    HTTP/1.1 200 OK
8    Content-Type: application/jrd+json
9    {
10     "subject": "acct:di34koj@lrz.de",
11     "links":
12       [
13        {
14         "rel": "http://openid.net/specs/connect/1.0/issuer",
15         "href": "https://server.lrz.de"
16        }
17       ]
18   }
```

Listing 3.7: Beispiel eines WebFinger-Requests, aus [SBJJ14]

WebFinger [JSJS13] verwendet das entsprechende Protokoll WebFinger, welches als `[RFC-7033]` spezifiziert ist. Die Relying Party verwendet für die Suche nach dem OpenID Provider einen `Identifier names resource`. Die `resource` kann beispielsweise die Struktur

- einer E-Mail-Adresse (`acct:di34koj@lrz.de`),
- einer URL (`https://lrz/di34koj`),
- von Hostname und Port (`lrz.de:8080`) oder
- der `acct` URI Syntax (`acct:di34koj%40oid.lrz@mnm.lrz.de`)

verwenden. Daraus wird durch standardisierte Normalisierungsregeln, wie in [SBJJ14] beschrieben, der Host ermittelt, auf dem der WebFinger Service betrieben wird. Der Host erstellt daraufhin einen Request mit dem Identifier, wie in Listing 3.7 zu sehen. WebFinger verwendet für den Request an den Host den Pfad `/.well-known/webfinger` mit den Parametern `res` für resource und `rel`, der die Art des gesuchten Dienstes angibt. `rel` ist für OpenID Connect mit `http://openid.net/specs/connect/1.0/issuer` vorbelegt.

```
HTTP/1.1 200 OK
  Content-Type: application/json

  {
   "issuer":
     "https://server.lrz.de",
   "authorization_endpoint":
     "https://server.lrz.de/connect/authorize",
   "token_endpoint":
     "https://server.lrz.de/connect/token",
   "token_endpoint_auth_methods_supported":
     ["client_secret_basic", "private_key_jwt"],
   "token_endpoint_auth_signing_alg_values_supported":
     ["RS256", "ES256"],
   "userinfo_endpoint":
     "https://server.lrz.de/connect/userinfo",
   "check_session_iframe":
     "https://server.lrz.de/connect/check_session",
   "end_session_endpoint":
     "https://server.lrz.de/connect/end_session",
   "jwks_uri":
     "https://server.lrz.de/jwks.json",
   "registration_endpoint":
     "https://server.lrz.de/connect/register",
   "scopes_supported":
     ["openid", "profile", "email", "address",
      "phone", "offline_access"],
   "response_types_supported":
     ["code", "code id_token", "id_token", "token id_token"],
   "subject_types_supported":
     ["public", "pairwise"],
        ...
   "request_object_signing_alg_values_supported":
     ["none", "RS256", "ES256"],
   "display_values_supported":
     ["page", "popup"],
   "claim_types_supported":
     ["normal", "distributed"],
   "claims_supported":
     ["sub", "iss", "auth_time", "acr",
      "name", "given_name", "family_name", "nickname",
      "email", "email_verified", "locale", "zoneinfo",
      "http://lrz.de/claims/groups"],
   "claims_parameter_supported":
     true,
   "service_documentation":
     "http://server.lrz.de/connect/service_documentation.html",
   "ui_locales_supported":
     ["de-DE", "en-GB"]
  }
```

Listing 3.8: Beispiel einer WebFinger-Response, aus [SBJJ14]

Die Antwort des WebFinger Servers enthält, wie in Listing 3.8 zu sehen, eine Liste mit einem Verweis an den Ort des benutzten OPs sowie mögliche Claims, Verschlüsselung, Dokumentation, Sprache und weitere Informationen.

PAPE und Authentication Context Reference

OpenID bietet, im Gegensatz zu SAML, grundsätzlich kein Trust-Modell, so dass die Relying Parties selbst entscheiden müssen, welchem Provider sie vertrauen. Um das *Accept-all-comers-Prinzip* einzuschränken, wurde OpenID Provider Authentication Policy Extension (PAPE) [RJB+08] entwickelt. PAPE bietet einen Mechanismus an, damit die RP bestimmte Policies zur Authentifizierung vom OP anfragen kann und damit OpenID Provider das Level der Authentifizierung kommunizieren.

Im Rahmen des inzwischen obsoleten Yadis Discovery Workflows können OpenID Provider optional unterstützte Policies zur Authentifizierung anmerken. Das bringt Relying Parties in die Lage zwischen verschiedenen OPs zu entscheiden. Die Relying Party fügt verschiedene Parameter in der Authentifizierung hinzu, um ihre Präferenzen und Anforderungen zu beschreiben.

Nachdem die Spezifikationen von Yadis und OpenID 2.0 als obsolet gelten, sollen bei Nutzung von OpenID Connect hierfür Claims im Token verwendet werden. Anstelle der Parameter für Namen und des Typ des LoA spezifiziert beispielsweise der Claim zur *Authentication Context Class Reference* (`acr`) den Level of Assurance. Der Wert kann entweder eine absolute URI oder ein in `[RFC6711]` registrierter Name sein. Der String des `acr` kann den Wert 0 annehmen, wenn die Authentifizierung nicht den Anforderungen von International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC) 29115 Level 1 entspricht. Dies ist der Fall, wenn kein Vertrauen besteht, dass der Nutzer über konsekutive Authentifizierungen derselbe ist, was in Abschnitt 3.6 näher beschrieben wird.

Durch User-Managed Access, welches bereits in Abschnitt 2.6 vorgestellt wurde, soll es dem Nutzer möglich sein eine feingranulare Auswahl an zu versendeten Attributen zu treffen. Die Entscheidung, ob und wie der User Consent überprüft wird, liegt aktuell bei der Relying Party und variiert dadurch.

Bewertung

Äquivalent zu SAML beschreibt OpenID Connect nicht die Integration in die bestehende Umgebung, wodurch [FA-Konnektor] und [FA-Integration] nicht gegeben sind. [FA-Attributswahl] und [FA-Monitoring] sind nur mit der Erweiterung UMA möglich, während das Schema sehr eingeschränkt ist. Die Fehlermeldungen haben ein festgelegtes Schema, jedoch kann jeder Provider seine Fehlermeldung selbst wählen. OpenID Connect wird insgesamt wie folgt be-

wertet:

Anforderung	Priorität	Bewertung	Anforderung	Priorität	Bewertung
Funktionale Anforderungen					
[FA-Aktualisierung]	2	+	[FA-Konnektor]	2	-
[FA-Attributswahl]	2	-	[FA-Langlebigkeit]	1	o
[FA-Automatisierung]	2	o	[FA-LoA]	2	+
[FA-Datenkategorisierung]	1	-	[FA-Lokalisierung]	1	+
[FA-Dynamik]	2	+	[FA-LoT]	2	-
[FA-Fehlermanagement]	2	+	[FA-Metadaten]	2	o
[FA-Föderation]	1	-	[FA-Monitoring]	2	-
[FA-Grenzüberschreitend]	1	+	[FA-Pull&Push]	1	+
[FA-Initiierung]	2	o	[FA-Realisierbarkeit]	1	o
[FA-Integration]	1	-	[FA-Reichweite]	2	o
[FA-Interaktion]	1	+	[FA-Rollen]	2	o
[FA-Konfiguration]	1	o	[FA-Schema]	2	o
Nichtfunktionale technische Anforderungen					
[NFA-Dokumentation]	1	+	[NFA-Performanz]	2	+
[NFA-Implementierungsunabhängigkeit]	2	o	[NFA-Protokollunabhängigkeit]	2	-
[NFA-Koexistenz]	1	o	[NFA-Skalierbarkeit]	1	+
[NFA-OpenSource]	1	+	[NFA-Usability]	2	+
Sicherheitsanforderungen					
[SEC-ARPs]	1	-	[SEC-Integration]	2	-
[SEC-Auditing]	2	o	[SEC-LoA]	2	+
[SEC-Authentifizierung]	1	+	[SEC-LoT]	2	-
[SEC-Automatisierung]	2	-	[SEC-Multilateral]	1	-
[SEC-Datenübertragung]	1	o	[SEC-Systemsicherheit]	1	-
[SEC-Initiierung]	2	o			
Datenschutzanforderungen					
[DSA-ARPs]	1	-	[DSA-LoT]	2	-
[DSA-Datenschutz]	1	-	[DSA-Selbstbestimmung]	1	o
[DSA-Interaktion]	2	+	[DSA-Zustimmung]	2	+

Tabelle 3.2.: Bewertung von OpenID Connect

Bei den wünschenswerten Anforderungen werden [FA-SelfAsserted], [FA-Identitätswahl] und [NFA-Portabilität] vollständig erfüllt, während [FA-Homeless] als teilweise erfüllt zählt, da es diese Kategorie in OpenID Connect nicht gibt, aber ein IdP hierfür aufgesetzt werden kann. Insgesamt zeigt sich, dass OpenID Connect dynamisch Vertrauen aufbauen kann und benutzerfreundlich ist, jedoch fehlen Funktionalitäten, wie die Kategorisierung von AAs, die für R&E notwendig sind. Ferner ist im Standard nicht direkt beschrieben, wie der Datenschutz eingehalten werden soll.

3.2. SAML Implementierungen

In dem folgenden Abschnitt werden drei OpenSource SAML Implementierungen, Shibboleth, SimpleSAMLphp und PySAML2, erläutert. Den Implementierungen gemeinsam sind der technische Vertrauensaufbau mittels Metadaten, der auf den Protokollen zu SAML aufbaut, sowie der Datenschutz. Zusätzlich wird die kommerzielle Implementierung ADFS von Microsoft analysiert. Darauf folgt eine genauere Betrachtung der Implementierungen anhand der folgenden Aspekte.

Aufbau: Technischer Aufbau der Implementierung anhand der Produkte.

Lokalisierungsdienst: Trusted Third Party, durch die eine Verbindung zwischen IdP und SP zustande kommt, vgl. essentielle Anforderung [FA-Lokalisierung]. Die Implementierungen basieren auf dem im vorherigen Abschnitt beschriebenen Identity Provider Discovery Profile.

Attribute Handling: Damit Service Provider die Benutzerinformationen im geforderten Format bekommen, müssen die Attribute gegebenenfalls umgewandelt werden, vgl. wichtige Anforderung [FA-Schema]. Zusätzlich ist die Filterung der Attribute wichtig, damit beispielsweise kein SP nur intern verwendete Informationen erhält, vgl. essentielle Anforderung [SEC-ARPs].

User Consent: Ein elementarer Aspekt des Datenschutzes ist die Zustimmung des Nutzers zur Übertragung der Daten an einen SP, vgl. essentielle Anforderung [FA-Interaktion]. Zusätzlich bietet diese Anwendung die Möglichkeit [FA-Attributswahl] zu realisieren.

Abschließend werden die Implementierungen anhand des erstellten Kriterienkatalogs bewertet.

3.2.1. Datenschutz und Trust

Der technische Vertrauensaufbau in den nationalen Föderationen und in den Inter-Föderationen eduGAIN und KALRMAR2 basieren auf den Ansatz von Dynamic SAML bzw. Distributed Dynamic SAML von Patrick Harding, Leif Johannson und Nate Klingenstein [HJK08]. Dynamic SAML ermöglicht es Metadaten zu vertrauen, die mit dem privaten Schlüssel signiert sind. Der öffentliche Schlüssel des X.509 Zertifikats ist in den Metadaten enthalten, um die Signatur überprüfen zu können. Das Zertifikat wird dafür von einer vertrauenswürdigen CA erstellt. Umgesetzt wurde dieses Konzept im [`SAML V2.0 Metadata Interoperability Profile (MetaIOP)`] [LCC09b], welches durch [`Interoperable SAML 2.0 Web Browser SSO Deployment Profile (SAML2int)`] [SCM+15] näher spezifiziert wird. [`MetaIOP`] beschreibt die Anforderungen, um allgemeine Informationen zu Entitäten sowie Vertrauen mit Metadaten auszutauschen, wobei der Austausch an sich nicht näher beschrieben wird. Dabei ist es wichtig, dass

- Informationen über Schlüssel oder Kerberos vorhanden sind,
- abgelaufene Schlüssel entfernt werden,
- kompromittierte Schlüssel ausgetauscht werden,
- die Entität, die die Metadaten erhält, entsprechend den Informationen konfiguriert werden kann und

- Metadaten, die über unsichere Kanäle versendet werden, verschlüsselt sind und nach einer gewissen Zeit ablaufen, d. h. dass entweder ein `validUntil` oder `cacheDuration` gesetzt ist.

[`SAML2int`] stellt zusätzliche Bedingungen, die für den Einsatz in Produktivumgebungen gelten, die diesem Profile folgen. Beispielsweise

- muss die Entität entweder im `IDPSSODescriptor` oder im `SPSSODescriptor` näher beschrieben werden.
- müssen, wenn Attribute über Metadaten ausgetauscht werden, sie ein `NameFormat` in Form einer URI verwenden.
- muss für den Authentication Request ein HTTP-Redirect Binding eingesetzt werden.
- müssen Response Nachrichten über das HTTP-POST Binding kommuniziert werden.

Die erste Implementierung dieses Ansatzes gab es in SimpleSAMLphp, jedoch können inzwischen auch andere SAML-Implementierungen, wie Shibboleth, hierfür verwendet werden. Während die Konzepte und Profiles die Form des Austausches von Metadaten nicht behandeln, hat sich in der Praxis die Aggregation und Vorab-Verteilung der Metadaten etabliert. Je nach Policy der Föderation und der Entität bedeutet das Vorhandensein der Metadaten nicht unbedingt Vertrauen im Sinne von *behavioural trust*, wie in Abschnitt 2.4.2 aufgezeigt. Dieses Vertrauen muss in den meisten Fällen trotzdem manuell hergestellt werden, was als Nachteil zu werten ist. Ferner sind die Größe der Datei sowie die Verteilung vorab durch die Limitierung der Kooperationen als suboptimal zu bezeichnen.

3.2.2. Shibboleth

Das Projekt Shibboleth [Shi15] wurde ursprünglich von einer Initiative der Internet2 Middleware 2000 gestartet. Anschließend wurde zusammen mit der OASIS SAML Working Group an der Software entwickelt. Shibboleth ist eine Open Source Implementierung von SAML, die unter der Apache Software License herausgegeben wird und aus den folgenden Produkten besteht:

Centralized Discovery Service: Zentraler Lokalisierungsdienst, der angepasst werden kann. Um den zentralen Dienst betreiben zu können, müssen die Metadaten der Teilnehmer gesammelt und aggregiert werden. Diesen Dienst nutzen beispielsweise Föderationen.

Embedded Discovery Service: Der eingebettete Lokalisierungsdienst läuft parallel zur SP-Software und zeigt nur diejenigen IdPs an, die der Service Provider kennt. Das Aussehen des Lokalisierungsdienstes kann durch die Webtechnologien HTML, Javascript und Cascading Style Sheets (CSS) leicht eingefügt und an das eigene Corporate Design

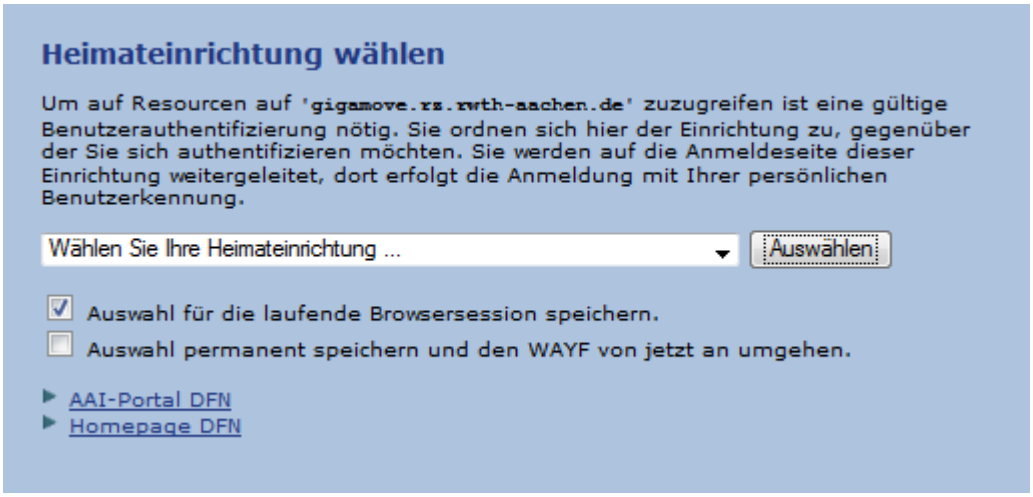

Abbildung 3.4.: Discovery Service eines Dienstes in der DFN-AAI

angepasst werden.

Metadata Aggregator: Tool für die Kommandozeile, welches auch durch einen Webdienst betrieben werden kann. Der Metadata Aggregator verwendet und veröffentlicht Metadaten. Die Software kann hierfür Metadaten aus verschiedenen Quellen einlesen, verifizieren, filtern und transformieren. Das Tool wird ebenfalls von Föderationen eingesetzt.

Identity Provider: Software für den Identity Provider. Sie ermöglicht es Benutzer zu authentifizieren, SAML Authentication Requests zu akzeptieren, Benutzerinformationen aus den lokalen Quellen, wie beispielsweise LDAP, zu sammeln und herauszugeben, Policies umzusetzen und verschlüsselte Benutzerinformationen an den SP zu übertragen. Die Software ist in Java geschrieben.

Service Provider: Das Gegenstück dazu ist die Software für den SP, die vor allem in C++ implementiert wurde.

Im Folgenden werden der Discovery Service, die Konfigurationsdateien Attribute Resolver und Attribute Filter auf Seiten des IdPs sowie der User Consent näher betrachtet.

Shibboleth Discovery Service

Wie bereits im vorherigen Abschnitt erwähnt, gibt es einen zentralen und einen eingebetteten Lokalisierungsdienst. Beide dienen dazu, den IdP des Nutzers herauszufinden. Dazu wählt der Benutzer manuell seinen Identity Provider aus einer Liste aus, wie in Abbildung 3.4 zu sehen. Dadurch hat der Benutzer eine freie IdP-Wahl, wodurch eine Person verschiedene digitale

Identitäten verwenden kann. Während die Liste beim zentralen Lokalisierungsdienst aus IdPs besteht, die in den Föderationsmetadaten enthalten sind, kann der Service Provider durch Verwendung des eingebetteten Lokalisierungsdienstes die auswählbaren IdPs auf diejenigen beschränken oder erweitern, denen er vertraut. Durch die Auflistung möglicher IdPs wird gleichzeitig ein grundlegender Schutz erreicht, da die Identity Provider vertrauenswürdig sind. Der grundlegende Ablauf sieht verschiedene Phasen vor:

Nutzer möchte SP nutzen und wird weitergeleitet. Der Benutzer möchte einen Dienst bei einem Service Provider nutzen. Wenn bereits eine gültige Session für den Benutzer existiert, kann dieser sofort den Dienst nutzen. Ansonsten, nachdem der SP den IdP des Benutzers nicht kennt, wird der Client des Nutzers zum Discovery Service über ein HTTP Redirect weitergeleitet.

Nutzer wählt IdP aus. Der Nutzer wählt auf der HTML-Seite des Lokalisierungsdienstes aus einer Liste von möglichen IdPs seine Heimatorganisation aus und wird über ein Redirect zum SP zurückgeleitet, der einen Authentication Request zum Identity Provider des Nutzers schickt.

Nutzer authentifiziert sich. Der IdP leitet den Benutzer zur Website weiter, wo er sich authentifizieren kann. Der Benutzer authentifiziert sich, meist über die Eingabe von Benutzername und Passwort, bei seiner Heimatorganisation. Wenn die Authentifizierung erfolgreich war, wird der Nutzer zum gewählten Service Provider zurück verwiesen. Der Identity Provider sendet eine Assertion mit den gefilterten Attributen des Nutzers zum Service Provider.

Nutzer bekommt Zugriff zum Dienst. Nach der erfolgreichen Authentifizierung kann der SP basierend auf den Benutzerinformationen entscheiden, ob dem Benutzer der Zugriff zum Dienst gewährt wird oder nicht. Da der Nutzer bereits authentifiziert ist, kann dieser nun, ohne weitere Zwischenschritte, den Dienst nutzen. Nachdem der IdP nur die notwendigen Attribute versendet, wird der Datenschutz eingehalten.

Zwei Aspekte sind beim Betrieb eines Lokalisierungsdienstes zu beachten: Verfügbarkeit und Sicherheit. Die Verfügbarkeit des Discovery Services ist essentiell, nachdem ansonsten der Nutzer keinen IdP auswählen und somit keinen externen Dienst nutzen kann. Ferner muss der Lokalisierungsdienst vertrauenswürdig sein, da es theoretisch die Möglichkeit gibt, dass der Lokalisierungsdienst einen Benutzer während eines Phishing-Angriffs auf die Seite eines Angreifers weiterleitet.

Attribute Handling

Damit der Benutzer einen Dienst beim Service Provider verwenden kann, muss der IdP dem SP benötigte Benutzerinformationen senden. Dafür sind verschiedene Schritte notwendig, wie in Abbildung 3.5 zu sehen. Der IdP liest durch einen `DataConnector` die Attribute aus

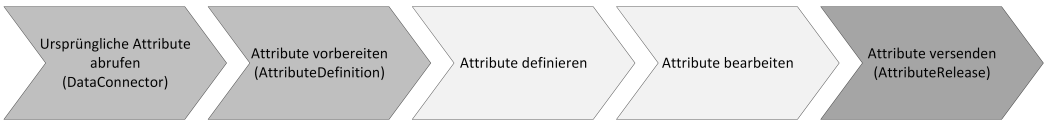

Abbildung 3.5.: Ablaufdiagramm Attribute Handling

dem lokalen Datenbestand, wie LDAP, aus. Die rohen Attribute können anschließend den Anforderungen des SPs entsprechend in der `AttributeDefinition` umgewandelt werden, hauptsächlich durch:

- *Renaming:* Umbenennen von Attributen, beispielsweise von der internen Bezeichnung `gecos` zu `displayName`.
- *Transforming:* Attribut in ein anderes Format transformieren; dies ist zum Beispiel bei unterschiedlichen Datumsformaten notwendig.
- *Merging und Splitting:* Attribut aus mehreren Teilen zusammen bauen bzw. aus einem Attribut eine Teilmenge extrahieren.

Diese Umwandlung findet bei Shibboleth in der `attribute-resolver.xml` statt, wo auch die Datenquelle definiert wird. Wie die Dateiendung bereits aussagt, findet die Umformung durch XML und bei Bedarf durch zusätzliche Skripte in European Computer Manufacturers Association (ECMA) Script statt. Das nachfolgende Beispiel 3.9 zeigt eine einfache Umbenennung.

```
1 <resolver:AttributeDefinition id="displayName" xsi:type="Simple" xmlns="
      urn:mace:shibboleth:2.0:resolver:ad" sourceAttributeID="gecos">
2   <resolver:Dependency ref="simauth" />
3   <resolver:AttributeEncoder xsi:type="SAML1String" xmlns="
      urn:mace:shibboleth:2.0:attribute:encoder" name="urn:mace:dir:attribute-
      def:displayName" />
4   <resolver:AttributeEncoder xsi:type="SAML2String" xmlns="
      urn:mace:shibboleth:2.0:attribute:encoder" name="urn:oid:2
      .16.840.1.113730.3.1.241" friendlyName="displayName" />
5 </resolver:AttributeDefinition>
```

Listing 3.9: Umbenennung eines Attributes in attribute-resolver.xml

In diesem Beispiel wird in Zeile 1 ein internes Attribut `gecos` in `displayName` umbenannt. Das interne Attribut wird aus dem lokalen LDAP durch Zeile 2 geladen und anschließend in eine URN (vgl. Zeile 3 und 4) verschlüsselt.

Zur Umwandlung stehen in Shibboleth verschiedene vorgefertigte Transformationen zur Verfügung, u. a. die Folgenden.

- Eine spezielle Definition erleichtert das *Mappen* von Attributen, d. h. Umbenennen.
- Zusätzlich können *regular expression (regex)* für spezielle Definitionen mit regulären Ausdrücken verwendet werden, beispielsweise um ein Ausgangsattribut zu splitten.
- *Template Attribute Definition* benutzt die Velocity Template Sprache, um verschiedene Attribute zu kombinieren (*mergen*).
- *Scoped Attribute Definition* wird verwendet, um Attribute mit einfachem Namen und einen bestimmten Geltungsbereich zu definieren.
- *Principal Name Attribute Definition* bildet den Principal Name.
- Basierend auf der verwendeten Methode zur Authentifizierung wird ein *Principal Authentication Method Attribute Definition* kreiert.
- Falls keine vorgefertigte Transformation passt, kann das gewünschte Attribute mit *ECMA Skripten* gebildet werden.

Die transformierten Attribute werden in Assertions verpackt und entsprechend vorbereitet, beispielsweise verschlüsselt, wie im Listing 3.9 in Zeile 3 und 4 geschehen. Dies passiert bevor die Attribute in der `attribute-filter.xml` gefiltert werden. Durch den Attribute Filter legt der IdP fest, welche Attribute an den SP gesendet werden, was aus datenschutzrechtlichen Gründen relevant ist. Die Regeln können pro Föderation, pro Service Provider, pro Benutzer und pro Attribut festgelegt werden. Ebenso sind allgemeine Regeln möglich, wie im folgenden Beispiel 3.10 zu sehen. Hier werden Attribute bestimmt, die jeder SP erhält.

```
1 <!-- Default ARP -->
2 <AttributeFilterPolicy id="DefaultARP">
3   <PolicyRequirementRule xsi:type="basic:ANY" />
4   <AttributeRule attributeID="transientId">
5     <PermitValueRule xsi:type="basic:ANY" />
6   </AttributeRule>
7   <AttributeRule attributeID="eduPersonTargetedID">
8     <PermitValueRule xsi:type="basic:ANY" />
9   </AttributeRule>
10  <AttributeRule attributeID="eduPersonAffiliation">
11    <PermitValueRule xsi:type="basic:ANY" />
12  </AttributeRule>
13  <AttributeRule attributeID="eduPersonScopedAffiliation">
14    <PermitValueRule xsi:type="basic:ANY" />
15  </AttributeRule>
16 </AttributeFilterPolicy>
```

Listing 3.10: Allgemeine Regeln in attribute-filter.xml

In Zeile 2 und 3 wird festgelegt, dass diese Filter für alle SPs gelten. Im Folgenden werden die Attribute `transientId` (Zeile 4), `eduPersonTargetedID` (Ziele 7), `eduPersonAffiliation`

(Zeile 10) und `eduPersonScopedAffiliation` (Zeile 13) zu dieser Regel hinzugefügt, so dass diese Attribute jeder Service Provider erhält.

Das folgende Beispiel 3.11 zeigt eine Eingrenzung auf diejenigen SPs, die sich an den Code of Conduct der Inter-Föderation eduGAIN halten, der als Entity Category in den Metadaten enthalten ist.

In Zeile 1 wird die ID (`releaseToCoC`) gesetzt. In den Zeilen 2-5 erfolgt nun die Einschränkung. So wird nach der Entity Category mit dem Namen `http://macedir.org/entity-category` und dem Wert `http://www.geant.net/uri/dataprotection-code-of-conduct/v1` in den Metadaten gesucht. Nur wenn das Ergebnis exakt diesem Wert entspricht, werden die Attribute `eduPersonPrincipleName`, `displayName`, `mail`, `uid`, `givenName` und `sn` gesendet. Dies ist auch nur der Fall, wenn diese Attribute als benötigt (`required`) in den Metadaten des SPs enthalten sind.

Nach der Filterung werden die übrig gebliebenen Attribute per HTTP-POST an den Service Provider gesendet.

```
1  <AttributeFilterPolicy id="releaseToCoC">
2    <PolicyRequirementRule
3     xsi:type="saml:AttributeRequesterEntityAttributeExactMatch"
4     attributeName="http://macedir.org/entity-category"
5     attributeValue="http://www.geant.net/uri/dataprotection-code-of-conduct/v1"
        />
6    <AttributeRule attributeID="eduPersonPrincipalName">
7      <PermitValueRule xsi:type="saml:AttributeInMetadata"
8          onlyIfRequired="true"/>
9    </AttributeRule>
10   <AttributeRule attributeID="displayName">
11     <PermitValueRule xsi:type="saml:AttributeInMetadata"
12         onlyIfRequired="true"/>
13   </AttributeRule>
14   <AttributeRule attributeID="mail">
15     <PermitValueRule xsi:type="saml:AttributeInMetadata"
16         onlyIfRequired="true"/>
17   </AttributeRule>
18   <AttributeRule attributeID="uid">
19     <PermitValueRule xsi:type="saml:AttributeInMetadata"
20         onlyIfRequired="true"/>
21   </AttributeRule>
22   <AttributeRule attributeID="givenName">
23     <PermitValueRule xsi:type="saml:AttributeInMetadata"
24         onlyIfRequired="true"/>
25   </AttributeRule>
26   <AttributeRule attributeID="sn">
27     <PermitValueRule xsi:type="saml:AttributeInMetadata"
28         onlyIfRequired="true"/>
29   </AttributeRule>
30 </AttributeFilterPolicy>
```

Listing 3.11: Eingrenzung der SPs auf die Entity Category Code of Conduct

Abbildung 3.6.: uApprove User Consent

User Consent

Damit der Benutzer nach dem Vertrauensaufbau seine Zustimmung darüber ausdrücken kann, dass der Identity Provider bestimmte Benutzerinformationen an den Service Provider sendet, können unterschiedliche Erweiterungen eingefügt werden. In der Inter-Föderation eduGAIN ist das von der SWITCHaai entwickelte Tool uApprove [SWI14] für Shibboleth Identity Provider 2.x am weitesten verbreitet. Es kann auch in Shibboleth IdP 3.x eingesetzt werden, während Shibboleth IdP 3.x ein eigenes Consent-Tool mitbringt. Die Erweiterung sorgt dafür, dass

- der Nutzer über den Versand von Attributen informiert wird. Dies geschieht bei der ersten Nutzung eines Dienstes oder wenn sich die abgefragten Daten geändert haben.
- der Identity Provider ein Tool bekommt, um den Datenschutz einzuhalten, indem er vor dem Versand um die Zustimmung des Nutzers bittet.
- der IdP den Nutzer zusätzlich nach seiner Zustimmung zu bestimmten Benutzungsbedingungen fragen kann.
- der IdP weiß, wann ein bestimmter Nutzer seine Zustimmung zum Versand der Attribute zu einem bestimmten Service Provider gegeben hat.

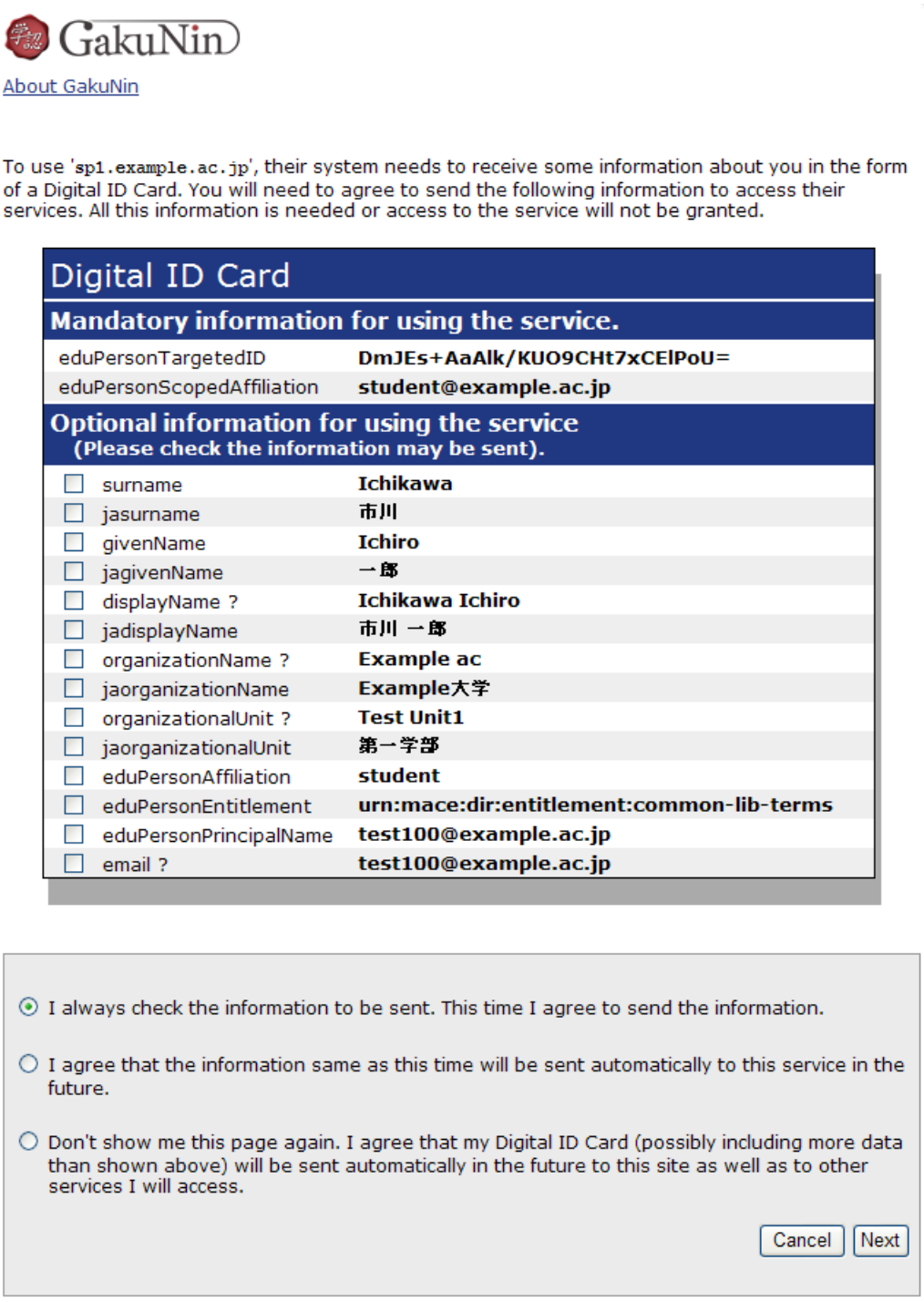

Abbildung 3.7.: uApprove.jp User Consent [Gak14]

Die Abfrage nach der Zustimmung des Benutzers geschieht nach einem vorgegebenen Entscheidungsmuster. So wird der übliche IdP-Workflow fortgesetzt, wenn der SP gesperrt ist oder sich die Attribute nicht geändert haben. Wenn hingegen der Service Provider unbekannt ist, sich die Benutzungsbedingungen oder Attribute geändert haben oder der Service Provider vom Nutzer zum ersten Mal verwendet wird, werden die Daten an die Erweiterung weitergeleitet und die Zustimmung des Nutzers, wie in Abbildung 3.6 zu sehen, abgefragt. Dabei ist nur eine Zustimmung oder Ablehnung möglich.

Die Erweiterung uApprove.jp [OYN+12] von der japanischen NREN-Föderation GakuNin ermöglicht es dem Nutzer über den Versand von optionalen Attributen selbst zu entscheiden. Dazu werden die Metadaten des SPs nach optionalen Attributen, d. h. `isRequired="false"`, durchsucht. Zusätzlich bietet diese Shibboleth-Erweiterung die Möglichkeit über zukünftige Aktionen zu entscheiden. Beispielsweise kann der Benutzer auch beim nächsten Besuch des SPs über seine Zustimmung gefragt werden oder die gewählten Attribute zukünftig von allen Service Providern abrufen lassen, wie in Abbildung 3.7 dargestellt. Diese Informationen werden in einer Datenbank oder im Filesystem gespeichert. Beim Service Provider kann der Nutzer durch die Erweiterung einsehen, welche Attribute an ihn gesandt wurden.

Bewertung

Shibboleth erfüllt alle essentiellen Anforderungen zumindest teilweise, die meisten sogar vollständig, wie in Tabelle 3.3 zu sehen. Nachdem die ARPs zwar feingranular konfiguriert werden können, aber es keine direkte Schnittstelle zu Datenschutzregeln oder eine derartige Überprüfung gibt, werden [DSA-ARPs] und [DSA-Datenschutz] mit teilweise erfüllt gewertet. Ferner hängt die Skalierbarkeit ([NFA-Skalierbarkeit]), die multilaterale Sicherheit ([SEC-Multilateral]), die Registrierung ([ORG-Registrierung]) und Validierung ([ORG-Validierung]) stark von den Föderationsmodellen ab und ist daher schwierig zu bewerten.

Insgesamt ist Shibboleth sehr gut einsatzfähig bei Betrachtung der essentiellen und wichtigen Anforderungen. Bei den empfehlenswerten Anforderungen erfüllt Shibboleth

- [FA-Kontext] und
- [NFA-Portabilität] vollständig sowie
- [FA-Homeless],
- [FA-Identitätswahl],
- [SEC-Kontext],
- [SEC-Metadaten],
- [ORG-Metadaten] und

Anforderung	Priorität	Bewertung	Anforderung	Priorität	Bewertung
Funktionale Anforderungen					
[FA-Aktualisierung]	2	+	[FA-Konnektor]	2	+
[FA-Attributswahl]	2	o	[FA-Langlebigkeit]	1	+
[FA-Automatisierung]	2	-	[FA-LoA]	2	+
[FA-Datenkategorisierung]	1	+	[FA-Lokalisierung]	1	+
[FA-Dynamik]	2	o	[FA-LoT]	2	-
[FA-Fehlermanagement]	2	o	[FA-Metadaten]	2	+
[FA-Föderation]	1	+	[FA-Monitoring]	2	-
[FA-Grenzüberschreitend]	1	+	[FA-Pull&Push]	1	+
[FA-Initiierung]	2	-	[FA-Realisierbarkeit]	1	+
[FA-Integration]	1	+	[FA-Reichweite]	2	o
[FA-Interaktion]	1	+	[FA-Rollen]	2	+
[FA-Konfiguration]	1	+	[FA-Schema]	2	o
Nichtfunktionale technische Anforderungen					
[NFA-Dokumentation]	1	+	[NFA-Performanz]	2	+
[NFA-Implementierungsunabhängigkeit]	2	o	[NFA-Protokollunabhängigkeit]	2	o
[NFA-Koexistenz]	1	+	[NFA-Skalierbarkeit]	1	o
[NFA-OpenSource]	1	+	[NFA-Usability]	2	o
Sicherheitsanforderungen					
[SEC-ARPs]	1	+	[SEC-Integration]	2	+
[SEC-Auditing]	2	+	[SEC-LoA]	2	+
[SEC-Authentifizierung]	1	+	[SEC-LoT]	2	-
[SEC-Automatisierung]	2	-	[SEC-Multilateral]	1	o
[SEC-Datenübertragung]	1	+	[SEC-Systemsicherheit]	1	+
[SEC-Initiierung]	2	-			
Organisatorische Anforderungen					
[ORG-Automatisierung]	2	-	[ORG-Realisierbarkeit]	1	+
[ORG-Föderation]	2	+	[ORG-Registrierung]	1	o
[ORG-Konfiguration]	2	o	[ORG-Schema]	2	o
[ORG-LoA]	2	o	[ORG-Supportprozesse]	2	-
[ORG-LoT]	2	-	[ORG-Validierung]	1	o
[ORG-Migration]	2	+			
Datenschutzanforderungen					
[DSA-ARPs]	1	o	[DSA-LoT]	2	-
[DSA-Datenschutz]	1	o	[DSA-Selbstbestimmung]	1	+
[DSA-Interaktion]	2	o	[DSA-Zustimmung]	2	+

Tabelle 3.3.: Bewertung von Shibboleth

- [DSA-CoCo] teilweise.

Hier zeigt sich, dass neben einzelnen wichtigen Anforderungen auch Verbesserungsbedarf bei wünschenswerten Anforderungen besteht. Die Protokollunabhängigkeit besteht nach aktuellem Stand noch nicht, soll aber in Version 3.x eingebaut werden.

3.2.3. SimpleSAMLphp

SimpleSAMLphp [UNI14] ist eine SAML-Implementierung in Hypertext Preprocessor (PHP). Das Projekt wird von dem norwegischen NREN UNINETT geleitet und setzt bei der Weiterentwicklung auf die Open Source Gemeinde. SimpleSAMLphp ist, im Gegensatz zu Shibboleth, ein einzelnes Produkt, welches sowohl für IdPs als auch für SPs eingesetzt werden kann. Es wurde hauptsächlich für SAML 2.0 [`SAML2Core`] [CKPM05] entwickelt, unterstützt jedoch weitere Protokolle und Standards, wie Shibboleth 1.3, OpenID und OAuth. Durch eine Extension API ist es möglich, Erweiterungen selbst zu programmieren. Im Folgenden wird auf den Lokalisierungsdienst DiscoJuice, Attribute Handling und das Consent Module eingegangen.

DiscoJuice

Der Lokalisierungsdienst *DiscoJuice* ist eine Implementierung des IdP Discovery Protocols [LCWC08] durch UNINETT in Javascript und PHP. Ebenso wie beim Shibboleth Discovery Service gibt es für DiscoJuice verschiedene Varianten:

- DiscoJuice als Embedded IdP Selector Popup in der Anwendung, hinzugefügt durch das Javascript Framework jQuery. Der Lokalisierungsdienst erscheint in einem Popup-Fenster, welches durch die Anwendung aufgerufen wird.
- Service Provider IdP Discovery Service ist äquivalent zum Embedded Discovery Service bei Shibboleth.
- Zentraler IdP Discovery Service für Föderationen.
- Globaler IdP Discovery Service, der mehrere Föderationen umschließen kann.

Der Ablauf ist prinzipiell derselbe wie beim Shibboleth Discovery Service. Jedoch wird nicht eine statische Liste an vertrauenswürdigen IdPs angezeigt, sondern die Position des Nutzers durch die HTML5 Geo-location API ausgelesen und passende IdPs angezeigt, wie in Abbildung 3.8 zu sehen. Dabei wird gleichzeitig das aktuelle Land identifiziert. Zusätzlich kann der Nutzer seinen Identity Provider suchen, vergleiche Abbildung 3.9. Der vom Benutzer ausgewählte IdP wird anschließend als Cookie gesetzt. Alternativ kann die Wahl des IdPs auch beim DiscoJuice gespeichert werden.

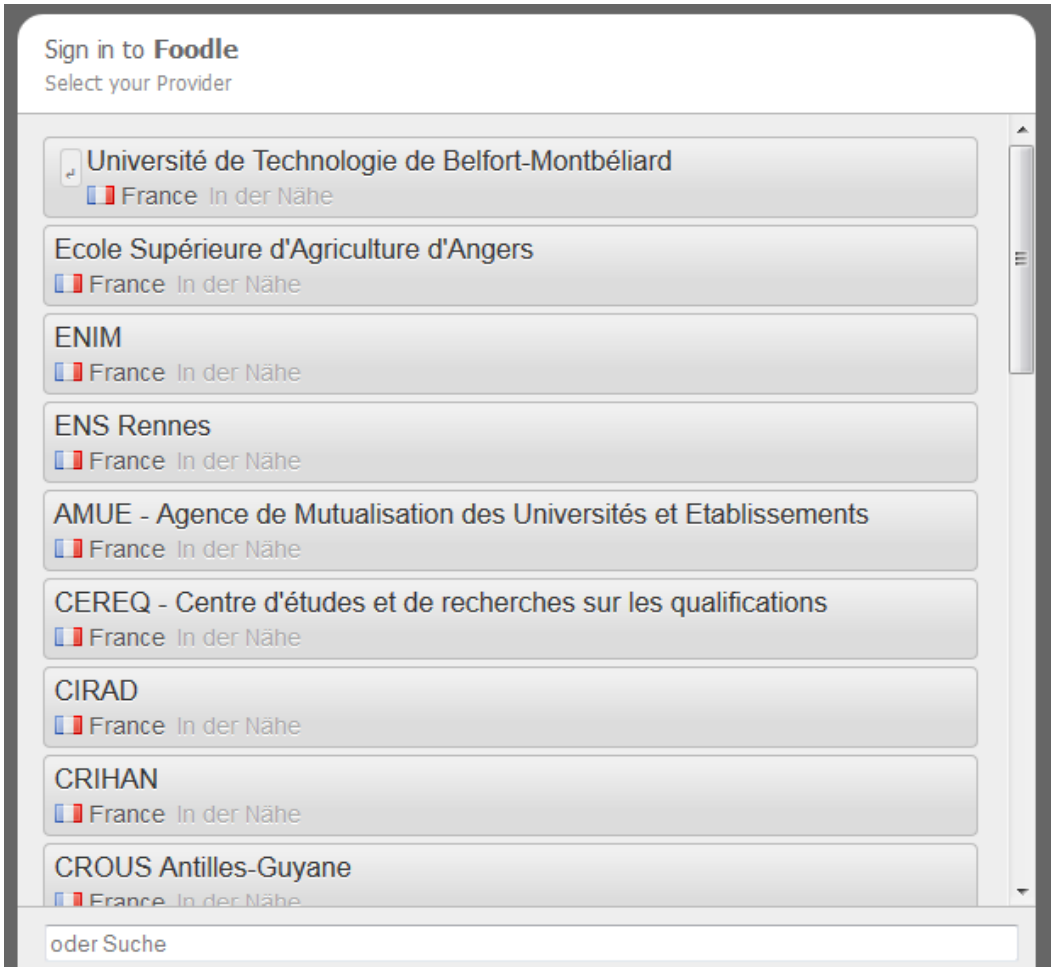

Abbildung 3.8.: Lokalisierungsdienst von DiscoJuice

Attribute Handling

Bevor ein Identity Provider Attribute an einen Service Provider schicken kann, muss der Identity Provider den Service Provider kennen. Dies geschieht, indem die Metadaten mit der URI zur `entityID` und den URLs zu `AssertionConsumerService` und `SingleLogoutService` in `metadata/saml20-sp-remote.php` bzw. `metadata/shib13-sp-remote.php` gespeichert werden, vgl. Listing 3.12.

Abbildung 3.9.: Suchdienst von DiscoJuice

```
1 <?php
2 $metadata['https://sp.lrz.de/simplesaml/module.php/saml/
3 sp/metadata.php/default-sp'] = array(
4     'AssertionConsumerService' => 'https://sp.lrz.de/simplesaml/module.php/saml/
        sp/saml2-acs.php/default-sp',
5     'SingleLogoutService'      => 'https://sp.lrz.de/simplesaml/module.php/saml/
        sp/saml2-logout.php/default-sp',
6 );
```

Listing 3.12: Speicherung der Metadaten eines SPs

In diesem Beispiel werden für den Service Provider mit der `entityID` `https://sp.lrz.de/simplesaml/module.php/saml/sp/metadata.php/default-sp` (Zeile 2 und 3) der `Asser-`

`tionConsumerService` (Zeile 4) und der `SingleLogoutService` (Zeile 5) bestimmt.

Wenn die Metadaten als XML vorliegen, müssen sie erst von SimpleSAMLphp durch das Tool `/admin/metadata-converter.php` konvertiert werden. Dies ist in der Regel für die Metadaten der Föderation und Inter-Föderation nötig. Attribute Handling geschieht anschließend anhand von Authentication Processing Filtern. Durch eine Vielzahl an vorgefertigten Filtern und die Möglichkeit selbst Filter zu definieren gibt es viele Optionen, beispielsweise

- Attribute zu filtern, die an den SP gesendet werden,
- Attribute durch Umbenennung und Merging zu modifizieren und
- den User Consent abzufragen, wie im folgenden Abschnitt näher beschrieben.

An insgesamt fünf Stellen können Authentication Processing Filter konfiguriert werden:

- Global für die gesamte Organisation sowie all ihren Kooperationen in `config.php`.
- Aus SP-Sicht für einen speziellen SP in `authsources.php`. Dies ist nötig, wenn eine Organisation mehrere SPs betreibt.
- Aus SP-Sicht für einen speziellen IdP in `saml20-idp-remote` oder `shib13-idp-remote`.
- Aus IdP-Sicht für einen speziellen IdP in `saml20-idp-hosted` oder `shib13-idp-hosted`. Dies wird benötigt, wenn eine Organisation, wie das LRZ, mehrere IdPs betreibt.
- Aus IdP-Sicht für einen speziellen SP in `saml20-sp-remote` oder `shib13-sp-remote`.

Um einen Filter zu konfigurieren, müssen Klassendefinition und Priorität bekannt sein. Die Prioritäten der Filter bestimmen die Reihenfolge der Abarbeitung. Aktuell werden 24 verschiedene Filter mitgeliefert, wobei folgende insbesondere für das Attribute Handling interessant sind:

- `core:AttributeAdd`: Fügt Attribute der Antwort (`Response`) hinzu.
- `core:AttributeAlter`: Sucht und ersetzt Werte von Attributen.
- `core:AttributeLimit`: Begrenzt die Attribute in der Response.
- `core:AttributeMap`: Benennt Attribute um.
- `core:AttributeRealm`: Erstellt Attribute durch die Hilfe des Benutzers.
- `core:PHP`: Modifiziert Attribute durch PHP Code.
- `core:ScopeAttribute`: Fügt Gültigkeitsbereich zu Attributen hinzu.

- `core:ScopeFromAttribute`: Erstellt ein neues Attribut basierend auf den Gültigkeitsbereich eines anderen Attributes.

Das Umbenennen von Attributen geschieht entweder durch eine eigenen Konfigurationsdatei oder der Angabe der Attribute, die umbenannt werden sollen, als zusätzliche Attribute in der eigentlichen Konfigurationsdatei, wie im folgenden Beispiel 3.13 zu sehen. Hier werden `mail` (Zeile 4), `uid` (Zeile 5) und `cn` (Zeile 6) aus den rechts stehenden Attributen gebildet. Die Art der Umformung steht in Zeile 3, `core:AttributeMap`.

```
1 'authproc' => array(
2         50 => array(
3                 'class' => 'core:AttributeMap',
4                 'mail' => 'email',
5                 'uid' => 'user',
6                 'cn' => array('name', 'displayName'),
7         ),
8 ),
```

Listing 3.13: Umbennung von Attributen in SimpleSAMLphp

Attribute, die nicht durch einen geeigneten Filter erstellt werden können, müssen durch PHP gebildet werden. Im folgenden Beispiel 3.14 wird aus der `uid` (Zeile 6) die E-Mail-Adresse des Nutzers (Zeile 7 und 8) generiert.

```
1  10 => array(
2    'class' => 'core:PHP'
3    'code' => 'if(empty($attributes["uid"])){
4  ␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣throw␣new␣Exception("Missing␣uid␣attribute.");
5  ␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣}
6  ␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣$uid␣=␣$attributes["uid"][0];
7  ␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣$mail␣=␣$uid."@lrz.de";
8  ␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣$attributes["mail"]␣=␣array($mail);
9  ␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣',
10 ),
```

Listing 3.14: Bildung der E-Mail-Adresse mit PHP

Die so gebildeten Attribute werden per HTTP POST an den Service Provider gesendet. Eine Freigabe pro Entity Category ist zur Zeit jedoch nicht möglich.

Consent Module

Das Consent Module in SimpleSAMLphp ist als Authentication Processing Filter implementiert. Dadurch kann es durch die globale Datei `config.php` konfiguriert werden. Damit die Zustimmung des Nutzers abgefragt werden kann, muss das Modul aktiviert werden. Der Nutzer Consent kann auf zwei Arten gespeichert werden: Cookie oder Datenbank. Dabei werden

jeweils die wichtigsten Daten, wie ID des Services, Attribute, gehashte ID des Nutzers und Nutzungsdatum, gespeichert, wie beim Erstellen der Tabelle `consent` in der Datenbank im Beispiel 3.15 zu sehen.

```
1 CREATE TABLE consent (
2          consent_date TIMESTAMP NOT NULL,
3          usage_date TIMESTAMP NOT NULL,
4          hashed_user_id VARCHAR(80) NOT NULL,
5          service_id VARCHAR(255) NOT NULL,
6          attribute VARCHAR(80) NOT NULL,
7          UNIQUE (hashed_user_id, service_id)
8 );
```

Listing 3.15: Erstellung der Datenbank für Consent Module

Das Aussehen der Consent Seite kann jede Organisation selbst anpassen. Jedoch ist es dem Benutzer nur möglich der Weitergabe seiner Daten zuzustimmen oder diese abzulehnen.

Bewertung

Nachdem SimpleSAMLphp (SSp) als Alternative für Shibboleth für Föderationen der NRENs implementiert wurde, sind die Funktionalitäten sehr ähnlich (vgl. Tabelle 3.4).

Die Bewertung der empfehlenswerten Anforderungen entspricht der bei Shibboleth. Während SimpleSAMLphp unterschiedliche Protokolle unterstützt, bietet Shibboleth bessere Anleitungen zur Installation. Zusätzlich ist die Konfiguration der Filter, auch ARP, bei SimpleSAMLphp schwieriger und umständlicher, wodurch [DSA-ARPs] und [DSA-Datenschutz] schlechter bewertet wurden.

3.2.4. PySAML2

Als dritte OpenSource Implementierung wird PySAML2 [Hed11] betrachtet, die federführend von Roland Hedberg in Python geschrieben wurde. PySAML2 basiert auf Web Service Gateway Interface (WSGI), einem standardisierten Interface zwischen Webserver und Webanwendung. Für den Lebenszyklus einer Anfrage wird das WSGI Framework *repoze.who* verwendet. *repoze.who* teilt Anfrage auf zwei Bereiche auf:

- *Ingress*: request classification, identification, authentication, metadata provision
- *Egress*: challenge decision, challenge, remember

Die Implementierung unterstützt nativ virtuelle Organisationen und Attribut Aggregation. Hingegen existieren Lokalisierungsdienst und Consent nur rudimentär. Im Folgenden werden

Anforderung	Priorität	Bewertung	Anforderung	Priorität	Bewertung
Funktionale Anforderungen					
[FA-Aktualisierung]	2	+	[FA-Konnektor]	2	+
[FA-Attributswahl]	2	o	[FA-Langlebigkeit]	1	+
[FA-Automatisierung]	2	-	[FA-LoA]	2	+
[FA-Datenkategorisierung]	1	+	[FA-Lokalisierung]	1	o
[FA-Dynamik]	2	-	[FA-LoT]	2	-
[FA-Fehlermanagement]	2	o	[FA-Metadaten]	2	+
[FA-Föderation]	1	+	[FA-Monitoring]	2	-
[FA-Grenzüberschreitend]	1	+	[FA-Pull&Push]	1	+
[FA-Initiierung]	2	-	[FA-Realisierbarkeit]	1	+
[FA-Integration]	1	+	[FA-Reichweite]	2	+
[FA-Interaktion]	1	+	[FA-Rollen]	2	+
[FA-Konfiguration]	1	+	[FA-Schema]	2	o
Nichtfunktionale technische Anforderungen					
[NFA-Dokumentation]	1	+	[NFA-Performanz]	2	+
[NFA-Implementierungsunabhängigkeit]	2	+	[NFA-Protokollunabhängigkeit]	2	+
[NFA-Koexistenz]	1	+	[NFA-Skalierbarkeit]	1	o
[NFA-OpenSource]	1	+	[NFA-Usability]	2	o
Sicherheitsanforderungen					
[SEC-ARPs]	1	o	[SEC-Integration]	2	+
[SEC-Auditing]	2	+	[SEC-LoA]	2	+
[SEC-Authentifizierung]	1	+	[SEC-LoT]	2	-
[SEC-Automatisierung]	2	-	[SEC-Multilateral]	1	o
[SEC-Datenübertragung]	1	+	[SEC-Systemsicherheit]	1	+
[SEC-Initiierung]	2	-			
Organisatorische Anforderungen					
[ORG-Automatisierung]	2	-	[ORG-Realisierbarkeit]	1	o
[ORG-Föderation]	2	+	[ORG-Registrierung]	1	o
[ORG-Konfiguration]	2	o	[ORG-Schema]	2	o
[ORG-LoA]	2	o	[ORG-Supportprozesse]	2	-
[ORG-LoT]	2	-	[ORG-Validierung]	1	o
[ORG-Migration]	2	o			
Datenschutzanforderungen					
[DSA-ARPs]	1	o	[DSA-LoT]	2	-
[DSA-Datenschutz]	1	o	[DSA-Selbstbestimmung]	1	+
[DSA-Interaktion]	2	o	[DSA-Zustimmung]	2	+

Tabelle 3.4.: Bewertung von SimpleSAMLphp

daher diese Aspekte nicht betrachtet. Die Konfiguration von PySAML2 bezüglich Attribute unterscheidet sich mit ihren Möglichkeiten von Shibboleth und SimpleSAMLphp und wird ausführlicher beschrieben.

Attribute Handling

```
"virtual_organization" : {
    "urn:mace:example.com:it:tek":{
        "nameid_format" : "urn:oid:1.3.6.1.4.1.1466.115.121.1.15-NameID",
        "common_identifier": "umuselin",
    }
}
```

Listing 3.16: Definition von virtuellen Organisationen, aus [Hed11]

PySAML2 unterstützt mehrere Namensschemata inklusive `common identifier`, wie in Listing 3.16 zu sehen. Dies hilft IdPs und SPs, die in mehreren Föderationen und Communities vertreten sind, wenn sie jeweils ein gemeinsames Namensschema verwenden.

```
MAP = {
    "identifier": "urn:oasis:names:tc:SAML:2.0:attrname-format:basic",
    "fro": {
        'urn:mace:dir:attribute-def:aRecord': 'aRecord',
        'urn:mace:dir:attribute-def:aliasedEntryName': 'aliasedEntryName',
        'urn:mace:dir:attribute-def:aliasedObjectName': 'aliasedObjectName',
        'urn:mace:dir:attribute-def:associatedDomain': 'associatedDomain',
        'urn:mace:dir:attribute-def:associatedName': 'associatedName',
        ...
        },
    "to": {
        'aRecord': 'urn:mace:dir:attribute-def:aRecord',
        'aliasedEntryName': 'urn:mace:dir:attribute-def:aliasedEntryName',
        'aliasedObjectName': 'urn:mace:dir:attribute-def:aliasedObjectName',
        'associatedDomain': 'urn:mace:dir:attribute-def:associatedDomain',
        'associatedName': 'urn:mace:dir:attribute-def:associatedName',
        ...
    }
}
```

Listing 3.17: Mapping von Attributen, aus [Hed11]

PySAML2 verwendet ein Python Dictionary, um Attribute zu Mappen (vgl. Listing 3.17). `identifier` beschreibt das Namensformat, welches unterstützt werden soll. `to` und `fro` enthalten anschließend das Mapping zwischen den Namen. Dies ist ein Beispiel für einfaches Umbenennen. Die Umbenennung erfolgt durch `attribute_converter.py`. Für kompliziertere Konvertierungen müssen regex oder eigener Python-Code eingesetzt werden. Nativ werden alle Attribute an den Service Provider gesendet, wenn dies nicht anders konfiguriert ist.

Über die Konfiguration kann entschieden werden, ob IdPs bzw. AAs für bestimmte Service Provider anders reagieren bezüglich Attribute. `default` und die SP EntityID werden hier zur Identifizierung verwendet. Default gilt immer dann, wenn kein Eintrag für diesen speziellen Service Provider existiert.

```
"service": {
    "idp": {
        "policy": {
            "default": {
                "lifetime": {"minutes":15},
                "attribute_restrictions": None, # means all I have
                "name_form": "urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
            },
            "urn:mace:example.com:saml:daniela:sp": {
                "lifetime": {"minutes": 5},
                "attribute_restrictions":{
                    "givenName": None,
                    "surName": None,
                }
            }
        }
    }
}
```

Listing 3.18: Konfiguration von Attributen, aus [Hed11]

Listing 3.18 zeigt eine solche beispielhafte Konfiguration. Die hierfür verwendeten Identifier sind:

- `lifetime`: Beschreibt die maximale Lebensdauer.
- `attribute_restrictions`: Einschränkung der versendeten Attribute. Im Beispiel wird an `urn:mace:example.com:saml:daniela:sp` nur der Name aus `givenName` und `surName` gesendet.
- `name_form`: Verwendetes Namensschema.

Einschränkungen können über regex erweitert werden, beispielsweise können nur E-Mail-Adressen herausgegeben werden, die auf `lrz.de` enden, wenn die Einschränkung folgendermaßen heißt: `"mail": [".*.lrz.de$"],` Nachdem mehrere Namensschemata unterstützt werden, hilft dies Entitäten, die mit unterschiedlichen Schemata arbeiten müssen.

Bewertung

Auf Grund der unterschiedlichen Funktionen von PySAML2 ist die Bewertung diffizil. Wie in Tabelle 3.5 zu sehen, ist die Konfiguration ebenso wie die Schema-Unterstützung sehr gut, während Lokalisierung und Interaktion schlecht abschneiden.

Anforderung	Priorität	Bewertung	Anforderung	Priorität	Bewertung
Funktionale Anforderungen					
[FA-Aktualisierung]	2	+	[FA-Konnektor]	2	+
[FA-Attributswahl]	2	o	[FA-Langlebigkeit]	1	o
[FA-Automatisierung]	2	o	[FA-LoA]	2	+
[FA-Datenkategorisierung]	1	+	[FA-Lokalisierung]	1	o
[FA-Dynamik]	2	-	[FA-LoT]	2	-
[FA-Fehlermanagement]	2	o	[FA-Metadaten]	2	+
[FA-Föderation]	1	+	[FA-Monitoring]	2	-
[FA-Grenzüberschreitend]	1	+	[FA-Pull&Push]	1	+
[FA-Initiierung]	2	-	[FA-Realisierbarkeit]	1	+
[FA-Integration]	1	+	[FA-Reichweite]	2	+
[FA-Interaktion]	1	o	[FA-Rollen]	2	+
[FA-Konfiguration]	1	+	[FA-Schema]	2	+
Nichtfunktionale technische Anforderungen					
[NFA-Dokumentation]	1	+	[NFA-Performanz]	2	+
[NFA-Implementierungsunabhängigkeit]	2	+	[NFA-Protokollunabhängigkeit]	2	+
[NFA-Koexistenz]	1	+	[NFA-Skalierbarkeit]	1	o
[NFA-OpenSource]	1	+	[NFA-Usability]	2	o
Sicherheitsanforderungen					
[SEC-ARPs]	1	+	[SEC-Integration]	2	+
[SEC-Auditing]	2	+	[SEC-LoA]	2	+
[SEC-Authentifizierung]	1	+	[SEC-LoT]	2	-
[SEC-Automatisierung]	2	-	[SEC-Multilateral]	1	o
[SEC-Datenübertragung]	1	+	[SEC-Systemsicherheit]	1	+
[SEC-Initiierung]	2	-			
Organisatorische Anforderungen					
[ORG-Automatisierung]	2	-	[ORG-Realisierbarkeit]	1	o
[ORG-Föderation]	2	+	[ORG-Registrierung]	1	o
[ORG-Konfiguration]	2	+	[ORG-Schema]	2	+
[ORG-LoA]	2	o	[ORG-Supportprozesse]	2	-
[ORG-LoT]	2	-	[ORG-Validierung]	1	+
[ORG-Migration]	2	o			
Datenschutzanforderungen					
[DSA-ARPs]	1	o	[DSA-LoT]	2	-
[DSA-Datenschutz]	1	o	[DSA-Selbstbestimmung]	1	o
[DSA-Interaktion]	2	o	[DSA-Zustimmung]	2	o

Tabelle 3.5.: Bewertung von PySAML2

3.2.5. Active Directory Federation Services

ADFS ist eine kommerzielle Implementierung von Microsoft, die auf dem Active Directory (AD) aufbaut. ADFS existiert seit einigen Jahren und unterstützt inzwischen sowohl SAML als auch OAuth. Die Sprache von ADFS ist an OpenID Connect und OAuth angelehnt. Eine Assertion wird hier Security Token genannt, ein Identity Provider ist ein Claims Provider, der Service Provider wird Relying Party genannt und die Attribute sind Claims. Nachdem verstärkt Identity Provider auf ADFS wechseln, da AD häufig in Unternehmen und Universitäten eingesetzt wird, wird ADFS hier ebenfalls betrachtet.

Lokalisierung

ADFS [Mic15] ist Microsofts Implementierung des Web Services (WS)-Federation Passive Requestor Profile Protokolls, welches zudem SAML und OAuth Claims zulässt. ADFS wird über ein Administrationstool (*adfs.msc*) in der Microsoft Management Console verwaltet, indem es u. a. möglich ist

- Benutzerkonten und
- Partner-Entitäten hinzuzufügen sowie
- Attribute zu konvertieren.

Neben der eigentlichen ADFS-Instanz benötigt ADFS ein AD, eine Structured Query Language (SQL) Datenbank sowie möglichst den Web Application Proxy, der in der demilitarisierten Zone steht und ADFS mit Partner-Entitäten verbindet.

Partner-Entitäten können entweder manuell angelegt oder über Metadaten hinzugefügt werden. Um diese Schritte zu automatisieren, wurde von Cristian Mezzetti das Tool FEMMA[1] entwickelt. FEMMA besteht aus einem Python Skript, welches die Metadaten parst und die Entitäten über ein Windows PowerShell Skript einzeln in ADFS importiert. Zusätzlich fügt FEMMA Templates hinzu, um automatisch Claim Rules für die neuen Partner zu importieren. Damit ADFS als reiner IdP oder SP mit anderen Implementierungen interoperabel ist, müssen die Metadaten angepasst werden. ADFS-IdPs haben automatisch auch den SPDescriptor in den Metadaten, was bei anderen Implementierungen zu Fehlern führt.

Damit ein Nutzer eine Relying Party verwenden kann, gibt es bei ADFS verschiedene Lösungen zur Lokalisierung. ADFS bietet so genannte Sign In Pages, um Föderationsanfragen, was bei ADFS immer bilateral ist, zu behandeln:

- Claim Provider-initiierte Anfragen werden über die `IdPInitiatedSignOn.aspx` Seite

[1] Federation Metadata Manager for ADFS: `http://sourceforge.net/projects/femma/` [Online, abgerufen am 06.01.2016].

abgehandelt. Der Nutzer kann hier aus verschiedenen Anwendungen die gewünschte auswählen.

- `HomeRealmDiscovery.aspx` präsentiert dem Nutzer eine Auswahl an Organisationen, zu denen er gehören könnte. Dies ist vergleichbar mit den Lokalisierungsdiensten bei Shibboleth und SimpleSAMLphp.

- `AutoLogon.aspx` versucht den Nutzer automatisch über den Identity Selector zu authentifizieren. Dies funktioniert nur, wenn der Nutzer dies ausdrücklich wünscht.

Die Lokalisierung basiert jeweils auf dem Common Domain Cookie. Nachdem sich der Nutzer das erste Mal authentifiziert hat, wird die uniqueID des Claim Providers in den Cookie des Benutzers geschrieben. Diese Information kann die Relying Party verwenden.

Claim Rules und Claim Mapping

Claim Rules beschreiben, welche Claims an die Relying Party über ein Security Token gesendet werden. Zunächst müssen die Claims aus dem AD geholt werden. Nachdem Claims durch ADFS nicht das in R&E-Föderationen üblichen Format gesendet werden, müssen sie erst auf das Zielformat über die ADFS 2.0 Custom Rule Language transformiert werden. Dies geschieht erneut über das Administrationstool, in welches, nach der Auswahl der zu erstellenden Regeln, die Transformation geschrieben wird, wie in Listing 3.19 dargestellt.

```
1 c:[Type == "http://schemas.xmlsoap.org/ws/2005/05/identity/claims/upn"]
2 =>        issue(Type = "urn:oid:1.3.6.1.4.1.5923.1.1.1.6",
3           Value = c.Value,
4           Properties["http://schemas.xmlsoap.org/ws/2005/05/identity/
      claimproperties/attributename"] = "urn:oasis:names:tc:SAML:2.0:attrname-
      format:uri");
```

Listing 3.19: Umbenennung von Attributen in ADFS

ADFS fügt keine `scoped`-Elemente in die automatisch generierten Metadaten hinzu und es gibt keine Möglichkeit, Daten in die Metadaten hinzuzufügen. Nachdem R&E für viele Attribute `scope` verwendet, ist es wichtig, die Metadaten von Hand zu editieren und zu signieren oder einen Proxy hierfür zu verwenden.

Consent Modul

Die Seite `IdPInitiatedSignOn.aspx` Seite enthält die Eigenschaften `IsPassive/ForceAuthn`, `Consent` und `RequestedAuthenticationContext`, die jedoch erst angepasst werden müssen. Die `Consent` Eigenschaft enthält den Consent des Benutzers. Die ID des Consents ist eine URI, die spezifiziert, wie der Consent erbracht wurde. Wenn die `Consent` Eigenschaft

aktiviert ist, wird sie als Dropdown-Liste angezeigt.

Bewertung

Im Gegensatz zu Shibboleth und SimpleSAMLphp, wurde ADFS nicht für SAML implementiert und unterscheidet sich daher in einigen Eigenschaften, wie in Tabelle 3.6 zu sehen.

Anforderung	Priorität	Bewertung	Anforderung	Priorität	Bewertung
Funktionale Anforderungen					
[FA-Aktualisierung]	2	+	[FA-Konnektor]	2	+
[FA-Attributswahl]	2	-	[FA-Langlebigkeit]	1	o
[FA-Automatisierung]	2	-	[FA-LoA]	2	-
[FA-Datenkategorisierung]	1	-	[FA-Lokalisierung]	1	o
[FA-Dynamik]	2	-	[FA-LoT]	2	-
[FA-Fehlermanagement]	2	o	[FA-Metadaten]	2	o
[FA-Föderation]	1	o	[FA-Monitoring]	2	o
[FA-Grenzüberschreitend]	1	+	[FA-Pull&Push]	1	+
[FA-Initiierung]	2	-	[FA-Realisierbarkeit]	1	o
[FA-Integration]	1	+	[FA-Reichweite]	2	+
[FA-Interaktion]	1	o	[FA-Rollen]	2	+
[FA-Konfiguration]	1	o	[FA-Schema]	2	-
Nichtfunktionale technische Anforderungen					
[NFA-Dokumentation]	1	o	[NFA-Performanz]	2	+
[NFA-Implementierungsunabhängigkeit]	2	o	[NFA-Protokollunabhängigkeit]	2	+
[NFA-Koexistenz]	1	+	[NFA-Skalierbarkeit]	1	o
[NFA-OpenSource]	1	-	[NFA-Usability]	2	o
Sicherheitsanforderungen					
[SEC-ARPs]	1	o	[SEC-Integration]	2	+
[SEC-Auditing]	2	+	[SEC-LoA]	2	-
[SEC-Authentifizierung]	1	+	[SEC-LoT]	2	-
[SEC-Automatisierung]	2	-	[SEC-Multilateral]	1	o
[SEC-Datenübertragung]	1	+	[SEC-Systemsicherheit]	1	+
[SEC-Initiierung]	2	-			
Organisatorische Anforderungen					
[ORG-Automatisierung]	2	o	[ORG-Realisierbarkeit]	1	o
[ORG-Föderation]	2	o	[ORG-Registrierung]	1	o
[ORG-Konfiguration]	2	-	[ORG-Schema]	2	-
[ORG-LoA]	2	-	[ORG-Supportprozesse]	2	o
[ORG-LoT]	2	-	[ORG-Validierung]	1	-
[ORG-Migration]	2	o			
Datenschutzanforderungen					
[DSA-ARPs]	1	o	[DSA-LoT]	2	-
[DSA-Datenschutz]	1	o	[DSA-Selbstbestimmung]	1	o
[DSA-Interaktion]	2	-	[DSA-Zustimmung]	2	o

Tabelle 3.6.: Bewertung von ADFS

Die Integration mit AD kann als Vorteil gewertet werden, jedoch kann ADFS viele Anforderungen aus Kapitel 2 nicht oder nur teilweise erfüllen.

Weitere kommerzielle Produkte sind u. a. PingFederate, GlobalSign SSO, iSAML, Oracle Identity Federation, SecureAuth, Simplified und TrustBuilder. Es zeigt sich dabei, dass viele Hersteller parallel SAML und OpenID Connect unterstützen.

3.3. Technisches Vertrauen durch Metadaten

Im nachfolgenden Abschnitt wird auf die Metadaten-Verwaltung, vgl. wichtige Anforderung [FA-Metadaten], und die Distribution der Daten eingegangen. FIM-Software setzt den Austausch der Metadaten zwischen zwei Entitäten, die miteinander kommunizieren, technisch voraus, während Föderationen die organisatorische Umsetzung realisieren. In diesem Abschnitt werden sowohl aktuell vorhandenen Lösungen für Identity network und Hub-and-Spoke Föderationen als auch ein Forschungsansatz vorgestellt. Die Darstellung soll primär die bisherige Akzeptanz und die Umsetzung reflektieren, sowie Defizite in den aktuellen Lösungen aufzeigen.

3.3.1. Resource Registry der SWITCHaai

Die schweizerische Föderation SWITCHaai spielte bei der Metadatenverwaltung eine der Vorreiterrollen in Europa. Zur Administration der Metadaten der einzelnen Entitäten in der nationalen Föderation wurde eine Webanwendung erstellt, die sowohl von IdPs als auch SPs verwendet wird [Häm06]. Ähnliche Webanwendungen wurden anschließend in den meisten nationalen Föderationen eingeführt. Die sogenannte Resource Registry (RR) wurde zuletzt 2011 von SWITCHaai für die Teilnahme an der Inter-Föderation erweitert und bietet u. a. folgende Funktionen an:

- Die benötigten und gewünschten Attribute können vom Service Provider aufgelistet werden, während Identity Provider die Attribute, die sie anbieten, angeben können.
- Parallel dazu können SPs ihre Zielgruppen genauer spezifizieren. Dies hat den Hintergrund, dass IdPs nicht Metadaten erhalten sollen, die sie gar nicht nutzen können.
- Genehmigung der Entität und der Ressource nach der Registrierung durch die Föderationsverwaltung.
- Generierung der Metadaten der Föderation durch einen Metadata Aggregator. Hier agiert SWITCHaai als Broker des technischen Vertrauens.
- Halb-automatische Generierung der Attribute Release Filter anhand der gemachten Angaben.
- Generierung einiger Konfigurationsdateien durch die Informationen, die in der Datenbank hinterlegt sind.
- Allgemeine Informationsseite mit Statistiken und Listen über die Föderation.

Bei der Registrierung muss der zuständige Administrator Informationen zu seiner Entität sowie die Metadaten hochladen. Zu den Basisinformationen gehören u. a. der Name der Or-

ganisation, der Name des Dienstes, die Beschreibung des Dienstes, die EntityID, die URL des Dienstes, die URL des Servicedesks, die Gültigkeit des Zertifikats und die Sichtbarkeit. Aus den hochgeladenen Metadaten wird der aggregierte, nationale Metadatensatz gebildet. Ein interessanter Aspekt der Resource Registry ist die Angabe der Attribute. Wie bereits erwähnt, können Service Provider die benötigten und gewünschten Attribute angeben. Diese müssen nicht zwingend aus den durch die SWITCHaai unterstützten Schemata entstammen, sondern können auch andere, standardisierte Attribute sein, die beispielsweise intern oder in Kollaborationen verwendet werden. Dasselbe gilt für Identity Provider, die diejenigen Attribute benennen, die sie an die Service Provider schicken können. Die halb-automatische Generierung des Attribute Release Filters basiert auf einer grobgranularen Einstellung bezüglich des Gültigkeitsbereiches:

- Attribute, die an keinen veröffentlicht werden sollen, werden mit `Nobody` markiert.
- Attribute, die nur an die eigene Organisation gehen, bekommen die Bezeichnung `Resources of my organization`.
- Zudem können Attribute für die Föderation sowie für die Inter-Föderation freigegeben werden.

Für die Feineinstellung wird der Scope pro Attribut, unterteilt in benötigt und gewünscht, festgelegt. Während diese Funktionen Richtung Automatisierung führen, sind trotzdem viele manuelle Schritte über die Webanwendung nötig. Diese Defizite bestehen auch in den anderen nationalen Lösungen, wie beispielsweise der Metadaten-Verwaltung der DFN-AAI und in der SimpleSAMLphp-Erweiterung JANUS [vL15]. Zudem sind die Einstellungsmöglichkeiten für den Attribute Release Filter sehr generisch. Es können nur Attribute aus den Schemata, die die Föderation SWITCHaai unterstützt, ausgewählt werden. Ferner ist es nicht möglich Attribute auf eine Kollaboration zu beschränken oder zusätzliche Scopes einzurichten. Verschiedene Aspekte, wie Entity Categories, die Einrichtung in bestimmte Gruppen einteilen, und die Akzeptanz des Code of Conduct, werden nicht beachtet. Folglich ist die Lösung zwar ein Fortschritt, aber zu manuell und zu unflexibel für den Einsatz in dynamischen Föderationen.

3.3.2. IdP-Proxy

Während vollvermaschte Föderationen eine zentrale Webanwendung einsetzen, benutzen Hub-and-Spoke Föderationen, wie die niederländische Föderation SURFconext Federation, einen IdP-Proxy, um Identity Provider und Service Provider miteinander zu verbinden. Ein SAML IdP Proxy agiert als Bridge oder Gateway zwischen den IdPs und SPs [CS12] [Lin09]. Aus Sicht des IdPs operiert der IdP-Proxy als Service Provider, während der SP im IdP-Proxy einen Identity Provider sieht. Ein IdP-Proxy hat die folgenden Eigenschaften:

- Zwischenspeicherung von Attributen, was wiederum die Effizienz steigern kann.

- Zentrale Attributskonvertierung ist relativ einfach realisierbar.
- Kontrollierter Zugriff zu den IdPs einer Föderation.
- Möglichkeit der Filterung von SAML Nachrichten, d. h. Requests und Responses.

Auf Grund der zentralen Lage reduziert sich die Anzahl der geschlossenen Verträge von nxm auf $n+m$, wobei n die Anzahl der IdPs und m die Anzahl der SPs darstellt. Ein Vorteil des IdP-Proxys ist die zentrale Möglichkeit der Attributskonvertierung. Während der Proxy mit einer Bridge für die Übersetzung in andere Protokolle verbunden sein kann, ist es innerhalb einer Föderation Stand 2016 nicht möglich unterschiedliche Protokolle zu vermischen. Zudem kann es bei einer größeren Anzahl an Entitäten zu Performanceeinbußen kommen, während zugleich der IdP-Proxy ein potentieller Single Point of Failure darstellt. Laut den Erfahrungen von SURFconext ist der Betrieb eines IdP-Proxys inklusive der Verwaltung arbeitsintensiver im Vergleich zu vollvermaschten Föderationen. Dies widerspricht dem flexiblen Einsatz für dynamische Föderationen.

3.3.3. Metadata Distribution Service in eduGAIN

Die Inter-Föderation eduGAIN stellt sowohl einen rechtlichen Rahmen als auch die Werkzeuge für den Austausch der Metadaten. Zum Policy Framework gehören folgende Dokumente, die u. a. die Kommunikation zwischen Entitäten regeln:

- *eduGAIN Declaration*: Formales Dokument, welches die Pflichten der teilnehmenden Föderationen auflistet.
- *eduGAIN Constitution* beschreibt die Anforderungen an Föderationen, beispielsweise bezüglich Metadaten. So muss jede teilnehmende Föderation ihre Metadaten veröffentlichen und das *eduGAIN SAML 2.0 Metadata Profile* verwenden.
- *Data Protection Code of Conduct* legt die rechtliche Grundlage für Service Provider bei der Verarbeitung von Daten fest.
- *eduGAIN SAML 2.0 WebSSO Profile*: Beschränkung der erlaubten Profile auf [`SAML2int`].
- *eduGAIN Metadata Profile* ergänzt das *SAML V2.0 Metadata Interoperability Profile* um zusätzliche Anforderungen. So müssen die Kontaktdaten in den Metadaten der einzelnen Entitäten aufgeführt werden.
- *eduGAIN Attribute Profile* enthält die empfohlenen Attribute, die von IdPs unterstützt werden sollen.

Basierend auf dem rechtlichen Rahmen erstellt eduGAIN den Metadatensatz der Inter-Föderation. Dazu müssen die teilnehmenden Föderationen die Metadaten aller Entitäten,

die in eduGAIN teilnehmen über eine Art Resource Registry sammeln und anschließend mit einem Metadata Aggregator aggregieren. Jede Föderation muss für die Inter-Föderation eine eigene Metadaten-Datei erstellen, da nicht alle Entitäten der Föderation in eduGAIN teilnehmen, wie in der folgenden Tabelle 3.7 zu sehen:

Föderation	Größe der Datei	Entitäten	Entitäten in eduGAIN
DFN-AAI	3,5 bzw. 3,2 MB	473	86
Fédération Éducation-Recherche	12,8 MB	936	273
UK Federation	20,7 MB	2920	815
eduGAIN	17,9 MB	2556	-

Tabelle 3.7.: Gegenüberstellung der Entitäten in Föderationen und in eduGAIN

So nehmen in Deutschland 86 von 473 Entitäten in der Inter-Föderation teil, während in Frankreich bei Fédération Éducation-Recherche 273 von insgesamt 936 Entitäten bei eduGAIN sind. In der UK Federation machen 815 von 2920 Entitäten mit. Die veröffentlichten Metadaten für die Inter-Föderation werden vom MDS geladen, validiert, aggregiert und veröffentlicht. Jede teilnehmende Föderation lädt den so aggregierten Metadatensatz, filtert eigenen Entitäten heraus und veröffentlicht die Datei erneut. Auch wenn dieses Konzept bislang funktioniert, hat es Defizite. So wächst die Größe der aggregierten Metadaten zunehmend, da immer mehr Entitäten und Föderationen in eduGAIN sind. Die Größe schlägt sich auf die Performanz nieder. Je mehr Entitäten und Föderationen an der Inter-Föderation teilnehmen, desto größer wird der Metadatensatz. Gleichzeitig benötigt keine Entität die Metadaten aller Teilnehmer, da sie nur zu bestimmten anderen Entitäten Beziehungen pflegen, wie bereits im Abschnitt 2.3.6 aufgezeigt wurde.

3.3.4. Metadata Query Protocol und PEER

Das Metadata Query Protocol [You15], welches von Ian Young im Rahmen von Research and Education Federations (REFEDS) spezifiziert wird, fokussiert den Austausch von Metadaten. Durch die Verwendung von spezifischen HTTP GET Anfragen sollen die gewünschten Metadaten abgerufen werden können. Darauf basierend wird ein Profile spezifiziert, welches die Verwendung des Metadata Query Protocols für SAML Metadaten beschreibt. Als Identifikator soll die EntityID verwendet werden, die einmalig und in den Metadaten enthalten ist. In dem Beispiel 3.20 werden die Metadaten des LRZs abgefragt:

```
1 GET /service/entities/http%3A%2F%2Flrz.de%2Fidp HTTP/1.1
2     Host: metadata.lrz.de
3     Accept: application/samlmetadata+xml
```

Listing 3.20: HTTP GET Anfrage zum Abruf von Metadaten

Als Antwort werden, wie in Beispiel 3.21 zu sehen, mit Typ `samlmetadata+xml` die entsprechenden Metadaten geliefert. Äquivalent zu HTTP werden die Status-Codes verwendet,

um Fehler oder einen Erfolg anzuzeigen.

```
1  HTTP/1.x 200 OK
2  Content-Type: application/samlmetadata+xml
3  ETag: abcdefg
4  Last-Modified: Thu, 15 Apr 2010 12:45:26 GMT
5  Content-Length: 1234
6
7  <?xml version="1.0" encoding="UTF-8"?>
8  <EntityDescriptor entityID="http://lrz.de/idp"
9      xmlns="urn:oasis:names:tc:SAML:2.0:metadata">
10 ....
```

Listing 3.21: HTTP Antwort mit Metadaten

Die Implementierung des Metadata Query Protocols ist eng verknüpft mit dem PEER Projekt, welches eine zentrale Open Source Registrierungskomponente entwickelt und auf dem Konzept einer Metadaten-Schicht von Ian A. Young und Chad La Joie [YLJ09] beruht. Der bei REFEDS implementierte Dienst von PEER wird als REEP bezeichnet und akzeptiert weitere Metadaten-Elemente, wie Attributsanforderungen und Entity Categories. Die Metadaten werden durch Aggregation gefiltert, transformiert, validiert und zusammengefügt, was durch Shibboleth Metadata Aggregation und python Federation Feeder (pyFF) implementiert ist. pyFF ist ein in Python geschriebenes Programm, welches Metadaten verarbeitet, indem es sie aggregiert, validiert, kombiniert, transformiert, signiert und veröffentlicht. Der Abruf der Metadaten soll anschließend über das Metadata Query Protocol geschehen. Während eine zentrale Verwaltung der Metadaten das Metadaten-Management minimiert, soll PEER bzw. REEP parallel zur aktuellen Föderationsstruktur existieren. Wenn Föderationen REEP nicht akzeptieren, müssen die Entitäten erneut an mehreren Stellen ihre Metadaten pflegen. Auch wenn das Metadata Query Protocol das Potential hat einzelne Metadaten nach Bedarf abzurufen, werden weiterhin aggregierte Metadaten, beispielsweise pro Föderation und Inter-Föderation, versandt.

Somit bleiben viele Defizite bei den betrachteten Ansätzen weiterhin bestehen, wie auch aus der Tabelle 3.8 hervorgeht. Die Aktualisierung muss manuell angestoßen werden. Eine Automatisierung ist bei PEER zwar möglich, wird allerdings nicht eingesetzt. Folglich gibt es auch keine Dynamik bei der Etablierung von Vertrauen und keine Initiierung durch den Nutzer. Es werden mehrere Föderationen unterstützt, wobei die Resource Registry wie auch ein üblicher IdP-Proxy nur die eigene verwaltet. Prinzipiell sind grenzüberschreitende Kooperationen möglich, jedoch nur, soweit die Metadaten vorab ausgetauscht wurden. Daher wird diese Anforderung nur teilweise erfüllt. Die feingranularste Konfiguration bietet die RR, solange sich der Kooperationspartner innerhalb der gleichen Föderation befindet. Dasselbe gilt für die Verwendung unterschiedlicher Schemata. Zugleich ist es möglich, sich seine Metadaten und einfache Konfigurationsdateien generieren zu lassen. Bei den nichtfunktionalen technischen Anforderungen gibt es ebenfalls Defizite, beispielsweise im Bereich des OpenSource oder der Dokumentation. Die multilaterale Sicherheit wird von keinem Ansatz betrachtet. Die organisatorische Einbindung ist bei der Resource Registry am besten, während bei PEER die Unterstützung fehlt. Der Code of Conduct kann in jedem Metadatensatz

eingebunden werden, wobei dies nur in der Resource Registry und Metadata Distribution Service aktiv propagiert wird.

Anforderung	Priorität	RR	IdP-Proxy	MDS	PEER
Funktionale Anforderungen					
[FA-Aktualisierung]	2	o	+	o	o
[FA-Automatisierung]	2	-	-	-	o
[FA-Dynamik]	2	-	-	-	-
[FA-Föderation]	1	+	+	+	+
[FA-Grenzüberschreitend]	1	o	-	o	o
[FA-Initiierung]	2	-	-	-	-
[FA-Integration]	1	+	+	+	+
[FA-Konfiguration]	1	o	o	-	-
[FA-Langlebigkeit]	1	+	+	+	o
[FA-Metadaten]	2	+	o	o	o
[FA-Monitoring]	2	+	+	-	o
[FA-Reichweite]	2	o	-	o	o
[FA-Schema]	2	o	+	o	-
Nichtfunktionale technische Anforderungen					
[NFA-Dokumentation]	1	+	+	o	o
[NFA-Koexistenz]	1	o	-	+	+
[NFA-OpenSource]	1	o	o	o	+
[NFA-Performanz]	2	o	-	o	o
[NFA-Skalierbarkeit]	1	-	o	-	o
Sicherheitsanforderungen					
[SEC-Metadaten]	3	+	+	+	o
[SEC-Multilateral]	1	-	o	-	-
[SEC-Systemsicherheit]	1	o	+	o	o
Organisatorische Anforderungen					
[ORG-Föderation]	2	+	+	+	o
[ORG-Konfiguration]	2	+	+	o	-
[ORG-Metadaten]	3	+	o	o	o
[ORG-Registrierung]	1	+	o	-	o
[ORG-Schema]	2	+	o	-	-
[ORG-Validierung]	1	+	o	-	+
Datenschutzanforderungen					
[DSA-CoCo]	3	+	+	+	o
[DSA-Datenschutz]	1	-	o	-	-

Tabelle 3.8.: Bewertung der Ansätze im Bereich der Metadatenverwaltung

3.4. Forschungsansätze zu Vertrauen in Föderationen

Allgemein kann in auf SAML basierenden Kollaborationen zwischen technischem Vertrauen (*technical trust*) über den Austausch der Metadaten und behavioural trust unterschieden werden. Wenn in einer Gruppe, im Federated Identity Management beispielsweise in einer Föderation, jeder jedem vertraut, ist auch von einem *CoT* die Rede. Dieser Begriff stammt von der Liberty Alliance. Madsen et al. [MKT05] definieren eine Föderation als eine Gruppierung von Verträgen, kryptographisches Vertrauen und Nutzer-IDs oder Attribute über Security und Policy Domänen, um nahtlose Geschäfts-Interaktionen über verschiedene Domänen zu erreichen. Im Gegensatz dazu definieren Jøsang et al. [JFH+05] eine Föderation als ein Satz von Vereinbarungen, Standards und Technologien, die es einer Gruppe von SPs erlauben IDs und Entitlements von Nutzern innerhalb der Gruppe zu erkennen und zu verwenden. Das zeigt, dass eine Föderation bzw. ein CoT aus einer Menge an SPs, Nutzern, Attributen und eine Art Verträgen besteht. Das Vertrauen basiert beim CoT auf Regeln, Verantwortungen und Verpflichtungen. Eine darauf aufbauende Inter-Föderation besteht aus einer Topologie von Vertrauen (*trust topology*). Wenn eine Organisation außerhalb des Circle of Trust mit einem Mitglied des CoT eine Kooperation möchte, kann laut Latifa Boursas [Bou09] ein anderes Mitglied des CoT gefragt werden, welches das Nichtmitglied kennt. Somit gibt es direktes und indirektes Vertrauen. Direktes, bilaterales Vertrauen existiert beispielsweise zwischen zwei Organisationen in einem CoT (*trust by membership*) oder wenn beide Geschäftsbeziehungen miteinander pflegen. Indirektes, transitives Vertrauen wiederum ist dann vorhanden, wenn eine dritte Organisation als TTP agiert. Vertrauen hat unterschiedliche Dimensionen. Latifa Boursas [Bou09] schlüsselt für das transitive Vertrauen folgende Aspekte auf:

Trust by delegation and recommendations: Eine Form des transitiven Vertrauens, bei der eine bekannte Organisation die Berechtigung an eine unbekannte Entität erteilt, um bestimmte Funktionen auszuführen.

Trust from past experience: Hier wird das Verhalten konstant überwacht, um zukünftige Interaktionen auf Basis des Ergebnisses zu entscheiden.

Trust by reputation: Die Reputation einer Entität wird in diesem Fall mit anderen Entitäten geteilt, beispielsweise im Bereich des eCommerce.

Trust by belief: Falls die drei oben genannten Aspekte nicht vorhanden sind, spielt der Glaube an eine Entität eine Rolle, beispielsweise basierend auf der Motivation der unbekannten Organisation.

Die Szenarien in Kapitel 2 zeigten, dass selbst in einem CoT nicht unbedingt Vertrauen zwischen den Teilnehmern vorhanden sein muss. Vertrauen hat, wie gerade gezeigt, mehrere Aspekte, wozu auch die Konsequenzen im Fall, dass sich die vertrauenswürdige Entität nicht so verhält wie erwartet, zählen. Somit korreliert das Vertrauen mit dem Risiko bei Fehlern. Das Risiko wiederum besteht aus der Kombination aus Möglichkeit, dass ein Ereignis eintritt, und den Auswirkungen eines solchen Ereignisses.

Vertrauen zwischen zwei Entitäten ist notwendig, damit Benutzerinformationen ausgetauscht werden und ein Benutzer einen Dienst nutzen kann. Vertrauen besteht aus der Bereitschaft durch eine andere Entität geschädigt zu werden, im FIM durch die Weitergabe von persönlichen Daten der Nutzer, die für die erste Entität eine wichtige Aktion durchführt, und kann daher wie folgt beschrieben werden:

> *"...the willingness of a party to be vulnerable to the actions of another party based on the expectations that the other will perform a particular action important to the trustor, irrespective of the ability to monitor or control that other party"* [MDS95]

Wie bereits im vorherigen Abschnitt beschrieben, besteht der Vertrauensaufbau in aktuellen Föderationen und Inter-Föderationen auf dem vorherigen Austausch von signierten Metadaten. Dabei wird eine größere Anzahl an Daten ausgetauscht als unbedingt notwendig. Verschiedene Forschungsansätze versuchen dieses Defizit zu minimieren. Im Folgenden sind beispielhaft die Ansätze von Dynamic Identity Management and Discovery System (DIMDS), Federated Attribute Management and Trust Negotiation (FAMTN), IdMRep, Dynamic Identity Federation und Trust Service Provider (TSP) beschrieben.

3.4.1. Forschungsansatz DIMDS

Das generische Konzept DIMDS von Konstantinos Lampropoulos und Spyros Denazis [LD09] hat das Ziel, die Benutzeraktivititäten durch ein zentrales Identity Management System mit einem einheitlichen DIMDS Benutzerkonto zu reduzieren. Dazu wird ein Netz aus verteilten Servern aufgesetzt, die für die Anfragen je einer Zone zuständig sind und als Content Addressable Network (CAN) durch Distributed Hash Table (DHT) verbunden sind. Benutzer legen einen DIMDS Benutzerkonto mit einer *User account ID* an, um ihre verschiedenen Identitäten zentral zu speichern und sie zu verknüpfen. Für jedes Benutzerkonto wird eine Zufallszahl, *Random Identity Number*, generiert, die die Entität ebenfalls erhält und die zur Identifikation von Nutzern bei zukünftigen Kommunikationen dient. Durch diese Zufallszahl soll es unmöglich sein einen Nutzer mit einer realen Person zu verknüpfen. Jedoch ist die Zufallszahl zusammen mit der Benutzer-ID und weiteren Attributen in einer Nutzerdatenbank gespeichert, welche unter Umständen durch diese zusätzlichen Attribute miteinander verknüpft werden können, was wiederum die Privatsphäre beeinträchtigt. Nachdem Service Provider die Zufallszahl nicht kennen, leitet DIMDS den Service Provider mit einer temporären Zufallszahl weiter zum Identity Provider, von dem der SP die Benutzerinformationen abfragen kann. Somit agiert DIMDS als eine Art Lokalisierungsdienst, der das Vertrauen zwischen Identity Provider und Service Provider über die temporäre Zufallszahl initiiert. Es werden keine Aspekte des Vertrauensaufbaus an sich (*behavioural trust*) betrachtet, wodurch dieser Ansatz ungeeignet erscheint. Zudem müssen Benutzerkonten miteinander verknüpft werden, was aus Datenschutzsicht problematisch ist.

3.4.2. Forschungsansatz FAMTN

Das Konzept FAMTN von Abhilasha Bhargav-Spantzel et al. [BSSB07] basiert auf Föderationen, die aus Nutzern und FAMTN Service Providern bestehen. Die Kommunikation geschieht entweder zwischen Benutzer und einem SP oder zwei Service Providern. Dabei wird davon ausgegangen, dass ein Nutzer einen Dienst eines SPs nutzen will und dieser daraufhin Benutzerattribute abruft. Fraglich ist dabei, von wem die Attribute gesendet werden, da es in dem Ansatz keine IdPs gibt. Diese Benutzerinformationen können durch Verwendung der SSO ID bei der Benutzung zusätzlicher Dienste weiter verwendet werden. Externe Benutzer müssen gleich bei der ersten Kommunikation all ihre Attribute bereitstellen, damit diese mit einer temporären Nutzer ID verknüpft werden können. Dieser Ansatz hat den Nachteil, dass die SSO ID für Angriffe missbraucht werden kann. Ferner ist es möglich, dass ein Service Provider mehr oder weniger Attribute benötigt, was wieder eine Verletzung der Privatsphäre darstellen kann. Gleichzeitig ist es für die Kommunikation zwischen bestehenden Föderationen kompliziert alle Benutzerattribute bei der ersten Kommunikation herauszugeben.

3.4.3. Forschungsansatz IdMRep

IdMRep von Patricia Arias Cabarcos et al. [ACAGMM13] ändert Federated Identity Management insofern, dass es von vorab etablierten Vertrauensbeziehungen hin zu einem dynamischen Vertrauensaufbau wechselt. Dieser basiert auf der Berechnung vom dynamischen Vertrauen durch Distributed Trust List (DTL) [AAMD09] und externen Reputationsdaten. Als Trust Anchor List (TAL) wird eine statische Liste mit Entitäten bezeichnet, denen vertraut wird. DTLs erweitern TALs um zusätzliche Informationen, die für das Vertrauen wichtig sind, beispielsweise

- Daten zur Entität,
- Trust Level,
- Punktezahl und
- Schlüssel.

Diese zusätzlichen Daten werden über eine Erweiterung aktualisiert, wenn andere Entitäten eine Empfehlung aussprechen oder eine Kooperation erfolgreich beendet wurde. Sie werden bei Entscheidungsfindung hinzugezogen, um u. a. die Informationen zum Vertrauensaufbau anzureichern. Während der Pre-Federation und Post-Federation Phase werden verschiedene Phasen der Beziehung zwischen zwei Entitäten durchlaufen:

R1. Speicherung, Weitergabe und Aggregation der Daten zur Reputation einer Entität.

R2. Lokale Berechnung der Werte des Vertrauens und Risikos.

R3. Dynamische Entscheidung basierend auf Vertrauen, Reputation und Risiko.

R4. Monitoring und Anpassung des Vertrauenswertes.

DTLs erhalten für dynamische Föderationen u. a. die Reputationsdaten von anderen Entitäten, jedoch funktioniert dieser Mechanismus nicht bei neuen Mitgliedern. Ferner akkumuliert sich die Datenmenge, die von der Trust Engine bei Updates verarbeitet werden muss, mit zunehmender Teilnehmerzahl, was zu einem Single-Point-of-Failure führt. Dieser Eindruck wird durch die Tatsache verstärkt, dass die Modellierung nur mit einer geringen Anzahl an Entitäten geschehen ist und größere Föderationen und Inter-Föderationen nicht betrachtet wurden.

3.4.4. Forschungsansatz Dynamic Identity Federation

Der Ansatz Dynamic Identity Federation von Md.Sadek Ferdous und Ron Poet [FP13] konzentriert sich auf die vollautomatische Bildung von Föderationen durch IdPs und SPs sowie den Aufbau von Vertrauen in dynamischen Föderationen. Dabei unterscheidet der Ansatz zwischen verschiedenen Vertrauenstypen:

- Fully trusted entities bezeichnet zwei Entitäten in einer klassischen Föderation, zwischen denen bereits Verträge abgeschlossen wurden.
- Semi-trusted entities sind Service Provider in dynamischen Föderationen, die durch Interaktion mit einem Benutzer zu einer Föderation hinzugefügt wurden und denen mindestens ein Identity Provider Attribute gesendet hat.
- Als untrusted entities werden Entitäten angesehen, die zu einer dynamischen Föderation hinzugefügt worden sind und die noch keinen Vertrag abgeschlossen haben.

Nur ein gültiger, authentifizierter Benutzer kann einen Service Provider zu einem IdP hinzufügen. Der SP muss den jeweiligen IdP in seine TAL hinzufügen, damit der Benutzer den Service Provider das nächste Mal ohne Konfiguration benutzen kann. IdPs sollen im Gegenzug dafür sorgen, dass semi-trusted entities keine sensitiven Attribute erhalten, da es keine Garantie dafür gibt, dass die SPs die Informationen passend behandeln. Um das Vertrauensverhältnis für SPs anzuzeigen, sollen untrusted IdPs maximal den NIST LoA 1 erhalten. Wie in aktuellen Föderationen gibt es Verbindungen zwischen IdPs und SPs, die durch Benutzer initiiert wurden, indem Benutzer bei einer ersten Authentifizierung einen Code generieren, den sie anschließend beim gewünschten SP mitsamt der *EntityID* des IdPs angeben müssen. Nach der Verifikation generiert der SP eine Anfrage (*Request*) mit zwei versteckten Feldern, `MetaAdd` und `ReturnTo`. Das Feld `MetaAdd` wird anschließend vom Identity Provider ausgelesen. Wenn das Feld nicht `null` ist, wird der Wert des Codes ausgelesen und mit der MySQL-

Datenbank verglichen, in der der ursprüngliche Code gespeichert wurde. Wenn eine Übereinstimmung gefunden wurde, ist die Anfrage gültig. Anschließend findet der Metadaten-Austausch statt, für den die Felder `MetaAdd` und `ReturnTo` verwendet werden. Nun kann der Nutzer seinen IdP beim Lokalisierungsdienst des gewünschten SPs auswählen, wodurch der Standard-SAML-Workflow für Webanwendungen startet. Nachdem die beteiligten Entitäten sich nicht vollkommen Vertrauen, werden durch den IdP bestimmte Attribute nicht gesendet, was beim Consent-Modul angezeigt wird. Wenn der Nutzer der Weitergabe seiner Benutzerinformationen zustimmt, wird der Service Provider zur Liste der semi-trusted entities hinzugefügt. Falls der Identity Provider des Nutzers kein Vertrauensverhältnis zum Service Provider hat, kann ein IdP alternativ als Proxy verwendet werden, wodurch sich der Vertrauensaufbau weiter verkompliziert. Durch folgende Defizite wird dieser Ansatz nicht für dynamische Föderationen herangezogen:

- Die Eingabe der EntityID des IdPs und die Generierung eines Codes sind nicht benutzerfreundlich. Dies ist insbesondere relevant, da viele Benutzer nicht wissen, was ein Identity Provider ist. Die Verwendung der EntityID kann folglich zu weiteren Problemen führen, wodurch die Anforderung [NFA-Usability] nicht erfüllt wird. Der zusätzliche Schritt bei IdP-Proxys verkompliziert den Workflow unnötig.

- Die Felder `MetaAdd` und `ReturnTo` sind unnötig, da die entsprechenden Informationen bereits in den Metadaten sowie in den Nachrichten für die Lokalisierung des IdPs verschickt werden.

- Die Einteilung in trusted, semi-trusted und untrusted mit den NIST LoA 1 bis 4 ist grobkörnig und bildet beispielsweise keine Föderationen der Forschungsgruppen und der Inter-Föderation eduGAIN ab.

- Außerdem macht dieses Konzept die Verwendung einer meist zusätzlichen Datenbank für jede Entität notwendig.

3.4.5. Forschungsansatz TSP

Beim Ansatz Trust Service Provider von Jian Jiang et al. [JDL+11] muss sich jede Entität beim zentralen Dienst des TSP registrieren und seine Metadaten hochladen. Diese TTP dient dazu, das Vertrauen zwischen zwei Entitäten zur Laufzeit auszuhandeln. Die Metadaten werden von der TSP geholt und können im Cache der Entität vorgehalten werden. Wenn ein Nutzer eines IdPs einen Dienst beim Service Provider nutzen will, überprüft der jeweilige SP seinen lokalen Cache, ob er die Metadaten des IdPs besitzt. Wenn dies nicht der Fall ist, sendet der Service Provider eine Anfrage an den TSP. Anschließend wiederholt der Identity Provider diesen Vorgang, um die Metadaten des SPs zu erhalten. Um die Metadaten aktuell zu halten, werden sie mit einer Versionsnummer versehen. Gleichzeitig ist der TSP ein Lokalisierungsdienst innerhalb der Föderation. Wenn ein Identity Provider außerhalb der Föderation liegt, kann der Service Provider seinen Heimat-IdP als IdP-Proxy für die indi-

rekte Authentifizierung verwenden. Hierfür werden die Assertions beim Heimat-IdP des SPs zwischengespeichert. Dies ist gleichzeitig ein Defizit des Ansatzes. Zum einen hat nicht jede Organisation die Rollen von sowohl einen Service Provider als auch einen Identity Provider, zum anderen werden so persönliche Daten bei einer weiteren Entität vorgehalten. Bei einer Vielzahl an internationalen Forschungsgruppen und der Inter-Föderation eduGAIN ist dieser Ansatz daher suboptimal.

3.4.6. Bewertung der Forschungsansätze

Es existieren weitere Ansätze in der Forschung, wie beispielsweise

- *Efficient Trust and Identity Management System for Federated Service Providers* von Makarand V. Bhonsle et al. [BPM13] beschreibt den Vertrauensaufbau zwischen zwei Service Providern ohne einer Trusted Third Party. Diese geschieht anhand einer Erweiterung der Methode Automated Trust Negotiations in einem Netzwerk, indem sich alle Service Provider bereits kennen und Informationen über die benötigten Attribute anderweitig ausgetauscht werden. Zwar ist die Architektur dynamisch, aber insbesondere die Initialisierung und die Aufnahme weiterer Entitäten ist in einem großen Netzwerk schwierig zu realisieren, da alle beteiligten Entitäten und ihre Vertrauensbeziehungen bekannt sein müssen. Ferner handelt es sich um Föderationen mit reinen Service Provider, die ihre Informationen gegenseitig austauschen, aber nicht um Föderationen mit Identity Providern. Die Problematik der unterschiedlichen Schemata wird nicht behandelt.

- Der Ansatz von *Trust Based Model for Federated Identity Architecture to Migitate Identity Theft* von Eghbal Ghazizadeh et al. [GZAM$^+$12] verwendet Trusted Computing, Federated Identity Management und OpenID Web SSO, um Identitätsdiebstahl im Cloud Computing zu vermeiden. Dazu muss das Vertrauen zwischen allen Beteiligten, Personen, Entitäten sowie Geräten auf Basis von OpenID mit SAML Tokens über verschlüsselte Nachrichten hergestellt werden. Während der Nutzer Kontrolle über die Weitergabe seiner Attribute durch die Verwendung eines Personal Identity Portals hat, werden andere Aspekte der Vertrauensbildung nicht betrachtet.

Diese Forschungsansätze konzentrieren sich ebenfalls auf einen bestimmten Ausschnitt und betrachten nicht das Gesamtbild mit seinen Defiziten. Diese Konzentrierung auf einen spezifischen Bereich ist bei der Bewertung der Forschungsansätze in den Tabellen 3.9 und 3.10 deutlich zu sehen. Dynamic Identity Federation ist hier als DIF abgekürzt.

Anforderung	Priorität	DIMDS	FAMTN	IdMRep	DIF	TSP
Funktionale Anforderungen						
[FA-Aktualisierung]	2	o	-	+	o	o
[FA-Attributswahl]	2	-	-	-	-	-
[FA-Automatisierung]	2	-	o	+	o	o
[FA-Dynamik]	2	-	o	+	+	o
[FA-Föderation]	1	o	o	-	o	+
[FA-Grenzüberschreitend]	1	-	o	+	o	o
[FA-Identitätswahl]	3	o	-	-	-	-
[FA-Initiierung]	2	o	-	-	+	+
[FA-Integration]	1	o	-	-	o	o
[FA-Interaktion]	1	-	-	-	o	-
[FA-Konfiguration]	1	-	-	o	+	-
[FA-LoA]	2	-	-	+	o	-
[FA-Lokalisierung]	1	+	o	-	+	+
[FA-LoT]	2	-	-	+	-	-
[FA-Realisierbarkeit]	1	-	-	o	o	o
[FA-Reichweite]	2	-	o	+	+	o
Nichtfunktionale technische Anforderungen						
[NFA-Koexistenz]	1	o	o	+	+	o
[NFA-Performanz]	2	-	-	o	o	o
[NFA-Skalierbarkeit]	1	-	o	+	-	-
[NFA-Usability]	2	-	-	-	-	o

Tabelle 3.9.: Bewertung 1/2 der Forschungsansätze

Anforderung	Priorität	DIMDS	FAMTN	IdMRep	DIF	TSP
Sicherheitsanforderungen						
[SEC-Authentifizierung]	1	+	o	o	+	+
[SEC-Automatisierung]	2	-	-	+	+	o
[SEC-Datenübertragung]	1	o	-	-	o	o
[SEC-LoA]	2	-	-	+	+	-
[SEC-LoT]	2	-	-	+	-	-
[SEC-Multilateral]	1	-	-	-	-	-
[SEC-Systemsicherheit]	1	-	-	-	-	o
Organisatorische Anforderungen						
[ORG-Automatisierung]	2	-	o	o	-	o
[ORG-Föderation]	2	-	-	-	o	o
[ORG-Konfiguration]	2	-	-	o	o	-
[ORG-LoA]	2	-	-	o	o	-
[ORG-LoT]	2	-	-	o	-	-
[ORG-Migration]	2	o	-	-	-	o
[ORG-Realisierbarkeit]	1	-	-	o	o	o
[ORG-Registrierung]	1	-	-	o	o	+
[ORG-Validierung]	1	-	-	-	-	-
Datenschutzanforderungen						
[DSA-Datenschutz]	1	-	-	-	o	-
[DSA-Initiierung]	3	o	-	+	+	o
[DSA-Interaktion]	2	-	-	-	-	-
[DSA-LoT]	2	-	-	+	-	-
[DSA-Selbstbestimmung]	1	o	-	-	o	o
[DSA-Zustimmung]	2	-	-	-	o	o

Tabelle 3.10.: Bewertung 2/2 der Forschungsergebnisse

3.5. Forschungsansätze zur Interoperabilität von Attributen

In den Inter-Föderationen eduGAIN und KALMAR2 werden bei den einzelnen teilnehmenden Föderationen unterschiedliche Schemata eingesetzt, was bedeutet, dass Attribute eine andere Syntax und eine verschiedenartige Semantik haben können. Auch außerhalb von eduGAIN besteht diese Problematik. Zudem können andersartige FIM-Protokolle als SAML, wie beispielsweise OpenID Connect, eingesetzt werden. Daher wurden unterschiedliche Ansätze zur Interoperabilität entwickelt. Im Folgenden werden ontologische Ansätze und die Forschungsansätze von Credential Conversion Service (CCS) und Federation Schema Correlation Service (FSCS) erläutert.

3.5.1. Ontologische Ansätze

Beim Austausch von Information gibt es allgemein drei Probleme:

- *Syntax* bzw. Datenformat betreffend,
- *Struktur*, äquivalent zu Homonymen, Synonymen oder unterschiedlichen Attributen in Datenbank-Tabellen, und
- *Semantik*, wie die beabsichtigte Bedeutung von Ausdrücken in einem speziellen Zusammenhang.

Während ein bilateraler Konflikt in der Regel ein einzelnes Objekt betrifft, wie beispielsweise `Vorname` in Entität A und `Firstname` in Entität B, beziehen sich multilaterale Konflikte auf mehrere Ebenen. So kann der Name einer Person unterschiedlich zusammen gesetzt werden, wie `Vorname` plus `Nachname`, während bei anderen Entitäten mehrere oder überhaupt keine Nachnamen eingetragen werden. Zusätzlich kommen Meta-Level-Konflikte vor, die sich auf die Verwendung unterschiedlicher Modellierungselemente stützen. Zur Lösung dieser Konflikte können Semantiken beschrieben und verglichen werden. Diese unterscheiden sich anhand der Ausdrucksstärke. So können einfache Hierarchien von Ausdrücken ausreichen, um Informationen auszutauschen. Komplexe Konzeptbeschreibungen werden *Ontologien* genannt. Sie haben unter anderem das Ziel, über explizite formale Spezifikation von Abhängigkeiten und Beziehungen die Syntax und Semantik von Datenmodellen zu definieren.
Ontology Mapping bezeichnet das aufeinander Abbilden von zwei oder mehreren Ontologien. Es gibt drei Architektur-Ansätze, um Informationen mit einer einheitlichen Semantik darzustellen.

- Der *Single-Ontologie-Ansatz* verwendet eine einzige globale Ontologie, die ein gemeinsames genutztes Vokabular zur Verfügung stellt. Daher müssen alle integrierten Informationsquellen die Informationen mit dem einheitlichen Vokabular der globalen Ontologie repräsentieren können. Dazu ist eine gleiche Sicht auf die Begriffe notwendig.

- Im Gegensatz dazu verwendet jede Ressource beim *Mehrfach-Ontologie-Ansatz* eine eigene Ontologie.

- Der *hybride Ansatz* verbindet beide Ansätze, indem jede Ressource eine eigene lokale Ontologie verwendet, die auf einem gemeinsamen globalen Vokabular basiert. Werden verschiedene hybride Ansätze miteinander verknüpft, muss ein ontologisches Mapping durchgeführt werden.

Es existieren mehrere Ansätze Ontologien auf das Identity Management anzuwenden, wie beispielsweise:

- Im Ansatz von Farah Layouni und Yann Pollet [LP09] werden drei unterschiedliche Kreise von Benutzerdaten gebildet – öffentlich, Bank und Telekommunikation. Jeder dieser Kreise repräsentiert Konzepte und Funktionalitäten, während die Schichten Model und Instances die Abstraktion der Objekte darstellen. Die Klasse Person ist wiederum der Schlüssel des Modells und enthält verschiedene Attribute, wie Name und Adresse. Dieser Ansatz erlaubt es Benutzer zu beschreiben, jedoch lässt er sich nicht direkt auf die momentan vorhandenen Föderationen im R&E übertragen, da beispielsweise Bankdaten normalerweise nicht gespeichert werden, aber das öffentliche Profil nicht nur öffentliche, für jeden zugängliche Daten enthält.

- Gail-Joon Ahn [AS11] erläutert die Verwendung von Links für finanzielles Risiko und persönliches Risiko in einer Identity Attribute Ontologie zur Beschreibung des Benutzers in sozialen Netzwerken. Dabei wird pro Attribut das finanzielle und persönliche Risiko bewertet. Während dieser Ansatz eine Person wiedergibt, ist die Risikobewertung pro Attribut zu ungenau. So kann sich das Risiko bei der Kombination bestimmter Attribute, beispielsweise Kreditkartennummer und Prüfsumme, erhöhen oder sinken, während die Prüfsumme alleine kaum ein finanzielles Risiko darstellt.

- Der Ansatz von Christian Emig et al. [ELBA07] beschreibt die Konvertierung von Attributsnamen zwischen Quellsystem und Zielsystem anhand einer einzelnen (single) Ontologie, die person ontology genannt wird. Dieses Vorgehen hat das Ziel den manuellen Aufwand bei der Konfiguration von Konnektoren zu minimieren. Auf Grund der divergierenden Schemata in den einzelnen Entitäten, beispielsweise in eduGAIN, ist es fraglich, ob eine einzelne Ontologie ausreicht, um alle Fälle abzudecken.

3.5.2. Forschungsansatz CCS

Der Ansatz *Credential Conversion Service* von Óscar Cánovas et al. [CLGS04] schlägt einen CCS-Dienst zur Integration von externen Authorisierungsschemata in Szenarien vor, bei denen SAML das Standard-Protokoll ist. Hierfür werden durch den zentralen Dienst nicht-SAML Assertions, wie beispielsweise von Zertifikaten, in SAML Assertions umgewandelt. Dazu muss der Nutzer angeben, was sein target scenario ist bzw. welche Credentials er um-

wandeln will. CCS definiert zwei Profile, Push Conversion und Pull Conversion, um Regeln zu definieren, wie Assertions in ein Protokoll eingefügt oder herausgenommen werden. Die umgewandelten Assertions können anschließend entweder beim Nutzer, bei der Heimatorganisation oder in einem Repository gespeichert werden.

eduGAIN Credential Conversion Service (eCCS) [LMCRLGS07] erweitert Credential Conversion Service für eduGAIN, welcher beim Metadata Distribution Service angesiedelt wird. Dabei konzentriert sich der Ansatz auf die Schemata `SCHAC` und `eduPerson`. Im ersten Schritt muss für jede Föderation festgelegt werden, wie die Attribute zwischen internem Schema und eCCS umgewandelt werden müssen. Ein Benutzer kann nun einen Dienst in einer anderen Föderation auswählen. eCCS muss nun bestimmen, welche Heimatorganisation diese Attribute hat und ob sie umgewandelt werden müssen. Der Service Provider erhält diese Informationen mitsamt den ursprünglichen Attributen der Heimatorganisation und kann diese durch eCCS in das gewünschte Format konvertieren.

Beiden Ansätzen ist gemein, dass sie sich auf ein festgelegtes Szenario beschränken. CCS betrachtet rein die Konvertierung von Zertifikatsinformationen zu SAML Assertions, wobei hier nicht festgelegt ist, welches Schema bzw. welche Semantik und welche Syntax vom Service Provider erwünscht ist. eCCS konzentriert sich auf die Umwandlung vom Föderationsschema zu `SCHAC` und `eduPerson`. Jedoch besitzt nicht jede Föderation, wie z. B. die schwedische Föderation SWAMID, ein eigenes Schema. Zusätzlich sind bestimmte Attribute nicht in den Schemata `SCHAC` und `eduPerson` festgelegt. Somit kann ein SP diese Attribute auch nicht anfragen. Außerhalb des festgelegten Szenarios ist eCCS schwer umzusetzen, da die Regeln für jede Konvertierung vorher durch eCCS festgelegt werden müssen. Zudem soll die Konvertierung in diesem Ansatz durch den Service Provider durchgeführt werden, die sich bisher auf den Identity Provider verlassen konnten.

3.5.3. Forschungsansatz FSCS

Wolfgang Hommel hat in seiner Dissertation [Hom07] (Abschnitt 4.4.12) ein Werkzeug namens *FSCS* spezifiziert, das den Austausch von Konvertierungsregeln innerhalb einer Föderation durch einen zentralen Dienst ermöglicht. Der Ansatz FSCS, der auf Shibboleth beruht, bietet ein Repository für Konvertierungsregeln an. Die Grundidee ist, dass ein Identity Provider, der als erster eine Regel erstellen muss, diese in das Repository hochlädt. Ein nächster IdP kann sich hierdurch Arbeit sparen, indem er nur die Regel herunterlädt und sie in die lokale Konfiguration einbindet. Die Regeln werden in der Transformationssprache Extensible Stylesheet Language Transformations (XSLT) gespeichert und können anschließend in das vom IdP verwendete XML transformiert werden. Die einzelnen Regeln lassen sich modular zusammensetzen und so innerhalb der zentralen Komponente wiederverwenden. Eine Schwierigkeit hierbei ist, dass eine Regel, die von anderen eingesetzt wird, gelöscht werden kann und somit die Konvertierung nicht mehr möglich ist. Zudem ist dieser Ansatz nur bedingt implementierungsunabhängig. Während die Regeln in Shibboleth als XML verfasst sind, verwendet SimpleSAMLphp PHP. Zudem sind in Shibboleth andere Transformationen

nativ verfügbar als in SimpleSAMLphp. Ähnliches gilt für ADFS und PySAML2, während OpenID Connect keine Aussagen über Konvertierungen trifft.

3.6. Level of Assurance

Neben den bereits erwähnten Level of Assurance von NIST gibt es weitere Klassifikationen, wie beispielsweise die Verlässlichkeitsklassen der DFN-AAI. In diesem Abschnitt werden unterschiedliche Kategorisierungen der Verbindlichkeit bzw. der Sicherheit, in aktuelle Praxis, Normen und Anwendung in Protokollen aufgeteilt, vorgestellt.

3.6.1. Level of Assurance in Föderationen

Während einzelne Föderationen mehrere Klassen haben, beschränken sich andere Föderationen auf Mindestanforderungen an alle teilnehmenden IdPs. Die DFN-AAI hat eigene Verlässlichkeitsklassen spezifiziert, wie in Abschnitt 2.3.6 erwähnt, während sich die US Föderation InCommon auf dem NIST-Standard aufbaut. Als drittes wird die finnische Haka-Föderation beschrieben, die einen Fragebogen für das Self-Assessment publiziert hat.

Die verschiedenen Verlässlichkeitsklassen der DFN-AAI werden durch unterschiedliche Metadatensätze realisiert. IdPs bestimmen selbst, welcher Verlässlichkeitsklasse sie angehören. Service Provider bestimmen im Gegenzug, welches Schutzbedürfnis ihre Ressourcen haben und legen somit fest, welchen Metadatensatz sie benötigen.

Identity Assurance Profiles in InCommon

Die US Föderation InCommon unterscheidet zwei Klassen, die Identity Assurance Profile (IAP) genannt werden: *Bronze* und *Silver*. Im Gegensatz zu *Silver* benötigt *Bronze* kein Audit. *Bronze* entspricht dem NIST Level 1, während *Silver* mit NIST Level 2 vergleichbar ist und für finanzielle Transaktionen empfohlen wird. Das *InCommon Identity Assurance Profiles Bronze and Silver* [InC13] beschreibt die Voraussetzungen für IAPs *Bronze* und *Silver*, aufgeteilt in die acht verschiedenen Bereiche:

Business Policy and Operational Criteria. Sowohl *Bronze* als auch *Silver* haben als Anforderung die Teilnahme an der Föderation InCommon, Risikomanagement, Fortführung der Klassifikation für mindestens 3 Jahre und Benachrichtigung der Föderationsverwaltung.

Registration and Identity Proofing. Während IAP *Bronze* nur den Schutz von personenbezogenen Informationen vorschreibt, ist es für IAP *Silver* u. a. wichtig, dass zusätzlich

die Identität überprüft und verifiziert wird, was außerdem schriftlich festgehalten werden muss.

Credential Technology. IAP *Bronze* benötigt neben einer eindeutigen ID einen einfachen Schutz gegen das Erraten des Passwortes, während IAP *Silver* einen starken Schutz beansprucht.

Credential Issuance and Management. Berechtigungen müssen für beide Klassen herausgegeben, zurückgezogen und erneuert werden können. Zusätzlich muss bei IAP *Silver* die Herausgabe der Berechtigung dokumentiert werden.

Authentication Process. IAP *Bronze* und *Silver* unterscheiden sich nicht bei der Authentifizierung. Beide Profile müssen Schutz vor verschiedenen Angriffen, wie Replay-Angriffe und Abhören, bieten.

Identity Information Management. Falls nicht alle Kriterien mit den in IAP genannten übereinstimmen, muss der IdP in *Bronze* und *Silver* Mechanismen haben, ein geeignetes IAP zu bestimmen.

Assertion Content. Sowohl für IAP *Bronze* als auch IAP *Silver* müssen Attribute dem Standard entsprechen, kryptografische Sicherheit vorhanden sein und das von der Föderation herausgegebene IAP in Assertions verschickt werden.

Technical Environment. Die größten Unterschiede liegen in der technischen Umgebung, bei dem IAP *Silver* zu regelmäßigen Softwareupdates, Netzwerksicherheit, physischen Sicherheit sowie verlässlichen Service verpflichtet.

Nach der Prüfung der Anforderungen fügt die Föderationsverwaltung einen Vermerk in die Metadaten hinzu, damit SPs die offizielle Zugehörigkeit eines IdPs zu einem Identity Assurance Profile überprüfen können. Anschließend erhält der Identity Provider ein neues Zertifikat, welches für drei Jahre gültig ist. Auch wenn das Vertrauensschema gut durchdacht aussieht, wird es kaum verwendet, wie die Statistik von Januar 2016 [InC16] zeigt: 5 Mal *Bronze* und 1 Mal *Silver*.

Self-Assessment in Haka

Bevor IdPs in die finnische Haka-Föderation aufgenommen werden, müssen sie zumindest die Pflichtfelder eines Fragebogens zur Selbsteinschätzung ihres Levels ausfüllen [Mik12]. Der Fragebogen enthält Fragen bezüglich der Prozesse, Daten und Workflows, um die Qualität und Aktualität der Daten und Server für das Identitätsmanagement zu beurteilen. Um an der Föderation teilnehmen zu können, muss der Identity Provider mindestens Level 3 nach Beantwortung der verpflichtenden Fragen in allen zwölf Kategorien erreichen. Die Kategorien umfassen u. a. die folgenden Bereiche:

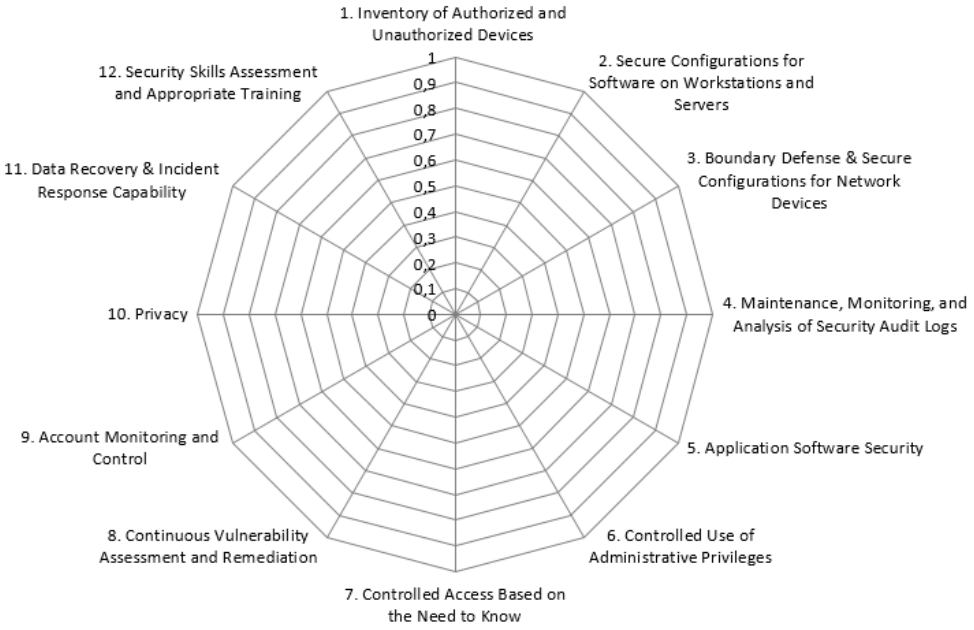

Abbildung 3.10.: Grafische Aufbereitung des Self-Assessments in der Föderation Haka [Mik12]

- Inventarisierung von autorisierten und nicht-autorisierten Geräten,
- sichere Konfiguration der Software auf Workstations und Servern,
- sichere Konfiguration von Geräten im Netzwerk und Netzwerksicherheit,
- Wartung, Monitoring und Analyse von Auditlogs,
- Softwaresicherheit und
- Verwendung von Berechtigungen durch Administratoren.

Wie in Abbildung 3.10 zu sehen, wird das erreichte Level grafisch aufgezeigt und anschließend von der Föderationsverwaltung ausgewertet. Durch das Self-Assessment wird ein Mindestmaß an Vertrauen zwischen den Teilnehmern der Föderation gewährleistet.

Bewertung der Level of Assurance in den Föderationen

Diese drei exemplarischen Beispiele für Verlässlichkeitsklassen in Föderationen veranschaulichen, dass Föderation unterschiedliche Anforderungen stellen. In eduGAIN existieren zudem Föderationen, die, ebenso wie die Inter-Föderation selbst, kein Level of Assurance verwenden. Als Beispiele hierfür sind die österreichische AAI Austrian Academic Computer Network (ACOnet) sowie die SWITCHaai genannt. Die UK Federation hat Anforderungen an die Entitäten, die bei der Registrierung überprüft werden. In der nachfolgenden Tabelle 4.28 werden die Voraussetzungen bei der DFN-AAI, InCommon, Haka, SWAMID und UK Federation gegenüber gestellt. Die Darstellung zeigt, dass die Anforderungen sehr unterschiedlich sind. Gemeinsamkeiten sind insbesondere bei der Identifizierung und Authentifizierung zu sehen, während es starke Abweichung beim Betrieb der Software und Hardware gibt.

3.6.2. Normen zu Level of Assurance

Parallel zu der aktuellen Praxis in den nationalen Föderationen existieren verschiedene Normen, die meistens das Vertrauen in vier Kategorien aufschlüsseln. Am bekanntesten ist das Konzept von NIST LoA, auf welches Secure identity across borders linked (STORK) Quality Authentication Assurance (QAA) verweist. Ferner existieren eine Norm der Standardisierungsorganisation ISO/IEC und der Kantara Arbeitsgruppe. Innerhalb einer IETF Working Group wird aktuell an einem neuen Ansatz gearbeitet. Außerdem wurde während der Arbeit an dieser Arbeit eine neue Verordnung der EU erlassen.

NIST Electronic Authentication Guide

In der NIST Special Publication 800-63-2 [BDN+13] werden vier Levels für die Verlässlichkeit spezifiziert[2]. Das Level 1 gilt als das niedrigste, während Level 4 das höchste ist. Die Level of Assurance basieren auf der Leitlinie des Federal Office of Management and Budget (OMB) zu *E-Authentication Guidance for Federal Agencies*. Die Leitlinie beschreibt ebenfalls vier Level of Assurance, die wie folgt definiert wird:

> *"... 1) the degree of confidence in the vetting process used to establish the identity of the individual to whom the credential was issued, and 2) the degree of confidence that the individual who uses the credential is the individual to whom the credential was issued."* [Bol03]

[2]Die am 8. Mai 2016 veröffentliche Neuerung wurde auf Grund des zeitlichen Rahmens nicht beachtet. Die Neuerung zeigt, dass NIST in Zukunft die einzelnen Aspekte Identity Proofing, Authenticators und Federated Assertions getrennt voneinander bewertet, äquivalent zum Ansatz von Vectors of Trust. Anstelle von 4 Levels besitzen die Aspekte 3 Stufen. `http://nstic.blogs.govdelivery.com/2016/05/08/announcing-draft-special-publication-800-63-3-digital-authentication-guideline/` [Online, abgerufen am 09.05.2016].

Kategorie	DFN-AAI		InCommon		Haka	UK Federation
	Basic	Advanced	Bronze	Silver		
Identifizierung	eindeutige Adresse	Ausweis, eindeutig identifizierbar	eindeutige Adresse, Schutz	eindeutig identifizierbar, Schutz, Doku	eindeutig identifizierbar, Schutz	soll eindeutig identifizierbar sein
Authentifizierung	digitale Adresse	persönliches Konto, Passwort	digitale Adresse, Schutz gegen Erraten des Passworts	starker Schutz, persönliches Konto	persönliches Konto, Passwort	-
Angriffsschutz	-	-	ja	ja	ja	ja
Software	-	-	Assertion verschlüsselt, gegen Angriffe geschützt	Assertion verschlüsselt, gegen Angriffe geschützt	kurze Lebensdauer der Assertion, aktuelle Metadaten, Doku	-
Hardware	-	-	-	Updates, Sicherheit	viele Vorgaben	-
Datenhaltung	3 Monate	2 Wochen	72 Stunden	72 Stunden, Doku	regelmäßig	aktuell
Risikomanagement	-	-	ja	ja	ja	-
Datenschutz	Recht	Recht	-	-	explizit	explizit
Audit	nein	nein	nein	ja	nein	möglich

Tabelle 3.11.: Level of Assurance in ausgewählten NRENs

Dieser Ansatz wurde von NIST erweitert. So beziehen sich die Anforderungen an die IdPs nicht nur auf die Überprüfung der Identität und die Registrierung, sondern auf die folgenden Bereiche:

- Tokens, d. h. Schlüssel, Einmalpasswörter oder Passwort, inklusive Token Management,
- Registrierung und Überprüfung der Identität,
- sowie das Authentifizierungsprotokoll an sich, u. a. mit Anforderung an die Qualität des Passworts, Schutz gegen Angriffen sowie Assertions.

Der Schutz vor Angriffen, wie Replay und Man-in-the-middle, wird in jedem dieser Bereich betrachtet. Daraus ergeben sich Mindestanforderungen an die vier Level of Assurance, wie beispielsweise:

Level 1: Für LoA 1 ist keine Überprüfung der Identität notwendig, jedoch muss der Benutzer authentifiziert werden. Das Passwort muss verschlüsselt übertragen werden. Zur Authentifizierung ist ein einfaches Challenge-Response-Protokoll möglich.

Level 2: LoA 2 verlangt eine einfache Überprüfung der Identität. Die Authentifizierung muss mindestens mit einem Passwort oder einer Personal Identification Number (PIN) geschehen, während bessere Mechanismen auch akzeptiert werden. Wichtig ist ferner ein sicheres Authentifizierungsprotokoll.

Level 3: Die Anforderungen für Level 3 beinhalten eine Multi-Faktor-Authentifizierung, wofür ein Schlüssel oder Einmalpasswort verwendet werden kann. Zudem ist die Verifikation der Identität notwendig.

Level 4: LoA 4 fordert zusätzlich eine starke kryptographische Authentifizierung, bei der ein Schlüssel notwendig ist.

Dies zeigt, dass sich NIST LoA, wie bereits die Leitlinie von OMB, auf Aspekte der Authentifizierung konzentriert, während andere Faktoren, beispielsweise die technische Infrastruktur, Management-Prozesse und die Aktualität und Qualität der Daten, nicht betrachtet werden.

STORK Quality Authenticator scheme

Das STORK Projekt hat eine europäische elektronische Identität (eID) Plattform zum Ziel, um so den Einwohnern die Nutzung von Diensten über Grenzen hinweg mit Hilfe einer nationalen eID zu ermöglichen. Dazu wird grenzübergreifende Authentifizierung bereitgestellt und innerhalb von fünf Pilotprojekten getestet. Eine Thematik dabei ist das Level of Assurance [HLE09], QAA genannt. Das QAA Framework von STORK umfasst vier Levels für die

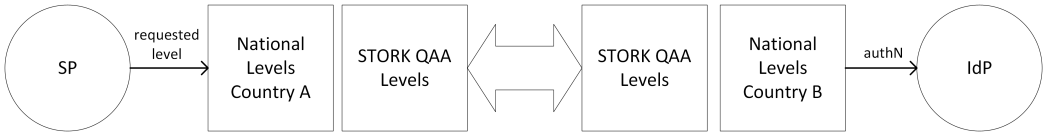

Abbildung 3.11.: Mapping der nationalen Levels durch QAA nach [HLE09]

Authentifizierung und zum Mapping nationaler Levels. Bei der Festlegung der Levels werden organisatorische und technische Aspekte bedacht:

Organisatorische Aspekte: Qualität der Identifikation des Antragstellers, Prozess der Ausstellung des Tokens, Qualität der ausstellenden Behörde.

Technische Aspekte: Authentifizierung mit den Faktoren Art und Robustheit des Tokens, Qualität des Mechanismus zur Authentifizierung.

Aus den unterschiedlichen Stufen der Aspekte ergibt sich die Zusammensetzung der QAA Levels wie folgt:

STORK QAA Level 1: Keine oder minimale Sicherheit. Credentials werden beispielsweise ohne Prüfung akzeptiert, E-Mail-Adressen werden nur auf Korrektheit überprüft.

STORK QAA Level 2: Geringe Sicherheit. Die Identität muss überprüft werden. Zudem muss das Token mit Garantien bezüglich Sicherheit und Genauigkeit geliefert werden.

STORK QAA Level 3: Umfangreiche Sicherheit. Registrierung der Identität mit Überprüfung der Identität des Antragsstellers. Ausstellende Organisation wird von der Regierung kontrolliert oder akkreditiert. Die Credentials entsprechen Zertifikate.

STORK QAA Level 4: Hohe Sicherheit. Die physikalische Präsenz des Antragstellers ist bei der Registrierung oder bei einem späteren physischen Treffen notwendig. Das Zertifikat ist ein hartes, zum Beispiel in Form einer Smartcard.

Das Mapping zwischen verschiedenen nationalen Levels soll durch einen Proxy oder eine Middleware anhand der Zuweisung der einzelnen nationalen Levels zu QAA Levels geschehen, wie in der Abbildung 3.11 dargestellt. Diese Zuordnung wurde anhand der Kriterien der QAA Levels getätigt und ist somit fest. Jedoch ergeben sich auch Schwierigkeiten daraus:

- Einige Mitgliederstaaten haben nur QAA Level 4, während andere Mitgliedsstaaten kein Level äquivalent zu QAA Level 4 besitzen, wodurch Bewohner letzterer Länder nicht in der Lage sind Dienste aus erst genannten Ländern zu nutzen.

- Einige Mitgliederstaaten haben mehrere nationale Levels, die in STORK einem einzigen Level entsprechen. Gleichzeitig gibt es Mitgliederstaaten, die ein nationales Level

aufweisen, welches mehreren Levels in STORK entspricht.

Im Gegensatz zu NIST LoA betrachtet STORK QAA auch organisatorische Aspekte. Trotzdem werden weniger bzw. andere Bereiche analysiert als in den nationalen Föderationen, die im vorherigen Abschnitt vorgestellt wurden.

ISO/IEC 29115:2013

Die Norm *ISO/IEC 29115:2013* [ISO13] spezifiziert ebenfalls vier Level of Assurance. Zusätzlich enthält die Norm eine Richtlinie zum Mappen von einem beliebigen Schema in das von ISO/IEC. Die Norm betrachtet dabei die Aspekte Technik sowie Management und Organisation für die Spezifikation der Levels:

- Technische Gesichtspunkte:
  - Enrolment phase: Anmeldung und Initialisierung, Registrierung, Verifikation und Überprüfung der Identität, Aufbewahrung und Aufbewahrungsfrist.
  - Credential management phase: Lebenszyklus der Credentials mit u. a. den Phasen Erstellen, Initialisierung, Binding, Aktivierung, Speicherung und Erneuerung.
  - Entity authentication phase: Authentifizierung und Logging.
- Management & Organizational: z. B. Etablierung des Dienstes, rechtliche und vertragliche Erfüllung, Informationssicherheit und externe Dienstkomponenten.

Insbesondere die Phasen Registrierung, Credential Management und Authentifizierung sind bei der Festlegung der Verlässlichkeitsklassen essentiell:

Level of assurance 1 (LoA1): Gewisses Vertrauen, dass der Nutzer über konsekutive Authentifizierungen derselbe ist. Es bestehen keine Anforderungen bezüglich der Authentifizierung und der Verschlüsselung.

Level of assurance 2 (LoA2): Gewisses Vertrauen in die beteuerte Identität. Einfache Authentifizierung ist akzeptabel, jedoch soll ein sicheres Authentifizierungsprotokoll verwendet werden. Die Effektivität von Angriffen, insbesondere von Abhören und Erraten von Passwörtern, soll durch geeignete Kontrollen reduziert werden.

Level of assurance 3 (LoA3): Hohes Vertrauen in die beteuerte Identität. Multi-Faktor-Authentifizierung soll verwendet werden, während die Informationen, die bei der Authentifizierung ausgetauscht werden, kryptographisch geschützt sein müssen.

Level of assurance 4 (LoA4): Sehr hohes Vertrauen in die beteuerte Identität. Zusätzlich zu den Anforderungen für LoA3 muss die Identität der Person überprüft und mani-

pulationssichere Hardware zur Speicherung aller Schlüssel verwendet werden. Zudem sind alle sensiblen Informationen, wie personenbezogene Daten, kryptographisch zu schützen.

Wenn miteinander kooperierende Organisationen unterschiedliche LoA-Schemata verwenden, müssen laut [ISO13] für jedes Schema die Kriterien separat definiert und kommuniziert werden. Um ein Schema zum ISO-Standard und somit zu einem anderen Schema zu vergleichen, soll jede Organisation eine Methodologie für das eigene Schema entwickeln und publizieren, in der erklärt wird, wie das eigene Schema und die dort spezifizierten Level of Assurance zu denen aus dem ISO-Standard sich verhalten. Dabei sollen folgende Aspekte betrachtet werden:

- Gefahren pro Level,
- Auswirkung der Gefahren,
- Identifizierung der Gefahren, welche pro Level kontrolliert werden müssen,
- Technologien, die für die Sicherheit eingesetzt werden müssen und
- Kriterien zur Bestimmung der Äquivalenz von verschiedenen Kombinationen.

Kantara Identity Assurance Framework

Die *Assurance Level (AL)* von Kantara basieren auf dem Identity Assurance Framework, welches von Mitgliedern verschiedener Sektoren entwickelt wurde. Das Framework besteht neben den Assurance Levels u. a. aus einem Glossar, Assurance Assessment Schema und assoziierten Profilen. Ebenso, wie die oben genannten Normen, baut Kantara auf vier verschiedene Levels, AL, auf, die ähnlich zu den bisher beschriebenen sind.

Assurance Level 1 hat nur minimale Kriterien, beispielsweise für eine Registrierung bei einer Nachrichtenseite:

Organisation: Minimale Kriterien. Die Organisation muss nur dem geltenden Recht nach Datenschutz einhalten.

Identitätsprüfung: Self Assertation.

Credential Management: PIN und Passwort.

Assurance Level 2 hat bereits eine Identitätsprüfung mit einem Ausweis, ansonsten sind die Kriterien ebenfalls eher minimal.

Organisation: Moderate Kriterien. Die Organisation muss genügend finanzielle Mittel vor-

weisen können, um einen sicheren Betrieb für einen längeren Zeitraum gewährleisten zu können. Nutzungsrichtlinien müssen aufbewahrt werden. Zudem wird ein Information Security Management erwartet.

Identitätsprüfung: Moderate Kriterien.

Credential Management: Single-Faktor und geschützt durch Protokoll.

Assurance Level 3 benötigt noch stärkere Identitätsprüfung und Verifikation. Hier wird nun auch Multi-Faktor-Authentifizierung eingesetzt, wobei die Tokens noch undefiniert sind.

Organisation: Strikte Kriterien. Das Risk Assessment Review muss alle sechs Monate durchgeführt werden. Zudem werden jedes Jahr ein internes Service Audit und die Verwendung eines Information Security Management Systems erwartet.

Identitätsprüfung: Strikte Kriterien.

Credential Management: Multi-Faktor Authentifizierung und kryptografisches Protokoll.

Assurance Level 4 schränkt auf Hard-Token ein und ist auch bei den weiteren Kriterien strikter:

Organisation: Zudem muss im operativen Bereich das Information Security Management System zertifiziert sein.

Identitätsprüfung: Noch striktere Kriterien.

Credential Management: Hard-Token und Schlüssel für Authentifizierung.

Um ein Kantara AL zu erhalten, muss man auditiert und somit akkreditiert werden. Dies hat beispielsweise die Föderation Swedish Academic Identity (SWAMID) gemacht.

Vectors of Trust

Vectors of Trust (VoT) [RJ15] ist aktuell, seit Juni 2015, ein I-D bei der IETF, der die Interoperabilität von verschiedenen LoAs durch Vektoren beheben will. Die Hautpverantwortlichen hierfür sind Leif Johansson und Justin Richer. Der Vektor von VoT ist dabei ein skalares Produkt mit orthogonalen Aspekten. Dabei werden drei Komponenten betrachtet: Identity Proofing, Credential Binding und Assertion Presentation.

Identity Proofing (IPV) gibt hier bei wieder, wie die Überprüfung der Identität und der Attribute geschieht:

P0: Keine Überprüfung, Daten müssen nicht konsistent über Sessions sein.

P1: Attribute sind self-asserted, aber konsistent über die Zeit.

P2: Identität wurde entweder persönlich oder über vertrauenswürdige Mechanismen (wie Social Proofing) überprüft.

P3: Gesetzliche oder vertragsmäßige Beziehung zwischen IdP und Nutzer.

Die *Primary Credential Usage* beschreibt die Kategorien des Credentials, das vielleicht (`MAY`) verwendet wird:

C0: Keine Credentials bzw. anonymer Service.

Ca: Einfache Session Cookies.

Cb: Bekanntes Gerät.

Cc: Credentials wie Benutzername und Passwort.

Cd: Verwendung von Schlüsseln.

Ce: Verwendung von asymmetrischen Schlüsseln.

Cf: Versiegelte Hardware-Tokens, vertrauenswürdige Biometrik oder Verwendung von Trusted Platform Module (TPM)-Schlüsseln.

Das *Credential Management (CM)* beschreibt die Verwaltung des Credentials, welches vielleicht verwendet wird:

Ma: Self-Asserted Credentials bzw. keine zusätzliche Verifikation.

Mb: Herausgabe und Rotation bzw. Verwendung von Backup Recover Credentials oder Löschung nach Wunsch des Nutzers.

Mc: Verifikation für Herausgabe und Rotation bzw. Widerrufen bei verdächtigen Aktivitäten.

Assertion Presentation (AP) untergliedert die Art, wie eine Assertion verschickt wird, beispielsweise verschlüsselt, signiert und ohne Schutz:

Aa: Kein Schutz bzw. unsignierter Bearer-Identifier, wie Session Cookie.

Ab: Signierter und verifizierter Token, über Browser versendet.

Ac: Signierter und verifizierter Token, über einen anderen Kanal versendet.

Ad: Verschlüsseltes Token, welches vor Dritten geschützt ist und nur über den Schlüssel der

RP geöffnet werden kann.

Alle vier Dimensionen, und weitere, die in zukünftigen Arbeiten definiert werden können, werden zu einem Vektor kombiniert, der über die Metadaten kommuniziert wird. Der Vektor kann beispielsweise durch `P1:Cb:Mb:Ab` repräsentiert werden. Die Anwendung in den Metadaten ist äquivalent zu üblichen LoAs mit dem Unterschied, dass eine URN zur Repräsentation verwendet wird.

eIDAS

Die Durchführungsverordnung (EU) 2015/1502 der Kommission vom 8. September 2015 zur Festlegung von Mindestanforderungen an technische Spezifikationen und Verfahren für Sicherheitsniveaus elektronischer Identifizierungsmittel gemäß Artikel 8 Absatz 3 der Verordnung (EU) Nr. 910/2014 des Europäischen Parlaments und des Rates über elektronische Identifizierung und Vertrauensdienste für elektronische Transaktionen im Binnenmarkt bildet die deutsche Umsetzung der auch als *Electronic identification and trust services (eIDAS)* [ER14] bezeichneten Verordnung. Diese Verordnung hält sich an die internationale Norm ISO/IEC 29115 und berücksichtigt zudem die im Projekt STORK spezifizierten QAA. In den Mitgliedstaaten angewandte bewährte Verfahren in Bezug auf Sicherheitsniveaus sollen eingerechnet werden. Dasselbe gilt für die Direktive 95/94/EC. Im Gegensatz zu den genannten Normen sollen unterschiedliche Beweismittel für die Identitätsprüfung zugelassen werden, wie Register, Urkunden und Stellen. Die Sicherheitsniveaus werden hier als *"niedrig"*, *"substanziell"* und *"hoch"* festgelegt. Diese Sicherheitsniveaus werden in den folgenden Elementen betrachtet und müssen von jedem Element erfüllt werden, um das entsprechende Niveau zu erreichen:

Anmeldung: Beantragung und Eintragung, Identitätsnachweis und -überprüfung sowie Verknüpfung von elektronischen Identifizierungsmitteln natürlicher und juristischer Personen.

Verwaltung der elektronischen Identifizierungsmittel: Merkmale und Gestaltung elektronischer Identifizierungsmittel, Ausstellung, Auslieferung und Aktivierung, Aussetzen, Widerruf und Reaktivierung, Verlängerung und Ersetzung.

Authentifizierung: Authentifizierungsmechanismus.

Management und Organisation: Allgemeine Bestimmungen, Veröffentlichte Bekanntmachungen und Benutzerinformationen, Informationsmanagement, Aufbewahrungsfristen, Einrichtungen und Personal, Technische Kontrollen, Einhaltung und Prüfung.

Auffallend dabei ist, dass bereits bei einem niedrigen Sicherheitsniveau interne Audits gefordert werden, während bei vielen Elementen die Anforderungen für substanziell und hoch äquivalent zu den Anforderungen für niedrig sind.

Bewertung der Normen

Insgesamt zeigt sich, dass alle drei erst genannten Normen gleichartig aufgebaut sind (vergleiche Tabelle 3.12). Es werden je vier Levels spezifiziert, die ähnliche Mindestanforderungen haben. Bei keinem der Standards wird ein Grund für die Anzahl der Levels genannt. Die größten Unterschiede existieren beim niedrigsten Level. Während NIST bei Level 1 beispielsweise die verschlüsselte Übertragung des Passwortes fordert, hat LoA1 bei ISO/IEC 29115 keinerlei Anforderungen diesbezüglich.

Bei den nachfolgend beschriebenen Normen zeigt sich folgendes:

- Kantara orientiert sich ebenfalls an den vorhandenen Normen, hat aber einen stärkeren Schwerpunkt auf interne Prozesse.
- eIDAS enthält nur drei Stufen, wobei die niedrigste Stufe höher angesiedelt ist, als die der anderen Normen. Auch hier wird der Fokus zusätzlich auf interne Prozesse und die Organisation gelegt.
- VoT stellt demgegenüber eine Ausnahme dar, da nun statt einem gesamten Level verschiedene Komponenten getrennt voneinander betrachtet werden. Dies erleichtert zum einen die Einordnung und den Vergleich, zum anderen werden jedoch nicht alle Aspekte betrachtet, die Teil der Normen sind.

3.6.3. Anwendungen in den FIM-Protokollen

Wie in Abschnitt 3.1 bereits beschrieben, kann unter OpenID 2.0 die Erweiterung PAPE bzw. ein Claim in OpenID Connect verwendet werden, um den Kontext der Authentifizierung zu beschreiben. In SAML ermöglicht das *SAML V2.0 Identity Assurance Profile* die Angabe der Authentifizierung mittels XML sowie die Nennung in den Metadaten. In beiden FIM-Standards ist es somit möglich, jedes beliebige Schema für LoA verwendet werden. Um die Suche und Wiederverwendung zu vereinfachen, erstellt `[RFC6711]` [Joh12] der Standardisierungsorganisation IETF einen Eintrag bei der Internet Assigned Numbers Authority (IANA) Registry für *Level of Assurance Profiles*, die dann in den Protokollen angewandt werden können. Dafür muss die URI des LoA Profiles, wie in OpenID verwendet, und die für SAML benötigte Context Class gespeichert werden. Weitere obligatorische Informationen sind der Name des Levels sowie die URL zur Dokumentationsseite des Schemas.

3.7. Zusammenfassung und Bewertung

Die in den Abschnitten 3.1 und 3.2 vorgestellten FIM-Standards und ihre Implementierungen spiegeln den aktuellen Stand der Technik wieder und werden bereits in der Praxis

Kategorie	NIST LoA				STORK QAA				ISO/IEC 29115:2013			
	LoA1	LoA2	LoA3	LoA4	QAA1	QAA2	QAA3	QAA4	LoA1	LoA2	LoA3	LoA4
Tokens	Passwort 6 Zeichen, Pin 4 Ziffern	Passwort 8 Zeichen, Pin 6 Ziffern	Einmal-passwort oder Token	Schlüssel	Passwort oder PIN	gutes Passwort oder guter PIN	Token	Schlüssel	keine Angabe			Zertifikat
Speicherung	einfach verschlüsselt	verschlüsselt und z. B. gesalzen	verschlüsselt	mehrere Sicherheitsmaßnahmen	-	-	-	-	-	Schutz gegen Angriff	verschlüsselt	mehrere Sicherheitsmaßnahmen
Datenhaltung	-	72 Stunden	24 Stunden	24 Stunden	-	-	-	-	-	-	-	-
Registrierung	keine Überprüfung	einfache Überprüfung	Verifizierung	persönlich	keine Überprüfung	Überprüfung	Verifizierung	persönlich	keine Überprüfung	Überprüfung	Verifizierung	persönlich
Authentifizierung	verschlüsselt, z. B. Kerberos	sicher, z. B. TLS	Multi-Faktor, z. B. über TLS	mit hartem Token	-	sicher	Multi-Faktor	mit hartem Token	-	Single-Faktor	Multi-Faktor	mit hartem Token
Angriffe	Replay, Raten	Eavesdropping, Session hijacking	Phishing	Man in the middle	keine	wenige	die meisten	alle	passend zur Risikoevaluation			
Assertions	Schutz gegen Wiederverwendung	Schutz gegen Manipulation	signierte Assertions, kurze Lebenszeit	u. a. Logging	-	-	-	-	keine Angabe		kryptographisch gesichert	
Security Management	-	-	-	-	-	-	-	-	-	dokumentiert	Information Security Management System	

Tabelle 3.12.: Übersicht über die Normen NIST LoA, STORK QAA und ISO/IEC 29115

bei Föderationen und Kollaborationen eingesetzt. Jedoch wurde anhand des in Kapitel 2 erarbeiteten Kriterienkatalogs auch aufgezeigt, dass nicht alle essentiellen Anforderungen vollständig erfüllt wurden und es noch Anstrengungen benötigt, die wichtigen und empfehlenswerten Anforderungen umzusetzen. Die Tabellen 3.13 und 3.14 zeigen zusammenfassend die Bewertungen der Standards SAML, OpenID Connect sowie der Implementierungen Shibboleth, SimpleSAMLphp, PySAML2 und ADFS. Defizite bestehen vor allem in den folgenden Bereichen:

- *Skalierbarkeit* der Föderationen und des Vertrauensaufbaus, wie bereits in Kapitel 2 aufgezeigt. Dieser Aspekt hängt mit der Architektur und Organisation der bestehenden Föderationen zusammen und kann nicht rein technisch gelöst werden.

- Damit eng verbunden ist das *Vertrauen* innerhalb der Föderationen, welches teilweise auf dem Metadatenaustausch beruht. In Abschnitt 3.3 wurden deshalb verschiedene Ansätze zur Metadatenverwaltung vorgestellt und ihre Defizite aufgezeigt. Die größte Problematik ist die Verteilung aller Metadaten einer Föderation oder Inter-Föderation, auch wenn keine Geschäftsbeziehungen zu vielen Organisationen bestehen. Diese Lösung skaliert u. a schlecht und ist wenig performant. Die in Kapitel 2 geforderte Automatisierung des Vertrauensaufbaus und dynamische Anpassung der Föderationen kann technisch durch SAML bzw. dessen Implementierungen nicht nativ umgesetzt werden. Ein Ansatzpunkt hierfür ist der Lokalisierungsdienst, da hier die vertrauenswürdigen Identity Provider ausgewählt werden können. Das Vertrauen besteht aus weiteren Aspekten, wie in Abschnitt 3.4 dargestellt. Ein Aspekt ist der Level of Assurance. Alle Protokolle und Implementierungen bieten die Möglichkeit ein LoA anzugeben, jedoch wird das in vielen aktuellen Föderationen und Inter-Föderationen im R&E-Umfeld nicht genutzt; Beispielsweise weil jede Föderation unterschiedliche Anforderungen stellt, wie an den Föderationen DFN-AAI, InCommon, Haka und der UK federation dargestellt. In Abschnitt 3.6 wurden aktuelle Verlässlichkeitsklassen der Föderationen aufgezeigt und miteinander verglichen. Ferner wurden unterschiedliche Normen beschrieben, die verwendet werden können. Weitere Aspekte des Vertrauens sind Level of Trust und die Einhaltung des Datenschutzes, die bisher wenig Beachtung erhalten.

- Die *Interoperabilität* durch Attributskonvertierung bzw. allgemein verständliche Attribute und die Protokollunabhängigkeit bilden einen weiteren Schwachpunkt aktueller Standards und Lösungen. Durch die uneinheitlichen Schemata und die manuelle Attributskonvertierung entsteht eine Wartezeit für Benutzer sowie Aufwand für die IdPs. PySAML2 bietet aktuell die beste Unterstützung, auch wenn diese weiter optimiert werden kann. Die in Abschnitt 3.5 vorgestellten Ansätze für die Interoperabilität in Föderationen und Inter-Föderationen lösen diese Problematik nicht.

- Ein Defizit von SAML und seinen Implementierungen im Gegensatz zu OpenID Connect ist die *Usability*. OpenID Connect und UMA im Besonderen legen einen größeren Wert auf die Einbindung des Benutzers, auch wenn hier ebenfalls Verbesserungen möglich wären, beispielsweise durch eine Entscheidungshilfe für Nutzer und eine komplexere

Funktionalität des UserConsents mit der Möglichkeit des Monitorings.

- *Schnittstellen* zu vorhanden Systemen, insbesondere zu Supportsystemen, werden nicht beschrieben. Ein wichtiger Aspekt hierbei ist insbesondere das organisationsübergreifende Fehlermanagement und die multilaterale Sicherheit, die für das organisationsübergreifende FIM wünschenswert sind. Hierzu gibt es bereits Ansätze und Frameworks aus dem MNM-Team, die eingesetzt werden könnten.

In Kapitel 4 wird deshalb eine Architektur erarbeitet, die den Vertrauensaufbau zwischen Service Provider und Identity Provider vereinfacht. Zur Umsetzung werden zusätzliche Werkzeuge zur Interoperabilität, zur Abschätzung des Vertrauens und dem damit zusammenhängenden Trust Management benötigt, die in Kapitel 5 erstellt werden.

Anforderung	Priorität	SAML	OpenID	Shibboleth	SSp	PySAML2	ADFS
Funktionale Anforderungen							
[FA-Aktualisierung]	2	+	+	+	+	+	+
[FA-Attributswahl]	2	-	-	o	o	o	-
[FA-Automatisierung]	2	-	o	-	-	o	-
[FA-Datenkategorisierung]	1	+	-	+	+	+	-
[FA-Dynamik]	2	-	+	o	-	-	-
[FA-Entscheidungshilfe]	3	-	-	-	-	-	-
[FA-Fehlermanagement]	2	o	+	o	o	o	o
[FA-Föderation]	1	o	-	+	+	+	o
[FA-Grenzüberschreitend]	1	+	+	+	+	+	+
[FA-Homeless]	3	o	o	o	o	o	o
[FA-Identitätswahl]	3	o	+	o	o	o	o
[FA-Initiierung]	2	-	o	-	-	-	-
[FA-Integration]	1	-	-	+	+	+	+
[FA-Interaktion]	1	-	+	+	+	o	o
[FA-Konfiguration]	1	-	-	+	+	+	o
[FA-Konnektor]	2	-	-	+	+	+	+
[FA-Kontext]	3	+	-	+	+	+	+
[FA-Langlebigkeit]	1	+	o	+	+	o	o
[FA-LoA]	2	+	+	+	+	+	-
[FA-Lokalisierung]	1	-	+	+	o	o	o
[FA-LoT]	2	-	-	-	-	-	-
[FA-Metadaten]	2	o	o	+	+	+	o
[FA-Monitoring]	2	-	-	-	-	-	o
[FA-Pull&Push]	1	+	+	+	+	+	
[FA-Realisierbarkeit]	1	o	o	+	+	+	+
[FA-Reichweite]	2	o	o	o	+	+	o
[FA-Rollen]	2	+	o	+	+	+	+
[FA-Schema]	2	o	o	o	o	+	-
[FA-SelfAsserted]	3	-	+	-	-	-	-
[FA-SLA]	3	-	-	-	-	-	-
Nichtfunktionale technische Anforderungen							
[NFA-Dokumentation]	1	+	+	+	+	+	o
[NFA-Implementierungsunabhängigkeit]	2	+	o	o	+	+	o
[NFA-Koexistenz]	1	+	o	+	+	+	+
[NFA-OpenSource]	1	+	+	+	+	+	-
[NFA-Performanz]	2	+	+	+	+	+	o
[NFA-Portabilität]	3	+	+	+	+	+	-
[NFA-Protokollunabhängigkeit]	2	-	-	-	+	+	+
[NFA-Skalierbarkeit]	1	o	+	o	o	o	o
[NFA-Usability]	2	-	+	o	o	o	o

Tabelle 3.13.: Ergebnisse 1/2 der Analyse auf Basis des Kriterienkatalogs

Anforderung	Priorität	SAML	OpenID	Shibboleth	SSp	PySAML2	ADFS
Sicherheitsanforderungen							
[SEC-ARPs]	1	o	-	+	o	+	o
[SEC-Auditing]	2	o	o	+	+	+	+
[SEC-Authentifizierung]	1	o	+	+	+	+	+
[SEC-Automatisierung]	2	-	-	-	-	-	-
[SEC-Datenübertragung]	1	o	o	+	+	+	+
[SEC-Initiierung]	2	-	o	-	-	-	-
[SEC-Integration]	2	-	-	+	+	+	+
[SEC-Kontext]	3	o	-	o	o	+	o
[SEC-LoA]	2	o	+	+	+	+	-
[SEC-LoT]	2	-	-	-	-	-	-
[SEC-Metadaten]	3	o	-	o	o	o	-
[SEC-Multilateral]	1	-	-	o	o	o	o
[SEC-Systemsicherheit]	1	o	-	+	+	+	+
Organisatorische Anforderungen							
[ORG-Automatisierung]	2	-	-	-	-	-	o
[ORG-Föderation]	2	-	-	+	+	+	o
[ORG-Konfiguration]	2	-	-	o	o	+	-
[ORG-LoA]	2	-	-	o	o	o	-
[ORG-LoT]	2	-	-	-	-	–	-
[ORG-Metadaten]	3	-	-	o	o	o	-
[ORG-Migration]	2	-	o	+	o	o	o
[ORG-Realisierbarkeit]	1	-	o	+	o	o	o
[ORG-Registrierung]	1	-	-	o	o	o	o
[ORG-Schema]	2	-	-	o	o	+	-
[ORG-SLA]	3	-	-	-	-	-	-
[ORG-Supportprozesse]	2	-	-	-	-	-	o
[ORG-Validierung]	1	-	-	o	o	+	-
Datenschutzanforderungen							
[DSA-ARPs]	1	-	-	o	o	o	o
[DSA-CoCo]	3	-	-	o	o	o	-
[DSA-Datenschutz]	1	-	-	o	o	o	o
[DSA-Initiierung]	3	-	-	-	-	-	-
[DSA-Interaktion]	2	-	+	o	o	o	-
[DSA-LoT]	2	-	-	-	-	-	-
[DSA-Selbstbestimmung]	1	-	o	+	+	o	o
[DSA-Zustimmung]	2	-	+	+	+	o	o

Tabelle 3.14.: Ergebnisse 2/2 der Analyse auf Basis des Kriterienkatalogs

Kapitel 4

Konzept einer Architektur

Inhalt dieses Kapitels

In diesem Kapitel wird auf Basis der im vorherigen Kapitel vorgestellten Ansätze, ihrer Defizite sowie den Anforderungen, die im Kapitel 2 anhand von Szenarien aufgestellt wurden, ein Architekturkonzept erstellt, das folgende Eigenschaften haben soll:

- Föderationen sollen *dynamisch* nach den Wünschen der Nutzer gebildet werden können. Dabei ist es wichtig, dass die Erstellung von Vertrauen über Föderationsgrenzen hinaus möglich ist, was zu einem Konzept von *dynamischen virtuellen Föderationen* führt.
- Die Bildung von *Inter-Föderationen* soll ebenfalls möglich sein.
- *Schnittstellen* zur Verwaltung von Föderationen und Inter-Föderationen sollen beachtet werden.
- Zugleich soll die *Interoperabilität* in Hinblick auf unterschiedliche Schemata verbessert werden.
- Ferner sollen Schnittstellen zum *Security Management* und *Change Management* vorgesehen sein.

Im nächsten Abschnitt 4.1 wird die *Zielsetzung* dieser Architektur näher erläutert. Dabei wird die Ausgangssituation erneut betrachtet, der hypothetische Idealumfang erläutert und die nachfolgende Vorgehensweise aufgezeigt. Anschließend wird in Abschnitt 4.2 ein *Überblick über die Föderationen* der Gesamtarchitektur gegeben. Hierbei wird der Begriff der *dynamischen virtuellen Föderationen* basierend auf der neutralen Perspektive der Ausgangssituation aus Abschnitt 4.1 und den Szenarien in Kapitel 2 definiert.

Die *Gesamtarchitektur* wird anhand von verschiedenen Modellen modelliert. Innerhalb der Architektur sind unterschiedliche Rollen beteiligt. Diese werden im *Organisationsmodell* im Abschnitt 4.3 näher betrachtet. Anschließend werden im Abschnitt 4.4 die Domänen des *Informationsmodells* beschrieben, bevor sie detailliert dargestellt werden. Das *Kommunikationsmodell* im Abschnitt 4.5 umfasst generische Kommunikationsmechanismen und SAML-spezifische Kommunikationsmechanismen. Abschließend werden im Abschnitt 4.6 die ver-

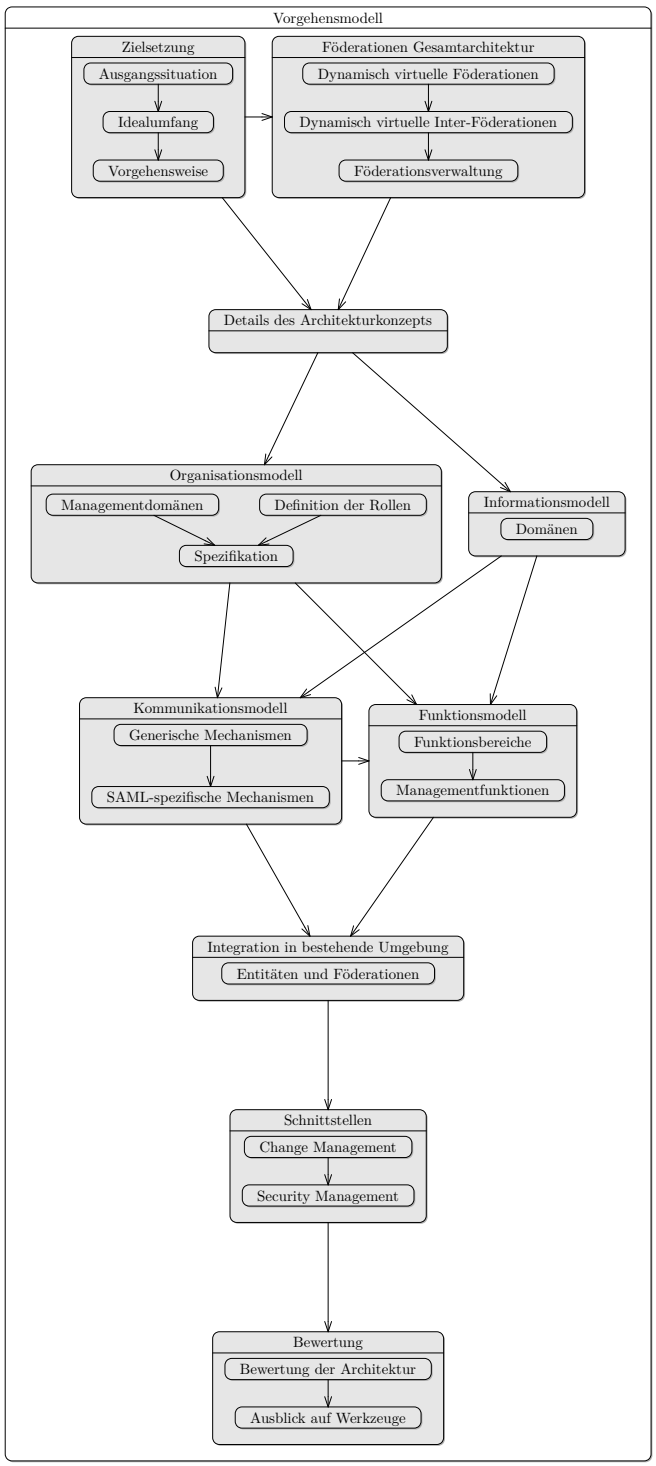

Abbildung 4.1.: Vorgehensmodell in diesem Kapitel

schiedenen *Funktionen* betrachtet, die für die Architektur relevant sind. Aufbauend auf den Funktionsbereichen werden *Managementfunktionen* beschrieben.

Durch Änderungen an der aktuellen Architektur wird ein *Vorgehen* benötigt, um die neue Architektur in die bestehende Umgebung zu integrieren. Dies wird in Abschnitt 4.7 betrachtet.

Ferner wird in Abschnitt 4.8 auf Schnittstellen zum *Security Management* und *Change Management* eingegangen. Hier, im Bereich des Security Managements, werden Auswirkungen auf die bestehende IT-Security Infrastruktur sowie Maßnahmen zur Sicherheit aufgezeigt. Die Architektur des dynamischen Metadatenaustausches und Vertrauensaufbaues hat Auswirkungen auf organisationsinterne IT Service Support Prozesse, die im Bereich des Change Managements betrachtet werden.

Abschließend wird die Architektur anhand des in Kapitel 2 erstellten Kriterienkatalogs bewertet und Möglichkeiten für weitere Werkzeuge aufgezeigt, die im folgenden Kapitel behandelt werden.

Das Vorgehen in diesem Kapitel wird durch die Abbildung 4.1 dargestellt. Die hier definierte Managementarchitektur wurde bereits in der Veröffentlichung [HP16] durch die Autorin und Wolfgang Hommel beschrieben.

4.1. Zielsetzung

Um die Ziele dieser Arbeit zu verdeutlichen, wird in diesem Abschnitt zunächst die Ausgangssituation, die in Kapitel 2 anhand von Szenarien aufgezeigt wurde, aus einer neutralen, szenarienunabhängigen Perspektive betrachtet. Dabei werden die Sichtweisen von Identity Provider, Service Provider, Nutzer und einer möglichen Föderationsverwaltung, die beispielsweise eine Community, eine nationale Föderation, ein Projekt oder eine Inter-Föderation darstellen kann, näher beleuchtet. Anschließend wird der Idealumfang mit möglichen, hypothetischen Funktionalitäten dargestellt. Der Schwerpunkt wird hier auf nicht erfüllte Anforderungen gelegt, um darauf aufbauend die Vorgehensweise in dieser Arbeit, die an dieses Ziel näher heranführt, zu erläutern. Zwangsweise werden nicht alle Aspekte behandelt, wodurch die Arbeit gegen nicht betrachtete Forschungsfelder abgegrenzt wird.

4.1.1. Ausgangssituation

Der Schwerpunkt wird in diesem Abschnitt auf die vorhandenen Defizite gelegt, um die bestehende Problematik nochmals aus verschiedenen Perspektiven zu beleuchten.

Föderationsspezifische Betrachtungsweise

Um über Identity Management innerhalb einer Föderation oder einer Inter-Föderation zu kommunizieren, ist ein gemeinsames Datenmodell notwendig, was die Semantik und Syntax der verwendeten Benutzerinformationen spezifiziert. Da es kein allgemein gültiges Schema gibt, divergieren die Schemata in den nationalen Föderationen. Darüber hinaus haben Communities eigene Anforderungen an Attribute, die nicht in den nationalen Föderationen abgebildet werden können. Zusätzlich zu diesen Schemata werden weltweit gültige Schemata wie `eduPerson` und `SCHAC` verwendet, die jedoch nicht alle benötigten Informationen enthalten. Um alle notwendigen Benutzerinformationen in der benötigten Semantik und Syntax bereitstellen zu können, wird eine Lösung benötigt. Bisher konvertiert jeder IdP die Benutzerinformationen, soweit er das Schema des SPs kennt. Wenn dies nicht der Fall ist, kann der Nutzer den Dienst im schlimmsten Fall nicht nutzen. Zudem verfügen die weltweit gültigen Schemata nur über den kleinsten gemeinsamen Nenner, wodurch manche Dienste weitere Schemata benötigen. Nachdem die SAML-Implementierungen wie auch OpenID Connect unterschiedliche Konvertierungsregeln in verschiedenen Formaten bereitstellen, ist die aktuelle Lösung suboptimal.

Darüber hinaus ist Vertrauen innerhalb einer Föderation essentiell, damit Entitäten miteinander kommunizieren. Das technische Vertrauen wird durch die Verteilung von Metadaten, die Informationen über die Kommunikationsendpunkte enthalten, geregelt. Die Metadatenverwaltung unterscheidet sich von Föderation zu Föderation und erfordert manuelle Schritte, insbesondere bei der Aufnahme von Entitäten. SPs und IdPs müssen bestimmte Anforderungen erfüllen, um überhaupt in der nationalen Föderation, aber auch in den Föderationen der Communities und der Inter-Föderation, teilnehmen zu können. Diese werden über unterschiedliche Methoden, wie Verträge, Self-Assessment oder gar Audits, überprüft. Das Vertrauen in den existierenden Implementierungen baut auf Föderationen auf und ist somit statisch, d. h. alle Mitglieder einer Föderation vertrauen einander gleich stark. Das dies in Wirklichkeit nicht unbedingt der Fall ist, wurde bereits im Szenario des LRZs in der Inter-Föderation eduGAIN sowie dem Community-Szenario von CLARIN in Kapitel 2 aufgezeigt. Das Vertrauen lässt sich lediglich statisch einschränken, wodurch zum einen viele SPs nicht die benötigten Attribute erhalten und zum anderen dynamische Mechanismen von Trust & Reputation Management nicht genutzt werden, wie auch in der Dissertation von Wolfgang Hommel [Hom07] (Abschnitt 4.1) beschrieben. Ein Abgleich verschiedener Verlässlichkeitsklassen erscheint auf Grund der Unterschiede als nicht möglich; Standards bezüglich Level of Assurance berücksichtigen die Anforderungen der Föderationen nicht und werden folglich ebenso wenig eingesetzt.

IdP-spezifische Betrachtungsweise

IdPs erhalten von den Föderationen, in denen sie teilnehmen, auf unterschiedlichen Wegen die Metadaten der Föderation. Falls die Organisation zusätzlich Mitglied einer Inter-Föderation ist, werden weitere Metadaten in die lokale Konfiguration hinzugefügt. Nach-

dem der Identity Provider nicht alle SPs verwendet, werden mehr Daten verarbeitet als benötigt. Die großen Datenmengen können zur verlangsamten Verarbeitung führen. Zudem verfügen die Föderationen über unterschiedliche Wege Metadaten der einzelnen Entitäten zu beziehen. Wenn ein IdP Mitglied in mehreren Föderationen ist, bedeutet das einen erheblichen Mehraufwand, der sich über eine allgemein-gültige Schnittstelle beseitigen lässt. Den in den Metadaten enthaltenen Entitäten soll, nach den vorhandenen Implementierungen, vertraut werden. Ob SPs überhaupt die Mindestanforderungen der IdPs erfüllen, muss manuell überprüfen werden bzw. teilweise haben IdPs keine geeigneten Möglichkeiten diese Analyse überhaupt durchzuführen. Dies wird dadurch erschwert, dass es für SPs kein Schema wie Level of Assurance gibt, um sie einzuordnen.

Nachdem die lokal gespeicherten Benutzerinformationen nicht dem Schema der Föderation entsprechen, müssen die Attribute zunächst in das vom Service Provider gewünschte Format konvertiert werden. In je mehr Föderationen der Identity Provider teilnimmt, umso mehr Regeln müssen erstellt werden, da die meisten Föderationen ein eigenes Schema erfordern. Zusätzlich ist es notwendig, dass die Attribute gefiltert werden, damit der SP nur die Benutzerinformationen erhält, die er benötigt. Diese Auswahl kann zusätzlich durch Vorgaben der Föderation, aber auch durch lokale Regeln, weiter eingeschränkt werden. Die Konfiguration, d. h. das Erstellen von Attributskonvertierungen und Attributsfiltern, ist bislang eine manuelle Aufgabe. Benutzer, die einen bestimmten Dienst nutzen wollen, aber dies auf Grund von fehlenden Attributen nicht geeignet können, müssen sich selbst an den IdP wenden, damit diese manuelle Konfiguration gestartet wird. Nachdem Nutzer meist nicht alle Informationen kennen, die der IdP hierfür benötigt, dauert es bis ein Dienst genutzt werden kann. Dieses Defizit könnte durch Automatisierung basierend auf den Informationen in Metadaten und Vertrauensinformationen behoben werden. Dabei ist zu beachten, dass Datenschutzbestimmungen eingehalten werden und der Dienst weiterhin verfügbar ist. Ferner müssen Änderungen protokolliert werden. Intern fehlt meist eine Anbindung an das Change Management, was zusätzlich auf das Security Management auch ausgeweitet werden kann.

SP-spezifische Betrachtungsweise

Der Service Provider muss für seinen FIM-fähigen Dienst im Vorhinein Attribute spezifizieren, die er benötigt. Diese sind möglichst Teil der Metadaten, damit IdPs aus dieser Information halb-automatisch ihre Konfiguration anpassen können. Da diese Konfiguration manuell erfolgt und sich, wenn beispielsweise IdP und SP in unterschiedlichen nationalen Föderationen sind, die verwendeten Schemata unterscheiden, bekommt der Service Provider nicht unbedingt alle Attribute, die er benötigt. Zusätzlich kann es der Fall sein, dass der Identity Provider nicht alle Informationen liefern kann, weil er zum Beispiel das Attribut nicht kennt oder es anders benannt ist. Ferner ist es möglich, dass der IdP zwar die angeforderten Attribute sendet, der SP diese aber nicht versteht, weil sich z. B. Semantik und Syntax unterscheiden. Als mögliche Folge suchen Nutzer nach einem ähnlichen Dienst, den sie ohne Wartezeit verwenden können.

Äquivalent zur IdP-spezifischen Betrachtungsweise ist die Metadatenverteilung aus Sicht des SPs ungenügend. Upload und Download unterscheiden sich von Föderation zu Föderation. Viele kommerziellen SPs, wie Springer Link, sind Mitglieder in verschiedenen Föderationen und müssen sich den Gegebenheiten der Föderationen anpassen, was unnötig Zeit kostet. Auf Grund der Föderationsstruktur und der eben beschriebenen Problematik erreicht der Service Provider nicht alle Nutzer, die den Dienst nutzen wollen. Dies ist insbesondere bei kommerziellen Diensten von Nachteil. Zudem ist es für SPs schwierig die Mindestanforderungen an IdPs zu überprüfen, da die Kennzeichnung der LoA bzw. der Verlässlichkeitsklassen von Föderation zu Föderation unterschiedlich ist, manche Föderationen keine geeigneten Mechanismus etabliert haben und föderationsübergreifend die Kennzeichnung noch komplexer ausfällt. Ein Service Provider, der in mehreren Föderationen teilnimmt und bestimmte Mindestanforderungen hat, kann ggf. diese nicht für jede Föderation im gleichen Maße stellen oder er schließt potentielle Nutzer aus. Ferner lassen sich die Verlässlichkeitsklassen häufig nicht einfach vergleichen, was ein weiteres Defizit ist.

Nutzerspezifische Betrachtungsweise

Der Nutzer will einen Dienst eines SPs nutzen. Nachdem er keine aktive Session hat, wählt er beim Lokalisierungsdienst seinen Identity Provider aus, falls dieser in der Liste der vertrauenswürdigen IdPs aufgelistet ist. Ist dies nicht der Fall, führen die oben beschriebenen Defizite dazu, dass der Nutzer den Dienst gar nicht, nach einer Wartezeit oder nur teilweise nutzen kann. Zudem ist er in der Pflicht seinen Identity Provider über die gewünschte Nutzung eines Dienstes zu informieren, falls ein Fehler auftritt. Da keine geeignete Schnittstelle vorhanden ist bzw. keine verständliche Fehlermeldungen erscheint, verzögert sich die Änderung der Konfiguration. Dem Nutzer sind meist nicht alle erforderlichen Informationen, wie beispielsweise die EntityID des SPs, bekannt. Dies wiederum verzögert die Nutzung des Dienstes, da der IdP entweder weitere Informationen anfordern oder selbst versuchen muss, auf diese Informationen zu schließen.

Wenn die Nutzung eines Dienstes möglich ist, sendet der Identity Provider des Nutzers die Attribute an den Service Provider. Bevor dieser Schritt abgeschlossen wird, kann sich der Nutzer über das Tool uApprove bzw. uApprove.jp die zu übermittelnden Daten anzeigen lassen und der Weitergabe zustimmen oder diese ablehnen. Auf Grund der Funktionalitäten bei uApprove ist dem Nutzer nicht möglich über die Filterung der Attribute mit zu entscheiden. Ferner kann er nicht, wie bei UMA, eigene Attribute verwalten, um beispielsweise die Nutzung eines Dienstes zu beschleunigen oder zu personalisieren. Der einfach zugängliche Überblick über die so übermittelnden Daten mit z. B. dem Widerruf von Freigaben fehlt meistens.

4.1.2. Idealumfang

Nachfolgend wird der Idealzustand beschrieben, der erreicht werden kann, wenn die Architektur auf dynamischen virtuellen Föderationen und Inter-Föderationen aufbaut, wodurch die Ordnung der existierenden festen Föderationen und Inter-Föderationen, als auch ihre Schemata abgelöst wird. Dieser Zustand ist auf Grund der gegebenen gewachsenen organisatorischen Strukturen, aber auch dem aktuellen Stand der Technik, unrealistisch. Der Idealzustand wird durch diese Arbeit somit auch nicht vollständig erreicht werden. Die Ergänzungen helfen jedoch zur Defizitminimierung. Die Beschreibung des Idealumfangs dient zum einen zur Formulierung von Randbedingungen, die bei den in dieser Arbeit zu entwickelnden Ergänzungen beachtet werden müssen, um eine passende Erweiterung möglich zu machen. Andererseits motiviert der Idealumfang weitere Arbeiten, die auf dieser aufbauen können. Die folgende Beschreibung geschieht aus Sicht von Föderationen, IdPs, SPs sowie Nutzern.

Föderationsspezifische Betrachtungsweise

Um den Anforderungen von den verschiedenartigen Föderationen, vergleiche Klassifikation von Föderationen in Kapitel 2, gerecht zu werden und die bisherigen Defizite bezüglich fester Föderationen und aggregiertem Metadatenaustausch auszugleichen, müssen dynamische virtuelle Föderationen und Inter-Föderationen etabliert werden. Diese sollen ein loser Zusammenschluss von IdPs und SPs bzw. Föderationen sein, die miteinander kooperieren. Diese Art von Föderation kann zum Beispiel nur kurzfristig für ein Projekt bestehen, aber auch langfristig angesetzt sein, wenn beispielsweise nationale Kooperationen entstehen. Der Begriff der dynamischen virtuellen Föderation wird Abschnitt 4.2 genauer definiert.

Um das Vertrauen zwischen den Teilnehmern entsprechend aufzubauen, benötigt es neben dem technischen Vertrauen in Form von Metadaten zusätzlich grundlegende Informationen über die Verlässlichkeit der Entitäten. Die eher statischen Informationen, wie Metadaten und die grundlegenden Informationen über die Verlässlichkeit, müssen in geeigneter Form dynamisch zu Beginn einer möglichen Kooperation ausgetauscht werden. Darauf aufbauend können die dynamischen Informationen eine weitere Quelle zur Entscheidungsfindung darstellen. Nachdem die bisherigen Verlässlichkeitsklassen sich teilweise stark unterscheiden und die Standards im Bereich LoA die Anforderungen nicht erfüllen, muss ein neues Trust Management diese Defizite beseitigen.

Für Föderationen soll die Metadatenverwaltung möglichst automatisiert ablaufen, damit an dieser Stelle keine Latenzen entstehen. Entitäten sollen dadurch aktuelle Metadaten laden können und nicht stundenlang warten müssen. Trotzdem soll es möglich sein, dass z. B. nationale Föderationen oder internationale Strukturen, wie die Inter-Föderation eduGAIN, weiterhin eine gewisse Verwaltungsfunktion übernehmen, indem es beispielsweise eine Schnittstelle für Policies gibt. Grundsätzliche Entscheidung über die Mindestanforderungen der Dienstgüte bezüglich LoA, LoT und ähnlichen Klassifikationen soll möglich sein, um

u. a. unverlässliche IdPs oder SPs ablehnen zu können. Hierfür ist ein Administrationstool notwendig, in das eine Benutzerverwaltung integriert ist. Zusätzlich können zum Beispiel ein Dashboard und diversen Statistiken einen visuellen Überblick über die Föderation bieten. Diese Funktionalitäten fehlen in den aktuellen Föderationen.

IdP-spezifische Betrachtungsweise

Für Identity Provider ist es wichtig, dass der Aufwand zur manuellen Konfiguration minimiert wird. Um eine Dienstnutzung und damit bilaterale Föderationen on demand zu erreichen, muss die Konfiguration automatisiert werden. Dafür müssen alle benötigten Informationen über Metadaten und ggf. über einen zentralen Dienst bereitgestellt werden. Folglich soll die Attributskonvertierung automatisch stattfinden, egal welches Schema der Service Provider verwendet. Dazu passend muss eine Filterung der Attribute erstellt werden. Der Aufwand soll sich nicht stark erhöhen, wenn der IdP an mehreren Föderationen teilnimmt. Falls der Nutzer einen Dienst außerhalb der Föderation des IdPs nutzen will, müssen erst manuell die Metadaten ausgetauscht werden, bevor überhaupt die Informationen zur Konfiguration bezüglich der Benutzerinformationen benötigt werden. Dieser manuelle Schritt gilt es ebenso zu automatisieren, wobei hier Vertrauen und Datenschutz beachtet werden müssen. Zudem sollen nur die benötigten Metadaten ausgetauscht werden, um eine bessere Skalierbarkeit bezüglich der Metadaten zu erreichen.

Zusätzlich soll der IdP die Möglichkeiten haben die Automatisierung an seinen Bedürfnissen anpassen zu können. Dies ist normalerweise über Konfigurationsdateien möglich. Der Identity Provider hat hier die Möglichkeit bei bestimmten SPs, z. B. kommerziellen oder wenig vertrauenswürdigen, vorher eine manuelle Zustimmung zu verlangen. Folglich ist es wichtig, dass der IdP auch Mindestanforderungen an potentielle Service Provider stellen kann, wie beispielsweise bestimmte Datenschutzrichtlinien. Dies soll über eine Art Trust Management geschehen, welches zu entwickeln ist. Für die dynamischen Änderungen an der Konfiguration müssen mögliche Schnittstellen zum Change und Security Management betrachtet werden. Zudem werden die Änderungen entsprechend dokumentiert, um bei einem Audit einen Überblick zu haben.

SP-spezifische Betrachtungsweise

Service Provider sollen alle benötigten Attribute erhalten. Gleichzeitig sollen die Attribute dem Schema entsprechen, welches vom SP verwendet wird. Daher soll der Service Provider vorher angeben, welche Benutzerinformationen er in welcher Semantik und Syntax für seinen Dienst benötigt. Dies soll unabhängig von existierenden Föderationen geschehen, damit alle potentiellen Nutzer einen interessanten Dienst verwenden können. Über diese Angaben soll es dem Identity Provider möglich sein, die Benutzerinformationen, unabhängig von der IdP-Software, in das geeignete Format zu konvertieren.

Äquivalent zur IdP-spezifischen Betrachtungsweise soll es für Service Provider möglich sein Mindestanforderungen an IdPs, beispielsweise in Form von Level of Assurance, zu stellen. Diese Verlässlichkeitsklassen müssen feingranular sein, um als Vorgabe Bestand zu haben. Gleichzeitig darf die Verwendung diese Verlässlichkeitsklassen nicht zu umständlich sein. Um verschiedene Anforderungen zu vergleichen, soll das zu entwickelnde Trust Management einen Abgleich erlauben. Ferner sollen für SPs ebenfalls passende Schnittstellen, beispielsweise für das Change Management und das Security Management, in die Betrachtung einbezogen werden.

Nutzerspezifische Betrachtungsweise

Der Nutzer soll, um aktuelle Defizite auszugleichen, den gewünschten Dienst möglichst unverzüglich und vollständig nutzen können. Dies soll unabhängig von existierenden nationalen Föderationen und verwendeten Schemata möglich sein. Dies erfordert einen dynamischen und benutzerfreundlich dargestellten Austausch von Metadaten, welcher durch den Nutzer initiiert wird. In den Metadaten müssen alle benötigten Informationen vorhanden sein, um die Konfiguration des IdPs anzupassen. Anschließend muss auch die Konfiguration des SPs angepasst werden. Mögliche Anforderungen an das Vertrauen sollen dabei beachtet werden.

Zugleich erhält der Nutzer die Macht selbst zu entscheiden, welchem Service Provider er vertraut, indem er den Vertrauensaufbau initiiert. Dabei kann der Benutzer selbst weitere Attribute an den SP senden und erhält einen besseren Überblick über seine personenbezogenen Daten, die bei den jeweiligen SPs gespeichert sind und die er passend verwalten kann. Ferner soll der Nutzer die Möglichkeit haben eigene Attribute zu verwalten und nach eigenen Regeln freizugeben.

4.1.3. Vorgehensweise

Basierend auf dem aktuellen Stand, den Anforderungen und dem eben vorgestellten Idealzustand wird in der zu entwickelnden Architektur auf folgende Rahmenbedingungen aufgesetzt:

- Die bereits in den Organisationen vorhandenen I&AM-Systeme sowie die darauf aufbauende FIM-Software werden als gegeben angesehen. Die zu entwickelnde Architektur wird auf den im Einsatz befindlichen Programmen aufbauen. Während I&AM-Systeme als ausgereift gelten, gibt es noch Defizite bei der FIM-Software, insbesondere bezüglich Kooperationen zwischen nationalen Föderationen, d. h. bei Inter-FIM. Die Ansätze aus der Literatur, aufgezeigt in Kapitel 3, bieten zwar interessante Ansätze, die jedoch allesamt Defizite aufweisen. So werden einzelne Aspekte betrachtet, wie beispielsweise die Metadatenverwaltung oder die Interoperabilität, jedoch fehlt ein ganzheitlicher Ansatz, der die gesamte Architektur verbessert.

- Die Architektur soll mit passenden Werkzeugen entwickelt werden, die ein dynamisches

Federated Identity Management erlauben. Im dynamischen FIM sollen Nutzer den Metadatenaustausch zwischen Identity Provider und Service Provider anstoßen können. Um den dynamischen Ansatz zu wahren, soll die Konfiguration der Entitäten automatisiert erfolgen, wobei vorhandene Anforderungen bezüglich Vertrauen und weitere Konfigurationen der Automatisierung beachtet werden müssen.

- Aktuelle nationale Föderationen, Inter-Föderationen und Communities sichern durch ihre Einschränkungen bei der Teilnahme einen gewissen Qualitätsstandard innerhalb des Zusammenschlusses. Ferner können diese Föderationsverwaltungen weitere Vorgaben, wie Policies, den Ausschluss bestimmter Entitäten und spezielle Datenschutzbestimmungen, erstellen, die für die Funktion von Föderationen für den jeweiligen Nutzerkreis sinnvoll sind. Folglich sollen nationale Föderationen, Inter-Föderationen und Communities nicht komplett aufgelöst werden, sondern eine Verwaltungsfunktion behalten, wofür geeignete Schnittstellen geschaffen werden müssen.

Aktuell bestehende FIM-Software ist nicht ausreichend, um alle Anforderungen, die in den Szenarien herausgearbeitet wurden, zu erfüllen. Daher werden im Folgenden die Konzepte der dynamischen virtuellen Föderationen und Inter-Föderationen beschrieben, die zur Automatisierung und Dynamisierung der Entstehung von Kooperation sowie für die Bildung von bedarfsgerechten Föderationen notwendig sind. Zur Realisierung dieser Konzepte wird die dafür notwendige Architektur beschrieben, die auf etablierten Strukturen aufbaut. Dabei muss beachtet werden, dass die Verwaltungsfunktion aktueller Föderationen erhalten bleibt.

Diese generelle Architektur und ihre Werkzeuge sind protokollunabhängig, um einen möglichst großen Nutzerkreis zu erschließen. Das Vorgehen reflektiert das Ziel die neuen Komponenten und die Architektur der dynamischen virtuellen Föderationen nahtlos in die bestehenden Systeme zu integrieren. Schnittstellen zur FIM-Software sollen dies ermöglichen. Die Betrachtungen des Change Managements und des Security Managements zur Einbindung in organisationsinterne Prozesse runden dieses Kapitel ab.

Bei der Konzeption werden die folgenden Teilziele bzw. Anforderungen beachtet:

- Nahtlose Integration in bestehende Strukturen und verwendete FIM-Software.
- Automatisierung der bestehenden Abläufe mit dem Ziel ein dynamisches FIM zu erreichen [FA-Dynamik].
- Hierfür sollen dynamische virtuelle Föderationen definiert und eine Managementarchitektur entwickelt werden, mit der dynamisches FIM möglich ist.
- Dieses Konzept soll hinsichtlich der Teilnehmer und des Metadatenaustausches skalierbar sein [NFA-Skalierbarkeit].
- Klassische Föderationen sollen abgebildet werden, um Parallellösungen zu vermeiden.

- Sowohl bei klassischen als auch bei dynamischen virtuellen Föderationen soll der Lebenszyklus abgebildet werden können.
- Berücksichtigung aller beteiligten Entitäten.
- Einbindung des Trust Managements mit beispielsweise den Klassifikationen LoA und LoT, um den dynamischen Metadatenaustausch sicherer zu gestalten.
- Sicherheit auch ohne Trust Management gewährleisten.
- Minimierung der bestehenden Defizite wie [FA-Reichweite] und [FA-Schema] durch ein generisches Konzept für alle Föderationen, Sektoren und Protokolle sowie eine Möglichkeit Benutzerinformationen effizient und Schema-unabhängig zu versenden.
- Anwendbarkeit und Wiederverwendbarkeit, u. a. durch Dokumentation.

Die Automatisierung dient dazu ein dynamisches FIM zu etablieren und bezieht sich dabei auf die folgenden Aspekte:

Automatisierung des Metadatenaustausches: Im Gegensatz zu den bisher vorab aggregierten und ausgetauschten Metadaten, soll der Metadatenaustausch dynamisch und automatisiert geschehen. Wenn ein Nutzer einen Dienst verwenden will und sich IdP und SP technisch nicht kennen, sollen die Metadaten über einen geeigneten Dienst automatisch, on demand ausgetauscht werden. Hierbei muss darauf geachtet werden, dass der Ansatz gut skaliert (vergleiche Anforderungen in Kapitel 2). Dies bedarf andere Workflows und somit auch andere Protokolle. Um die Metadaten in die lokale Konfiguration zu integrieren, werden Erweiterungen der IdP- und SP-Software benötigt. Damit die Automatisierung des Metadatenaustausches orchestriert werden kann, ist eine Managementplattform notwendig, für die in diesem Kapitel verschiedene Modelle aufgestellt werden. Die zentrale Komponente der Architektur dient auch dazu Entitäten zu registrieren und eine Verwaltungsmöglichkeit für Föderationen zu schaffen.

Automatisierung der Metadatenverwaltung: Die Metadatenverwaltung geschieht bisher auf unterschiedlicher Weise. Um dieses Defizit zu minimieren, werden Metadaten automatisch zwischen Entitäten ausgetauscht. Damit Entitäten keinen Mehraufwand durch die neue Architektur haben, sollen sie weiterhin ihre Metadaten lokal verwalten können. Falls dies nicht gewünscht ist, kann ein zentrales Verwaltungstool eingesetzt werden, welches Bestandteil der Managementarchitektur ist. Die mehrmalige Verwaltung von Metadaten soll vermieden werden, um nicht erneut einen hohen Aufwand und Inkonsistenzen zu riskieren.

Automatisierung der Bildung von Föderationen: Durch den automatischen Metadatenaustausch lassen sich Föderationen automatisch bilden. Bei bilateralem Metadatenaustausch ist das eine bilaterale Föderation. Um größere Föderationen zu bilden, soll das Konzept von dynamischen virtuellen Föderationen und Inter-Föderationen erstellt

werden. Diese Art von Föderationen können bei genügend hohem Anteil an bilateralen Kooperationen gebildet werden. Wenn anschließend eine festere Form von Föderation erwünscht ist, soll dies über die Managementplattform der Managementarchitektur ermöglicht werden, welche den Lebenszyklus von Föderationen abbildet. Ebenso wie bei festen Föderationen sollen sich dynamische virtuelle Föderationen und Inter-Föderationen auflösen lassen, beispielsweise wenn die Anzahl der Kooperationen unter ein bestimmtes Mindestmaß sinkt.

Automatisierung der Konvertierung: Bisher müssen Identity Provider Konvertierungsregeln für die Benutzerinformation von Hand einpflegen, falls sie die angefragten Informationen haben. Über generische Konvertierungsregeln, die sich an die Implementierung anpassen lassen, und einen gegebenenfalls notwendigen zentralen Dienst soll die Konvertierung von Benutzerinformationen automatisiert werden. Dies soll unabhängig von den verwendeten Schemata geschehen. Hierfür ist eine geeignete generische Darstellung notwendig, die zu konzipieren ist. Um die Konvertierung von Benutzerinformationen zu automatisieren, wird eine Erweiterung der IdP-Software benötigt.

Automatisierung der Konfiguration: Um die Metadaten und die Konvertierungsregeln anwenden zu können, muss die lokale Konfiguration der Entitäten angepasst werden. Damit hier kein Nutzer warten muss und keine zusätzliche Arbeit entsteht, soll dies automatisiert geschehen. Falls nicht genügend Informationen hierfür vorhanden sind, sollen diese fehlenden Informationen über ein zentrales Tool bereitgestellt werden. Auf Seiten der Entitäten ist hierfür eine Erweiterung der lokalen Software nötig. Die Konfiguration soll unabhängig von der Implementierung und somit für beispielsweise OpenID Connect, Shibboleth und SimpleSAMLphp möglich sein. Dabei ist zu beachten, dass Datenschutzbestimmungen eingehalten werden und der Dienst weiterhin verfügbar ist. Ferner müssen Änderungen protokolliert werden.

Automatisierung des Vertrauensabgleichs: Föderationen verwenden aktuell unterschiedliche Verlässlichkeitsklassen oder Level of Assurance. Um eine Vergleichbarkeit zu erreichen, soll ein generisches Format entwickelt werden, was entweder die vorhandenen Klassen ersetzt oder die Vergleichbarkeit herstellt. Damit nicht nur Service Provider IdPs abschätzen können, soll dieses Trust Management auch eine Abschätzung von SPs ermöglichen. Beide Vergleiche sollen automatisch vor dem Metadatenaustausch stattfinden, um dem Schutzbedarf der Entitäten zu genügen.

Anpassung der Automatisierung: Um den Anforderungen der Entitäten zu genügen, soll die eben genannte Automatisierung anpassbar sein. Beispielsweise kann hier ein IdP angeben, dass er allen SPs, die mindestens diese Anforderungen erfüllen, automatisch vertraut, während er für Service Provider, die nicht diese Anforderungen erfüllen, informiert werden will, um dann manuell zu entscheiden. Die Anpassung der Automatisierung wirkt sich auf den Grad der Automatisierung aus und macht, wenn zu restriktiv eingesetzt, die Automatisierung zunichte. Gleichzeitig ermöglicht die Anpassung der Automatisierung eine höhere Akzeptanz der Lösung.

Schnittstellen zum Change Management: Die durch das automatische FIM getätigten Änderungen in der lokalen Konfiguration müssen im Change Management beachtet werden. Dies wird im Abschnitt 4.8 näher betrachtet.

Da die Lösung niemals alle Defizite beseitigen kann, werden folgende Bereiche bewusst ausgeklammert und können Ansatzpunkt für weitere Arbeiten bilden:

- Die grafischen Oberflächen sowie deren Usability, insbesondere in Hinblick auf die Lokalisierung des IdPs, aber auch bei der Initiierung einer Kooperation, werden nicht näher analysiert. Für Untersuchungen in diesem Bereich sind spezielle Tests nötig, die außerhalb des Fokus liegen, da sie nur eine einzelne Anforderung direkt betreffen.

- Virtuelle Organisationen im Bereich des Grids verwenden mehrheitlich andere Protokolle und Technologien als SAML. Gleichzeitig arbeiten mehrere Forschungsprojekte an der Interoperabilität für diese Community, wodurch auf eine Konzeption von Plattformen für VOs und deren Einbindung verzichtet wird. VOs können jedoch die einheitliche Schnittstelle zur TTP einsetzen, die in dieser Arbeit konzipiert wird.

- Für ein dynamisches Trust & Reputation Management werden lediglich Schnittstellen bereitgestellt bzw. eine Erweiterbarkeit wird bei der Konzeption beachtet, da es bereits Konzepte für ein Trust & Reputation Management gibt, die verwendet oder erweitert werden können (siehe Latifa Boursas [Bou09]). Gleichzeitig soll das Trust Management eine genügende Sicherheit bieten, um das Vertrauen zwischen Entitäten einzuschätzen. Diese eher statischen Informationen vermindern die Gefahr eines Bottlenecks.

- Ein zentrales Account Linking wird auf Grund der hierbei gespeicherten persönlichen Informationen und der damit erhöhten Sicherheitsanforderungen verworfen. Ein SP-seitiges Account Linking ist dem vorzuziehen, da hierdurch nicht unnötig viele Benutzerinformationen gespeichert werden und das Angriffsziel minimiert wird. Da der SP bei dieser Art der Verknüpfung von Benutzerkonten den Kontext erhält, wird die Anforderung Kontext nicht weiter betrachtet.

Für die im Folgenden konzipierte Architektur werden neue Werkzeuge entworfen, die im Kapitel 5 genauer beschrieben werden.

4.2. Föderationen der Gesamtarchitektur

In diesem Abschnitt wird auf die Gesamtarchitektur der in dieser Arbeit zu konzipierenden Lösung eingegangen. Zunächst werden die vorhandenen und benötigten Arten von Föderationen und Inter-Föderationen beschrieben, bevor dynamische virtuelle Föderationen und eine Föderationsverwaltung erläutert werden. Basierend darauf und den nachfolgenden Modellen werden im nächsten Kapitel möglichen Architekturmustern aufgezeigt, um anschließend eine optimale Auswahl zu treffen.

4.2.1. Dynamische virtuelle Föderationen

Definition 6. Dynamische virtuelle Föderation

Dynamische virtuelle Föderation ist der Zusammenschluss von Entitäten, der dynamisch gebildet wird und parallel zu festen Föderationen bestehen kann.

Durch eine mögliche Automatisierung des Metadatenaustausch sind statische Föderationen nicht mehr notwendig, um FIM für Kooperationen einsetzen zu können. Neben den festen Föderationen, können auch *dynamische virtuelle Föderationen und Inter-Föderationen* realisiert werden, die auf dynamischen Metadatenaustausch setzen. Die dynamischen virtuellen Föderationen besitzen die folgenden Eigenschaften:

dynamisch. Der Vertrauensaufbau geschieht dynamisch, on demand, nach Bedarf der Nutzer. Der zweite Aspekt, der hier eine Rolle spielt, ist das dynamische Verhalten aus Sicht der Föderationen. Es muss möglich sein, dass sie

- dynamisch gebildet werden,
- sich dynamisch an die Bedürfnisse anpassen, d. h. dass Organisationen aus der Föderation austreten oder ihr beitreten können,
- folglich, dass die sich Größe der Föderation ändern kann,
- aber auch, dass die Föderation wieder beendet werden kann.

Der Grad der Dynamik bezüglich Föderationen hängt u. a. von der Kategorie der Föderation ab, vergleiche Klassifikation von Föderation in Kapitel 2. So ist eine Projekt-Föderation auf eine kurzfristige Lebensdauer angelegt und hat mehr Dynamik als beispielsweise eine nationale Föderation. Ein weiterer Faktor ist die Möglichkeit einer Föderation beizutreten. Wenn die Föderation abgeschlossen ist, kann sich an der Größe der Föderation nichts ändern, während eine offene Föderation mit Beschränkung oder eine allgemein offene Föderation ohne jegliche Beschränkung eine größere Dynamik bezüglich der Größe aufweist. Falls die Kooperation finanzielle Auswirkungen hat, z. B. wenn der Service Provider einen kommerziellen Dienst betreibt, sind in der Regel SLAs notwendig, die vor dem Vertrauensaufbau ausgehandelt werden müssen, was die Dynamik bremst. Zusätzlich kann es sein, dass ein IdP oder SP aus anderen, nicht bekannten Gründen der Kooperation explizit zustimmen will oder muss, was ebenfalls einen hemmenden Faktor bezüglich der Dynamik darstellt.

virtuell. Die Föderation muss nicht einer reellen, nationalen Föderation entsprechen, sondern kann aus Teilnehmern aus verschiedenen Föderationen bestehen, die miteinander kooperieren. Somit werden die bisherigen festen Strukturen aufgelöst oder aufgeweicht, um effizientere internationale Kooperationen zu ermöglichen.

Abbildung 4.2.: Klassifikation von dynamischen virtuellen Föderationen

Föderation. Durch die Eigenschaften dynamisch und virtuell wird der feste Föderationsbegriff aufgelöst. Eine Föderation ist ein Zusammenschluss von Teilnehmern, die auf Grund des Bedarfs miteinander kooperieren und gegebenenfalls eine Föderationsverwaltung benötigen. Die dynamische virtuelle Föderation kann basierend auf der Klassifikation von Föderationen aus Kapitel 2 wie folgt charakterisiert werden (vgl. Abbildung 4.2):

- Die Kooperationsstruktur kann sowohl die einer Ad hoc Föderation, einer Hub-and-spoke Föderation als auch einem Identity network entsprechen.
- Die Anzahl der Teilnehmer ist flexibel, d. h. sie kann die Eigenschaft bilateral, fest oder komplex annehmen.
- Die Gruppenstruktur hängt von den Anforderungen und Bedürfnissen der Teilnehmer ab und kann sowohl allgemein offen, als auch offen mit Einschränkungen oder abgeschlossen sein. Abgeschlossene Föderationen treten jedoch eher selten auf.
- Die Dimension der Föderation ist ebenfalls offen. Räumlich kann sie die Werte lokal, regional, national oder international annehmen.
- Bei der organisatorischen Dimension ist die Intra-Föderation möglich. Zusätzlich können auch dynamische virtuelle Inter-Föderationen erstellt werden.
- Die Dauer der Föderation hängt von den Anforderungen ab und kann projektbezogen oder zeitlich befristet sein. Durch die Eigenschaft der Dynamik werden unbefristete Föderationen eher selten auftreten.
- Die Zusammenarbeit kann alle Ausprägungen annehmen. Sowohl Projekte, VOs, aber auch andere Gründe können für eine Kollaboration ausschlaggebend sein.
- Die Koordination kann implizit erfolgen. Genauso sind explizit und Mischformen denkbar.
- Der Gründungsprozess ist tendenziell spontan ereignisgesteuert bzw. bei Bedarf, da der Vertrauensaufbau dynamisch geschieht. Jedoch sind geplante Gründungsprozesse mit einer abgeschlossenen oder mit Einschränkungen offenen Gruppenstruktur denkbar.
- Der Circle of Trust kann ebenfalls alle Ausprägungen außer statisch annehmen.
- Die Bindungsintensität wird tendenziell Absprachen am häufigsten entsprechen, da die Dynamik bei Verträgen und Kapitalbeteiligung gering ist. Jedoch können theoretisch auch diese Ausprägungen angenommen werden.
- Das Vertrauen wird meist direkt sein, aber auch indirekt ist denkbar.

4.2.2. Dynamische virtuelle Inter-Föderationen

Definition 7. Dynamische virtuelle Inter-Föderation

Dynamische virtuelle Inter-Föderation ist der Zusammenschluss von Föderationen, der dynamisch gebildet wird und parallel zu festen Inter-Föderationen bestehen kann.

Basierend darauf können auch dynamische virtuelle Inter-Föderationen aus zwei oder mehreren Föderationen gebildet werden. Hierbei können auch Entitäten außerhalb von Föderationen teilnehmen. Die Bildung kann ebenfalls dynamisch und virtuell geschehen, wofür entsprechende Schnittstellen und Workflows benötigt werden. Eine Voraussetzung hierfür ist die Zulassung der dynamischen Bildung einer Inter-Föderation durch die beteiligten Föderationen.

Dynamische virtuelle Föderationen entstehen, wenn eine Mindestanzahl an bilateralen Föderationen besteht, die großteils miteinander verbunden sind. Dasselbe Prinzip besteht für dynamische virtuelle Inter-Föderationen. Wird diese Mindestanzahl für eine bestimmte Dauer unterschritten, werden die dynamischen virtuellen Föderationen und Inter-Föderationen wieder aufgelöst. Genauso ist ein Übergang zu einer Art feste Föderation bzw. Inter-Föderation möglich. Die Unterscheidung zwischen dynamischen virtuellen Föderationen und Inter-Föderationen geschieht insbesondere, um das Szenario Föderation innerhalb einer Föderation darzustellen und um ein Konzept zu haben, beide in feste Föderationen überführen zu können, falls gewünscht.

4.2.3. Föderationsverwaltung

Zusätzlich zu den dynamischen virtuellen Föderationen können in der zu erarbeitenden Architektur weitere Ausprägungsformen von Föderationen teilnehmen, um parallele Konzepte zu vermeiden. Dies wird aus Abbildung 4.3 ersichtlich. Zusätzlich zu den Eigenschaften von dynamischen virtuellen Föderationen soll Folgendes möglich sein:

- Die organisatorische Dimension besteht nicht nur aus Intra-Föderationen. Auch Inter-Föderationen und organisatorische Föderationen können die zu konzipierende Architektur einsetzen.
- Die Dauer der Föderation muss nicht begrenzt sein.
- Die Zusammenarbeit kann somit auch auf Grund von FIM geschehen.
- Die Koordination kann durch die Schnittstelle für Föderationsverwaltungen auch explizit geschehen.

- Folglich sind auch geplante Föderationen machbar.
- Der Circle of Trust kann damit das Merkmal statisch annehmen.
- Einfache Verträge sollen abschließbar sein. Für Kapitalbeteiligung müssen für SLAs Wege außerhalb der zu konzipierenden technischen Lösung genommen werden.

Dazu wird eine einheitliche Schnittstelle für die Föderationsverwaltung, ähnlich wie das Management Tool in den Dienstmodellen der Szenarien, benötigt. Über diese Schnittstelle können die Verantwortlichen ihre Föderationen administrieren. Das bedeutet, dass sie u. a.

- Föderation anlegen,
- Aufnahmeprozess festlegen,
- potentielle Mitglieder aufnehmen oder ablehnen,
- Policies erstellen,
- Policies ändern,
- Mitglieder ausschließen,
- ein Mindestmaß an Vertrauen festlegen und
- Nutzern Berechtigungen geben bzw. diese ändern können.

Über die Schnittstelle der Föderationsverwaltung können ebenso Inter-Föderationen verwaltet werden. Für den Aufnahmeprozess müssen sich die interessierten Föderationen registrieren, bevor sie akzeptiert oder abgelehnt werden. Die Organisationen teilen den Teilnahmewunsch ihren Föderationen über die Föderationsverwaltung mit.

In den folgenden Abschnitten werden verschiedene Modelle definiert, um dynamische virtuelle Föderationen und die Föderationsverwaltung zu konzipieren. Die Realisierung der dynamischen virtuellen Föderation durch Werkzeuge wird in Kapitel 5 genauer spezifiziert.

4.3. Organisationsmodell

In den beiden Abschnitten FIM und Inter-FIM des Kapitels 2 wurden die beteiligten Akteure bereits genannt und kurz beschrieben. Als Entitäten sind Service Provider und Identity Provider vertreten. Zusätzlich können Attribute Authorities angefragt werden. Desweiteren soll es eine Möglichkeit für Föderationen und Inter-Föderationen geben, ihre Teilnehmer zu verwalten und Policies zu erstellen. Daher stellen sich im Organisationsmodell die folgenden Fragen:

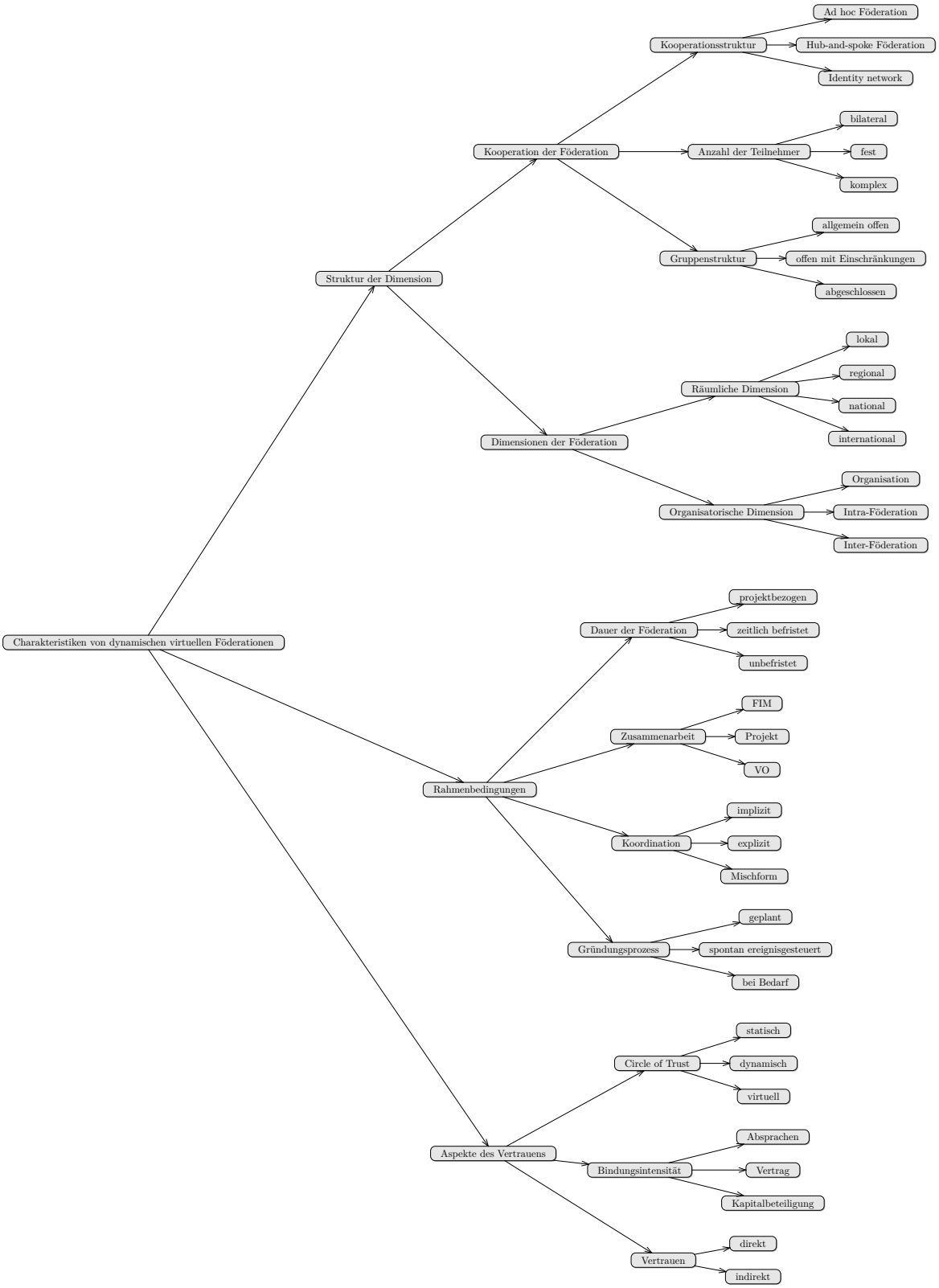

Abbildung 4.3.: Klassifikation von Föderationen, die die zu erarbeitende Architektur verwenden können

- Welche Managementdomänen sind notwendig für die FIM-/Inter-FIM-Managementarchitektur?
- Welche Rollen sind in den Managementdomänen vorhanden?
- Wie interagieren die Rollen der Managementdomänen untereinander?
- Wie kann die Software interagieren und welche Rollen bestehen hier?
- Wie setzt sich hierdurch das Organisationsmodell zusammen?

In diesem Abschnitt werden die Rollen, die Akteure und deren Interaktionen genauer definiert und somit die oben genannten Fragen beantwortet.

4.3.1. Managementdomänen

Aus der Beschreibung der Szenarien und der oben gemachten Definition von dynamischen virtuellen Föderationen und Inter-Föderationen wird deutlich, dass eine Einteilung in Managementdomänen notwendig ist. Die Einteilung in die unterschiedlichen Managementdomänen hilft die Interaktionen und Kommunikationen in diesem Kapitel besser zu definieren. Insgesamt werden die folgenden Managementdomänen betrachtet:

SPDomain: Die Service Provider Managementdomäne SPDomain repräsentiert die lokale Domäne des Service Provider, der den Dienst erbringt.

IdPDomain: Die Identity Provider Managementdomäne IdPDomain repräsentiert die lokale Domäne des Identity Provider, der die Benutzerinformationen verwaltet und zugleich die Nutzer-Domäne darstellt.

AADomain: Die Attribute Authority Managementdomäne AADomain repräsentiert die lokale Domäne der Attribute Authority, die zusätzliche Benutzerinformationen verwaltet.

fedDomain: Die Federation Managementdomäne fedDomain repräsentiert die Domäne der Föderation, die unterschiedliche Strukturen annehmen kann.

interfedDomain: Die Inter-Föderation Managementdomäne interfedDomain repräsentiert die Domäne der Inter-Föderation, die ebenfalls unterschiedliche Strukturen annehmen kann.

4.3.2. Definition der Rollen

Basierend auf den oben genannten Managementdomänen lassen sich verschiedene Rollen identifizieren. Die beteiligten Akteure wurden bereits im Abschnitt 2.2 definiert. In diesem Abschnitt werden die Akteure auf Rollen aufgeteilt und in die entsprechenden Managementdomänen eingeteilt. Zwischen den Rollen finden die Kommunikation und damit der Austausch der Informationen statt.

Rollen in der SPDomain

Die für das technische FIM/Inter-FIM wichtige Rolle in der SPDomain ist der *SP Administrator (SP-A)*. Zusätzlich wird die Rolle des *SP Relationship Manager (SP-RM)* eingeführt. Intern können die in Patricia Marcus Dissertation [Mar11] beschriebenen Rollen zum interorganisationalen Fehlermanagement eingesetzt werden.

Der *SP Administrator* ist diejenige Rolle, die für die Administration des SPs verantwortlich ist. Sie setzt die SP Software auf, aktualisiert die Software und die Konfiguration. Die Rolle koordiniert interne Rollen, die beispielsweise für das Fehlermanagement und die Verbesserung des Dienstes zuständig sind. Bei kommerziellen Diensten ist eine Schnittstelle zum Accounting Management verfügbar. Die Rolle ist in der Tabelle 4.1 zusammengefasst.

Rolle	*SP Administrator*
Bezeichner	SP-A
Mgmt. Domäne	SPDomain
Beschreibung	Administration des Service Providers
Assoziierte Akteure	SP

Tabelle 4.1.: Zusammenfassung der Rolle Service Provider Administrator

Der *SP-RM* wird durch den Akteur SP Manager (SPM) durchgeführt. Die Rolle ist zuständig für das Verhältnis zu anderen Entitäten und Föderationen. Wenn beispielsweise der SP Mitglied einer Föderation werden will, ist diese Rolle zuständig für die Kommunikation mit der Föderation. Dies kann beispielsweise die Akzeptanz von Policies beinhalten. Die Rolle ist in der Tabelle 4.2 dargestellt.

Rolle	*SP Relationship Manager*
Bezeichner	SP-RM
Mgmt. Domäne	SPDomain
Beschreibung	Relationship Management des Service Providers
Assoziierte Akteure	SPM

Tabelle 4.2.: Zusammenfassung der Rolle Service Provider Relationship Manager

Die Rolle SP Service Desk (SP-SD), vgl. Tabelle 4.3, nimmt Störungen auf und kommu-

niziert mit dem Federation Service Desk (Fed-SD) über administrative Änderungen.

Rolle	*SP Service Desk*
Bezeichner	SP-SD
Mgmt. Domäne	SPDomain
Beschreibung	Aufnehmen und Weiterleiten von Störungen innerhalb des SPs
Assoziierte Akteure	SP-SD

Tabelle 4.3.: Zusammenfassung der Rolle Service Provider Service Desk

Rollen in der IdPDomain

Das Konzept des Customers ist aus dem MNM-Dienstmodell von Garschhammer et al. (vgl. [GHH+01] [HAN99]) abgeleitet. Es differenziert auf Seite des Kunden zwischen der Rolle des tatsächlichen Nutzers und der Rolle des Customers. Der Nutzer ist derjenige, der den Dienst tatsächlich nutzt und somit auch mögliche Störungen im Betrieb des Dienstes bemerken kann. Während der Customer im MNM-Dienstmodell die Managementaufgaben seitens des Kunden übernimmt, wird hier der Customer aufgeteilt. Zum einen existiert ein *IdP Administrator (IdP-A)*, der die IdP-Software administriert. Zum anderen kümmert sich ein *IdP Relationship Manager (IdP-RM)* um die Außenbeziehungen, beispielsweise mit SPs und Föderationen. Dies ist u. a. bei Verträgen notwendig. Bei Störungen ist die Rolle IdP Service Desk (IdP-SD) zuständig.
Der Kunde (*User*) verwendet, wie gerade beschrieben, Dienste und lässt sich folgendermaßen (vgl. Tabelle 4.4) zusammenfassen.

Rolle	*User*
Bezeichner	User
Mgmt. Domäne	IdPDomain
Beschreibung	Nutzt Dienste
Assoziierte Akteure	User

Tabelle 4.4.: Zusammenfassung der Rolle User

Der *IdP Administrator* kümmert sich um den Betrieb der IdP-Software, wie beispielsweise Shibboleth. Er ist dafür zuständig die Software zu installieren und anschließend zu warten. Die Rolle ist zudem die Kontaktperson zu anderen Managementprozessen, wie Change und Accounting Management. Häufig ist die Rolle zudem für das Konfigurationsmanagement und das Change Management zuständig. Die Beschreibung der Rolle findet sich in Tabelle 4.5.

Äquivalent zum *SP Relationship Manager* ist die Rolle *IdP-RM* für die Außenbeziehungen des IdPs zuständig, siehe Tabelle 4.6.

Die Rolle *IdP-SD*, vgl. Tabelle 4.7, nimmt Störungen auf und kommuniziert mit dem Fed-SD über administrative Änderungen.

Rolle	*IdP Administrator*
Bezeichner	IdP-A
Mgmt. Domäne	IdPDomain
Beschreibung	Administration des Identity Providers
Assoziierte Akteure	IdP

Tabelle 4.5.: Zusammenfassung der Rolle Identity Provider Administrator

Rolle	*IdP Relationship Manager*
Bezeichner	IdP-RM
Mgmt. Domäne	IdPDomain
Beschreibung	Relationship Management des Identity Providers
Assoziierte Akteure	IdP Manager (IdPM)

Tabelle 4.6.: Zusammenfassung der Rolle Identity Provider Relationship Manager

Rolle	*IdP Service Desk*
Bezeichner	IdP-SD
Mgmt. Domäne	IdPDomain
Beschreibung	Aufnehmen und Weiterleiten von Störungen innerhalb des IdPs
Assoziierte Akteure	IdP-SD

Tabelle 4.7.: Zusammenfassung der Rolle Identity Provider Service Desk

Rollen in der AADomain

Ebenso wie bei SPDomain und IdPDomain benötigt die AADomain einen Administrator, *AA Administrator (AA-A)* genannt. Der *AA Administrator* betreibt die Attribute Authority, die softwareunterstützt Nutzerinformationen an IdP bzw. SP übergibt. Die Software basiert häufig auf den typischen SAML-Implementierungen für Identity Provider, wie Shibboleth IdP und SimpleSAMLphp. Die Rolle *AA Administrator* ist in der Tabelle 4.8 zusammenfassend beschrieben.

Rolle	*AA Administrator*
Bezeichner	AA-A
Mgmt. Domäne	AADomain
Beschreibung	Administration der Attribute Authority
Assoziierte Akteure	AA

Tabelle 4.8.: Zusammenfassung der Rolle Attribute Authority Administrator

Äquivalent zum SP Relationship Manager und IdP Relationship Manager ist die Rolle *AA Relationship Manager (AA-RM)* für die Außenbeziehungen des AAs zuständig, siehe Tabelle 4.9.

Rolle	*AA Relationship Manager*
Bezeichner	AA-RM
Mgmt. Domäne	AADomain
Beschreibung	Relationship Management der Attribute Authority
Assoziierte Akteure	AA

Tabelle 4.9.: Zusammenfassung der Rolle Attribute Authority Relationship Manager

Die Rolle *AA Service Desk (AA-SD)*, vgl. Tabelle 4.10, nimmt Störungen auf und kommuniziert mit dem Fed-SD und weiteren Service Desk (SD) über administrative Änderungen.

Rolle	*AA Service Desk*
Bezeichner	AA-SD
Mgmt. Domäne	AADomain
Beschreibung	Aufnehmen und Weiterleiten von Störungen innerhalb des AAs
Assoziierte Akteure	AA-SD

Tabelle 4.10.: Zusammenfassung der Rolle Attribute Authority Service Desk

Rollen in der fedDomain

Wie bereits in Abschnitt 2.2 beschrieben, können Föderationen verschiedene Strukturen annehmen. Dementsprechend unterschiedlich können die Rollen ausfallen. Dadurch wird im Folgenden zwischen den drei Strukturen unterschieden.

Bei einer Ad hoc federation werden keine zusätzlichen Rollen innerhalb der Föderation benötigt. Bei Identity Provider und Service Provider kümmern sich die Rollen *Relationship Manager* um die Kooperation, während die Rollen *Administratoren* die technische Kooperation betreuen.

Eine Hub-and-spoke federation betreibt meist einen IdP-Proxy, der IdPs und SPs miteinander verbindet. Dieser Proxy wird auch für Kooperationen außerhalb der Föderation eingesetzt. Neben dem *Administrator* werden die Rollen *Change Manager* und *Configuration Manager* eingeführt, da eine Veränderung und die Konfiguration des IdP-Proxys direkte Auswirkungen auf die Föderation haben. Ein *Federation Service Desk* kümmert sich um die Fragen und Probleme der Mitglieder, während die Rolle *Federation Technical Support* Störungen beseitigt. Neben dem *Relationship Manager*, von SP und IdP bekannt, wird ein *General Manager* eingeführt, der die Föderation leitet. Die Rolle *Initiator* hat ursprünglich die Föderation initiiert. Diese Rollen sind in den nachfolgenden Tabellen genauer dargestellt.

Der *Relationship Manager* (vgl. Tabelle 4.11) betreut die Beziehung nach außen. Für die fedDomain ist diese Rolle zusätzlich wichtig für das Management der Mitglieder der Föderation. Das bedeutet u. a. Policies zu erstellen, zu ändern, schauen, dass alle Mitglieder den Anforderungen entsprechen und gegebenenfalls Mitglieder von der Föderation ausschließen.

Rolle	*Federation Relationship Manager*
Bezeichner	Federation Relationship Manager (Fed-RM)
Mgmt. Domäne	fedDomain
Beschreibung	Relationship Management der Föderation
Assoziierte Akteure	Federation Manager (FM)

Tabelle 4.11.: Zusammenfassung der Rolle Federation Relationship Manager

Die Rolle des *General Manager* (vgl. Tabelle 4.12) ist dafür zuständig die Föderation als solche zu verwalten und nach außen hin zu repräsentieren. Er erhält von *Fed-RM*, *Federation Administrator (Fed-A)*, *Federation Technical Support (Fed-TS)*, *Federation Change Manager (Fed-CM)* und *Federation Configuration Manager (Fed-ConM)* regelmäßig Informationen und leitet diese Rollen an.

Eine besondere Rolle spielt der *Initiator* der Föderation (vgl. Tabelle 4.13). Der *Initiator* hat mitsamt anderer Organisationen die Föderation ins Leben gerufen. Diese Rolle ist meist durch einen Mitarbeiter innerhalb der federführenden Organisation besetzt, der mit anderen kooperierenden Organisationen verhandelt hat.

Rolle	*General Manager*
Bezeichner	Federation General Manager (Fed-GM)
Mgmt. Domäne	fedDomain
Beschreibung	Management der Föderation
Assoziierte Akteure	FM

Tabelle 4.12.: Zusammenfassung der Rolle Federation Relationship Manager

Rolle	*Initiator*
Bezeichner	Initiator
Mgmt. Domäne	fedDomain
Beschreibung	Initiator der Föderation
Assoziierte Akteure	FM

Tabelle 4.13.: Zusammenfassung der Rolle Initiator

Äquivalent zu IdP, SP und AA existiert ein *Administrator* der Föderation, der die Software verwaltet. Diese Rolle ist in der Tabelle 4.14 dargestellt.

Rolle	*Federation Administrator*
Bezeichner	Fed-A
Mgmt. Domäne	fedDomain
Beschreibung	Administration der Föderation
Assoziierte Akteure	Federation Administrator (FA)

Tabelle 4.14.: Zusammenfassung der Rolle Federation Administrator

Die Rolle *Federation Technical Support* (Fed-SD, vgl. Tabelle 4.15) nimmt Störungen innerhalb der Software auf, die für die Funktion der Föderation wichtig sind. Falls *Fed-SD* den Incident selbst nicht lösen kann, wird dieser an den *Federation Technical Support* weiter gegeben. In kleinen leitenden Organisationen kann neben *Federation Service Desk (FSD)* auch *FA* diese Rolle übernehmen.

Rolle	*Federation Service Desk*
Bezeichner	Fed-SD
Mgmt. Domäne	fedDomain
Beschreibung	Aufnehmen und Weiterleiten von Störungen innerhalb der Föderation
Assoziierte Akteure	FSD

Tabelle 4.15.: Zusammenfassung der Rolle Federation Service Desk

Zusätzlich zum Administrator wird die Rolle des *Federation Technical Support* etabliert. Dieser befasst sich, wie in Tabelle 4.16 dargestellt, mit der Lösung von Störungen innerhalb der Föderation bzw. der vom *Fed-A* administrierten Software. Die Störungen werden von der

Rolle *Federation Technical Support* an den *Fed-SD* übergeben. In den meisten Fällen wird der Akteur *FA* diese Rolle ausfüllen.

Rolle	*Federation Technical Support*
Bezeichner	Fed-TS
Mgmt. Domäne	fedDomain
Beschreibung	Lösen von Störungen innerhalb der Föderation
Assoziierte Akteure	FA

Tabelle 4.16.: Zusammenfassung der Rolle Federation Technical Specialist

Die Rolle *Change Manager* (vgl. Tabelle 4.17) verwaltet Changes, die für die Föderation relevant sind. Da Hub-and-spoke Föderationen meist einen IdP-Proxy betreiben, beinhaltet dies u. a. das Aktualisieren der Software. Dabei müssen die Auswirkungen auf die Föderation bedacht werden. Die Changes werden durch die Rolle *Federation Administrator* durchgeführt.

Rolle	*Federation Change Manager*
Bezeichner	Fed-CM
Mgmt. Domäne	fedDomain
Beschreibung	Verwaltung von Changes, die für die Föderation relevant sind
Assoziierte Akteure	FA

Tabelle 4.17.: Zusammenfassung der Rolle Federation Change Manager

Der *Configuration Manager* (vgl. Tabelle 4.18) ist für die Konfiguration der Föderation bzw. der zugehörigen Software zuständig. Zur Konfiguration gehören das Erstellen von Konvertierungsregeln und Attributfiltern sowie Änderungen an Konfigurationsdateien.

Rolle	*Federation Configuration Manager*
Bezeichner	Fed-ConM
Mgmt. Domäne	fedDomain
Beschreibung	Konfiguration der Föderation bzw. der zugehörigen Software
Assoziierte Akteure	FA

Tabelle 4.18.: Zusammenfassung der Rolle Federation Configuration Manager

Ein Identity network betreibt anstelle eines IdP-Proxys Federation Management Tools. Diese erfordern im Gegensatz zum Betrieb eines IdP-Proxies meist weniger Personal, wodurch die Rollen *Administrator*, *Service Desk*, *Technical Specialist* und *Relationship Manager* durch eine kleine Gruppe bzw. sogar eine Person besetzt sind. Im Gegensatz zu Hub-and-spoke federation fällt die Rolle *Federation Configuration Manager* weg, solange keine weitere Konfiguration benötigt wird. Die Rolle *Federation Change Manager* ist im Identity network weniger wichtig.

Rollen in der interfedDomain

Ebenso wie Föderationen, können Inter-Föderationen verschiedene Strukturen annehmen. Entsprechend der Beschreibung der Rollen in einer fedDomain werden bei einer interfedDomain ebenso keine zusätzlichen Rollen benötigt, wenn die Ausprägung eine Ad hoc federation darstellt. Äquivalent zur Hub-and-spoke federation existieren in einer Hub-and-spoke interfederation die nachfolgenden Rollen. Der *Inter-Federation Relationship Manager* (vgl. Tabelle 4.19) betreut die Beziehung nach außen und insbesondere innerhalb der Inter-Föderation. Die Rolle ist mit dem Akteur Inter-Federation Manager assoziiert.

Rolle	*Inter-Federation Relationship Manager*
Bezeichner	Inter-Federation Relationship Manager (IFed-RM)
Mgmt. Domäne	interfedDomain
Beschreibung	Relationship Management der Inter-Föderation
Assoziierte Akteure	Inter-Federation Manager (IFM)

Tabelle 4.19.: Zusammenfassung der Rolle Inter-Federation Relationship Manager

Die Rolle des *General Managers* (vgl. Tabelle 4.20) ist dafür zuständig die Inter-Föderation als solche zu verwalten und nach außen hin zu repräsentieren. Die Rollen

- *IFed-RM*,
- *Inter-Federation Administrator (IFed-A)*,
- *Inter-Federation Technical Support (IFed-TS)*,
- *Inter-Federation Change Manager (IFed-CM)* und
- *Inter-Federation Configuration Manager (IFed-ConM)*

arbeiten ihm zu.

Rolle	*General Manager*
Bezeichner	Inter-Federation General Manager (IFed-GM)
Mgmt. Domäne	interfedDomain
Beschreibung	Management der Inter-Föderation
Assoziierte Akteure	IFM

Tabelle 4.20.: Zusammenfassung der Rolle General Manager

Der *Initiator* der Inter-Föderation (vgl. Tabelle 4.21) gehört ursprünglich der Domäne *fedDomain* zu, die sich als führende Organisation der Inter-Föderation herausgebildet hat.

Rolle	*Initiator*
Bezeichner	Initiator
Mgmt. Domäne	interfedDomain
Beschreibung	Initiator der Inter-Föderation
Assoziierte Akteure	IFM

Tabelle 4.21.: Zusammenfassung der Rolle Initiator

Die Rolle des *Administrators* der Inter-Föderation ist in der Tabelle 4.22 dargestellt. Sie ist mit dem Akteur Inter-Federation Administrator (IFA) assoziiert.

Rolle	*Inter-Federation Administrator*
Bezeichner	IFed-A
Mgmt. Domäne	interfedDomain
Beschreibung	Administration der Inter-Föderation
Assoziierte Akteure	IFA

Tabelle 4.22.: Zusammenfassung der Rolle Inter-Federation Administrator

Die Rolle *Inter-Federation Service Desk* (vgl. Tabelle 4.23) nimmt Störungen innerhalb der Software auf, die für die Funktion der Inter-Föderation wichtig sind. Falls *Inter-Federation Service Desk (IFed-SD)* den Incident selbst nicht lösen kann, wird dieser an den *Inter-Federation Service Desk* weiter gegeben. Die Rolle ist mit dem Akteur Inter-Federation Service Desk (IFSD) assoziiert.

Rolle	*Inter-Federation Service Desk*
Bezeichner	IFed-SD
Mgmt. Domäne	interfedDomain
Beschreibung	Aufnehmen und Weiterleiten von Störungen innerhalb der Inter-Föderation
Assoziierte Akteure	IFSD

Tabelle 4.23.: Zusammenfassung der Rolle Inter-Federation Service Desk

Der *Inter-Federation Technical Support* befasst sich, wie in Tabelle 4.24 dargestellt, mit der Lösung von Störungen innerhalb der Inter-Föderation. Die Störungen werden von der Rolle *Inter-Federation Service Desk* an den *IFed-TS* übergeben.

Die Rolle *Inter-Federation Change Manager* (vgl. Tabelle 4.25) verwaltet Changes, die für die Inter-Föderation relevant sind. Die Changes werden durch den Akteur *Inter-Federation Administrator* durchgeführt.

Der *Inter-Federation Configuration Manager* (vgl. Tabelle 4.26) ist für die Konfiguration der zugehörigen Software zuständig.

Ein Identity network betreibt Inter-Federation Management Tools. Im Gegensatz zu Hub-

Rolle	*Inter-Federation Technical Specialist*
Bezeichner	IFed-TS
Mgmt. Domäne	interfedDomain
Beschreibung	Lösen von Störungen innerhalb der Inter-Föderation
Assoziierte Akteure	IFA

Tabelle 4.24.: Zusammenfassung der Rolle Inter-Federation Technical Specialist

Rolle	*Inter-Federation Change Manager*
Bezeichner	IFed-CM
Mgmt. Domäne	interfedDomain
Beschreibung	Verwaltung von Changes, die für die Inter-Föderation relevant sind
Assoziierte Akteure	IFA

Tabelle 4.25.: Zusammenfassung der Rolle Inter-Federation Change Manager

Rolle	*Inter-Federation Configuration Manager*
Bezeichner	IFed-ConM
Mgmt. Domäne	interfedDomain
Beschreibung	Konfiguration der Inter-Föderation bzw. der zugehörigen Software
Assoziierte Akteure	IFA

Tabelle 4.26.: Zusammenfassung der Rolle Inter-Federation Configuration Manager

and-spoke inter-federation fällt die Rolle *Inter-Federation Configuration Manager* weg.

Interaktionen zwischen Rollen

Basierend auf den bereits beschriebenen Rollen und Domänen werden in diesem Abschnitt die *Interaktionen* zwischen den Rollen betrachtet. Die Interaktion zwischen den Rollen erfolgt für die Identifikation, Initiierung, den Betrieb, die Anpassung und Auflösung von Föderationen und Inter-Föderationen. Die *Interaktionskanäle* werden in Rahmen verschiedener Funktionalitäten angeboten; hier wird jedoch nur eine Übersicht gewährt sowie in Ausschnitten kurz definiert.

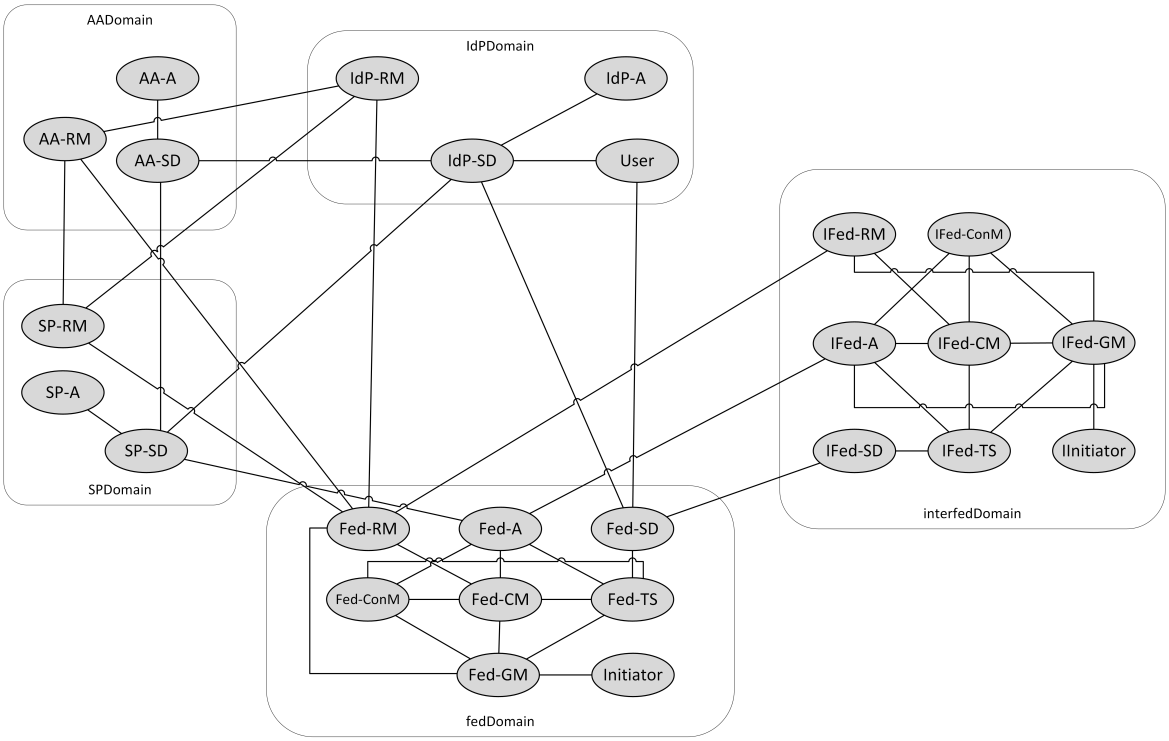

Abbildung 4.4.: Interaktionen in Inter-Federated Identity Management

Die Übersicht der Interaktionen in Abbildung 4.4 zeigt viele Kommunikationskanäle zwischen AADomain, SPDomain, fedDomain, interfedDomain und IdPDomain.
Die Rollen Change Manager (CM), Technical Support (TS), General Manager (GM) und Configuration Manager (ConM) sind ebenfalls in der SPDomain, IdPDomain und AADomain zu finden, jedoch zur Übersichtlichkeit herausgelassen worden. Ebenso können Interaktionen zwischen SPDomain bzw. IdPDomain und interfedDomain stattfinden, die jedoch vom Prinzip äquivalent zu den Interaktionen mit der fedDomain sind und dadurch zur Vereinfachung weggelassen wurden. Basierend auf der Abbildung werden im Folgenden bestimmte Abschnitte, wie die Interaktion zwischen IdPDomain, SPDomain und fedDomain, näher betrachtet.

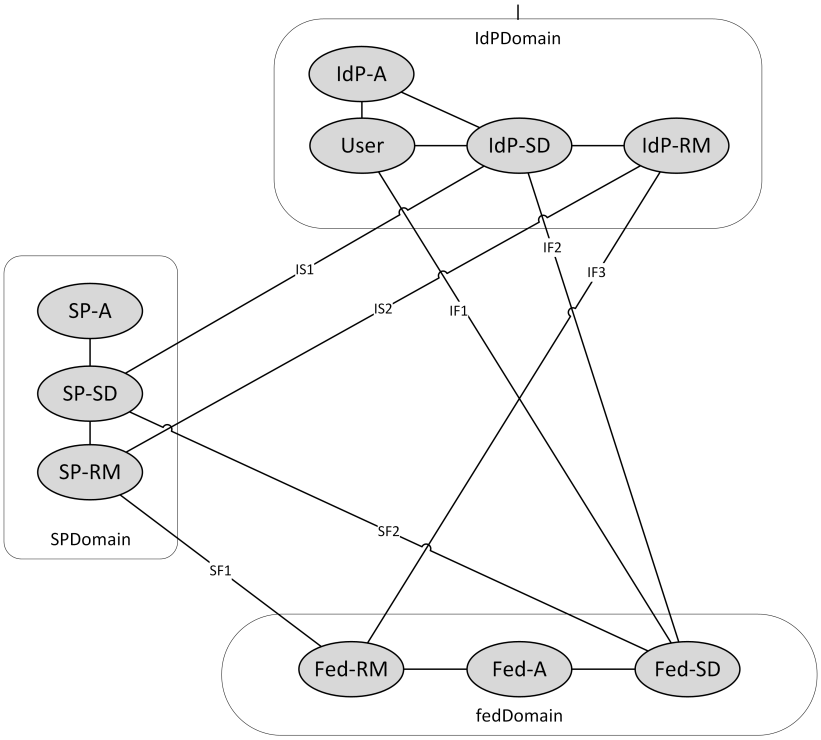

Abbildung 4.5.: Interaktionen in Inter-Federated Identity Management zwischen IdP und SP

Die Interaktionen zwischen IdPDomain, SPDomain und fedDomain wird in der Abbildung 4.5 aufgezeigt. Die Kommunikation verläuft insbesondere über die folgenden sieben Kommunikationskanäle:

IS1: IS1 wird von den Rollen IdP-A und SP-A genutzt, um Informationen bezüglich der Administration über die Rollen IdP-SD und SP-SD auszutauschen. Dies beinhaltet beispielsweise die Anpassung von Konfiguration und, über technische Kanäle, der Austausch von Metadaten.

IS2: Die Rollen IdP-RM und SP-RM kommunizieren für mögliche Verträge und Abstimmungen.

IF1: Die Rolle User verwendet die Rolle SD der Föderation, falls Probleme auftreten. Dies kann der Fall sein, wenn die Metadaten noch nicht ausgetauscht wurden oder die weitere Konfiguration nicht fehlerfrei ist. Basierend auf einen zentralen Ansatz einer TTP kann der Nutzer den Service Desk informieren, wenn der Metadatenaustausch fehlerhaft war oder es keine passenden Konvertierungsregeln gibt. Über den Service Desk der Föderation wird dieser Incident an die passenden Stellen weiter geleitet.

IF2: Die Kommunikation zwischen IdP-A und Fed-A erfolgt über IdP-SD und Fed-SD zu technischen Themen der Föderation, wie Policies, technische Aspekte der Teilnahme

an Inter-Föderationen oder allgemeine Änderungen.

IF3: IF3 ist ein aktiver Kanal, der für Verwaltungsthemen der Föderation genutzt wird. Ein Beispiel hierfür ist die Absprache von Änderungen und die Teilnahme an Inter-Föderationen.

SF1: Die Kommunikation zwischen SP-A und Fed-A über SP-SD und Fed-SD ist äquivalent zum Kanal IF2.

SF2: SF2 ist ebenso wie IF3 ein aktiver Kanal, insbesondere bei Änderungen.

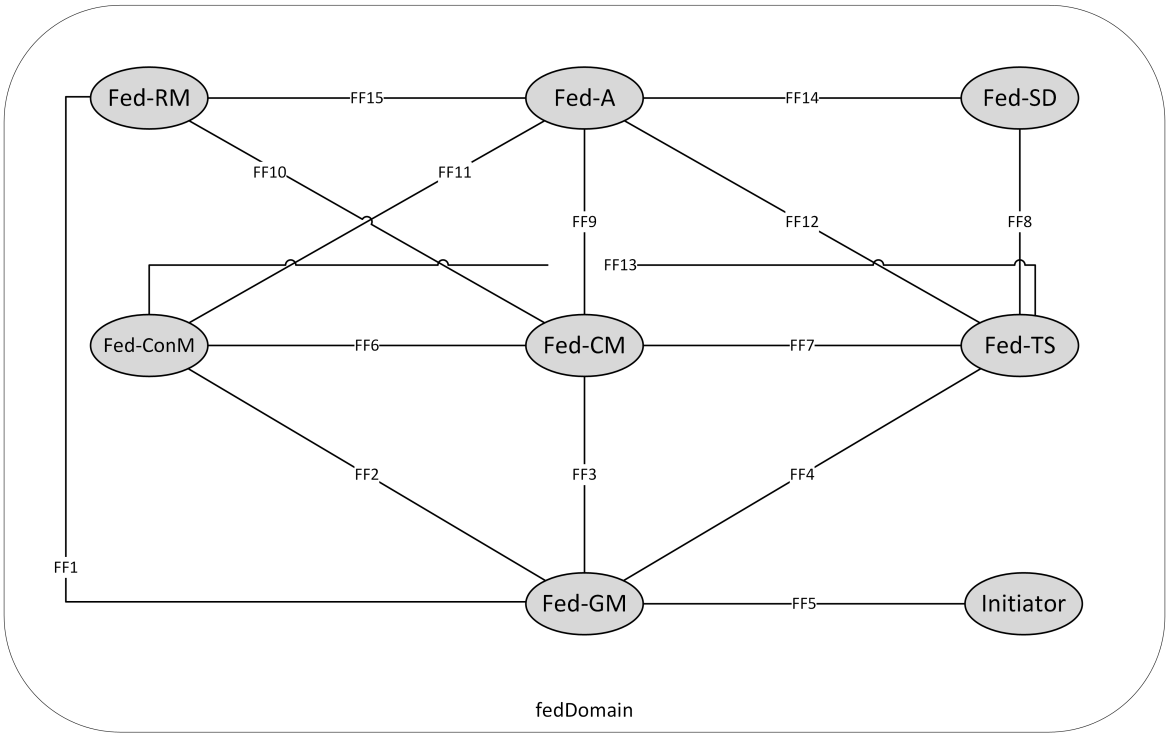

Abbildung 4.6.: Interaktionen in Inter-Federated Identity Management innerhalb der fedDomain

Nun soll die Interaktion der Rollen innerhalb einer Föderation genauer aufgezeigt werden. Die fedDomain besteht aus acht Rollen, die vernetzt sind. Abbildung 4.6 zeigt die verschiedenen Kanäle auf, die wie folgt kommunizieren. Die Kanäle FF1 bis FF4 dienen der Rolle Fed-GM zum Verwalten der fedDomain.

FF1: Fed-GM und Fed-RM kommunizieren zum Verwalten der Domain.

FF2: Fed-GM und Fed-ConM interagieren zum Verwalten der Domain.

FF3: Fed-GM und Fed-CM kooperieren zum Verwalten der Domain.

FF4: Fed-GM und Fed-TS interagieren zum Verwalten der Domain.

FF5: Die Rolle Initiator übergibt der Rolle Fed-GM die Verwaltung über die fedDomain und somit über die Föderation. Diese Kommunikation findet nur bei der Initiierung der Föderation statt.

FF6: Fed-ConM und Fed-CM interagieren, wenn eine Änderung, d. h. ein Change, der Konfiguration nötig ist.

FF7: Fed-TS und Fed-CM kommunizieren, wenn durch einen Incident oder aus anderen betriebstechnischen Gründen ein Change durchgeführt werden muss.

FF8: Fed-TS wird durch Fed-SD mit dem Lösen von Incidents beauftragt, die durch User aufgegeben wurden.

FF9: Fed-CM und Fed-A interagieren, wenn bei der Administration Changes notwendig werden, beispielsweise durch Software-Upgrades oder Patches.

FF10: Fed-RM und Fed-CM kommunizieren, wenn durch Mitglieder oder andere Föderationen bzw. Inter-Föderationen Changes anstehen. Dies kann ebenso bei der Teilnahme einer weiteren Entität sein wie bei der Initiierung einer Inter-Föderation.

FF11: Fed-A und Fed-ConM interagieren, wenn eine Änderung der Konfiguration benötigt wird.

FF12: Fed-TS und Fed-A sprechen, wenn Fed-A Probleme bei der Administration bemerkt.

FF13: Fed-TS und Fed-ConM kommunizieren indirekt, wenn für Lösungen von Incidents Änderungen an der Konfiguration benötigt werden. Die Kommunikation erfolgt über Fed-CM.

FF14: Fed-SD und Fed-A interagieren, wenn Incidents durch die Rolle Fed-A gelöst werden können oder wenn administrative Änderungen kommuniziert werden müssen.

FF15: Fed-RM und Fed-A kooperieren, wenn Änderungen der Beziehungen zu Teilnehmern oder Inter-Föderationen für den Administrator auftreten.

4.3.3. Spezifikation des Organisationsmodells

In den vorangegangenen Abschnitten wurden Rollen und Interaktionskanäle genauer beschrieben. Für ein Organisationsmodell müssen die Entitäten und das Domänenkonzept des FIM-/Inter-FIM-Organisationsmodells in Unified Modeling Language (UML) spezifiziert werden. UML eignet sich insbesondere, um die Bedeutung von Domänen, Rollen und die sie verbindenden Interaktionskanäle in der Modellierung hervorzuheben. Eine Vertiefung

der Interaktionen und Rollen wird im Abschnitt 4.4 vorgenommen.

Abbildung 4.7 beschreibt das Organisationsmodell für Federated Identity Management. Die Abbildung zeigt die Domänen *AADomain*, *IdPDomain*, *SPDomain* und *fedDomain*, die jeweils als Stereotyp der Klasse *FIMDomain* modelliert sind. Jeder Domäne sind statisch die entsprechenden Rollen als Klasse vom Stereotyp *FIMRole* zugeordnet. Wenn zwei Rollen in derselben oder in unterschiedlichen Managementdomänen interagieren, geschieht dies über eine Klasse des Stereotyps *InteractionChannel*. Der besseren Übersichtlichkeit halber sind in der Abbildung die Klassen der Interaktionskanäle grau hinterlegt und die Rollen Fed-GM sowie Initiator weggelassen. Die Interaktion zwischen der fedDomain und den Entitäten kann auf die Domäne interfedDomain übertragen werden, um so ein Organisationsmodell für Inter-Federated Identity Management zu erhalten; zur Übersichtlichkeit wurde jedoch darauf verzichtet. Somit zeigt die Abbildung 4.7 das Metamodell des FIM/Inter-FIM-Organisationsmodells. Zur Beschreibung der FIM/Inter-FIM-Rollen wird der Stereotyp *FIM-Role* als Erweiterung von UML-Klassen verwendet. Dabei definiert die Rolle einen Funktionsumfang, der von einer Organisation, einer Person oder allgemein von einer Entität bereitgestellt bzw. ausgeführt wird. Jede Entität kann mehrere Rollen gleichzeitig ausfüllen. Eine Domäne setzt sich aus einer oder mehreren Rollen zusammen, die sich über einen Interaktionskanal austauschen, der als Stereotyp modelliert ist. Ein Interaktionskanal bindet somit zwei Rollen innerhalb einer Domäne oder domänenübergreifend und bildet eine Schnittstelle.

In diesem Abschnitt wurde über Domänen, Rollen und Interaktionskanäle das Organisationsmodell des Federated Identity Managements spezifiziert sowie aufgezeigt, wie das Organisationsmodell des Inter-Federated Identity Managements definiert ist. Die entscheidenden Domänen sind *AADomain*, *IdPDomain*, *SPDomain*, *fedDomain* und *interfedDomain*. Entsprechende Rollen und Interaktionskanäle wurden als Stereotype definiert.

4.4. Informationsmodell

Das Informationsmodell bestimmt die Gesamtheit der Managementobjekte. Hierfür werden Managementobjekte zur Erbringung der Funktionalität definiert. Managementobjekte repräsentieren die Charakteristik der Ressource, die verwaltet wird. Das Informationsmodell muss somit spezifizieren, wie das Objekt identifiziert wird, aus was es besteht, wie es sich verhält, wie es manipuliert werden kann, in welchen Verhältnis es zu anderen Objekten besteht und wie es bedient wird. Das Informationsmodell einer Managementarchitektur definiert das Modell und die einheitliche Notation, um die Managementinformationen zu beschreiben. Dazu muss eine Spezifikationssprache für die Informationsmodellierung festgelegt werden. In dieser Arbeit wird UML verwendet. Die Managementinformationen können aus verschiedenen Perspektiven betrachtet werden, wie bereits im Organisationsmodell zu sehen. Mögliche Sichtweisen sind Ressourcen, Prozeduren, Transaktionen und Kunden.

Eine Strukturierung des untersuchten Sachgebiets ist notwendig. Das Informationsmodell

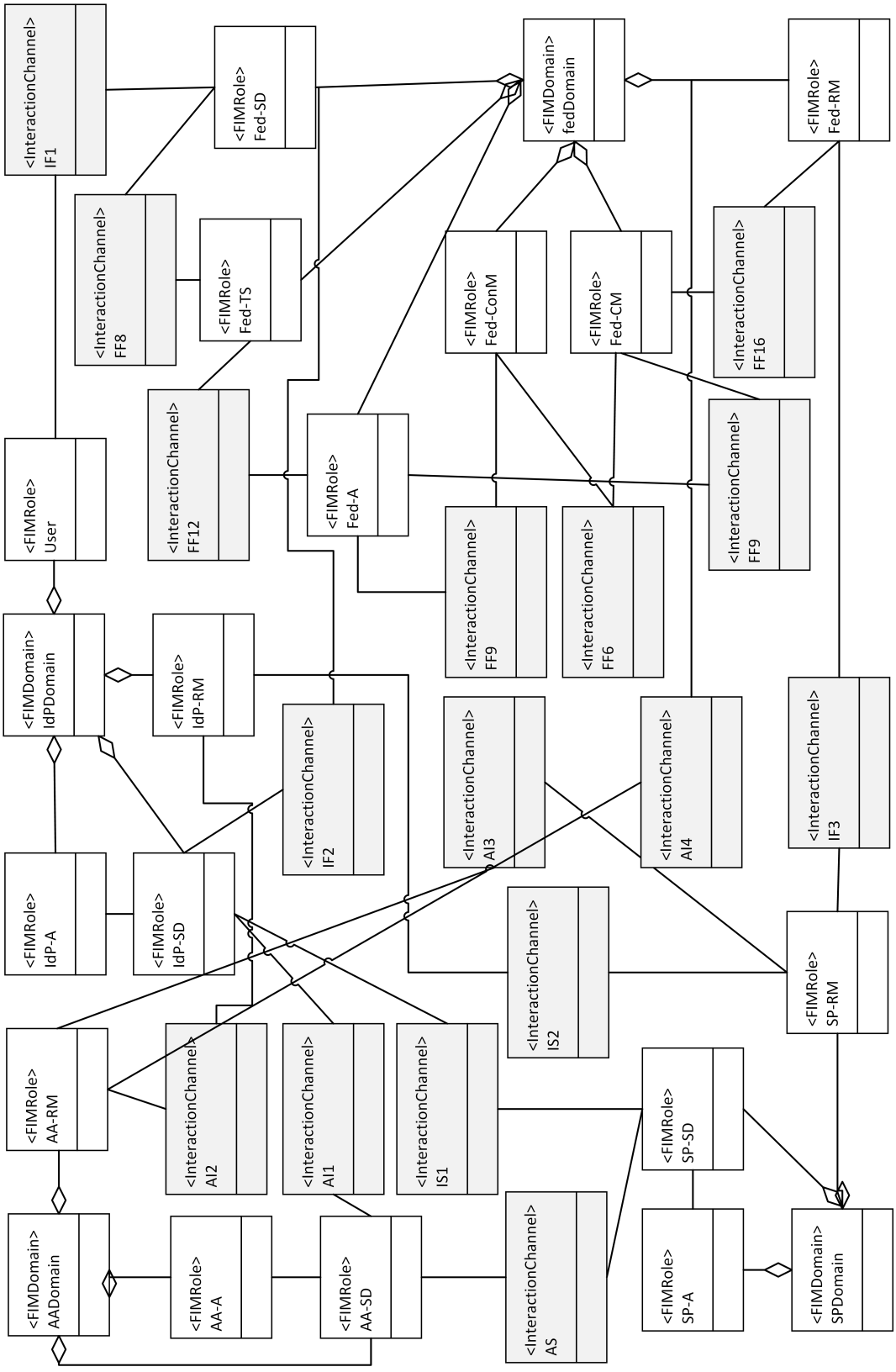

Abbildung 4.7.: Federated Identity Management Organisationsmodell

besteht aus separaten generischen Domänen, die als thematisch orientierte Gruppierung von Entitäten zur Strukturierung von Modellen fungieren. Somit stellen sich in diesem Abschnitt die folgenden Fragen:

- Aus welchen Domänen besteht die Managementarchitektur?
- Wie sind diese Domänen aufgebaut, d. h. welche Entitäten kommunizieren?
- Welche Informationen werden in welchem Format ausgetauscht und welche Ressourcen existieren?

Diese Fragen werden nachfolgend generisch beantwortet.

4.4.1. Domänen des Informationsmodells

Basierend auf dem Organisationsmodell wird zunächst die Domäne `Role` eingeführt, um formal die unterschiedlichen Domänen, Rollen und Interaktionskanäle übergreifend zusammenfassen zu können (vgl. Tabelle 4.27).

Domäne	*Role*
Bezeichner	ROL
Beschreibung	Domäne für das Management von Rollen
Spezifikation	4.4.10

Tabelle 4.27.: Zusammenfassung der Domäne Role

Die Rollen sind, wie im Organisationsmodell zu sehen, in Föderationen, Inter-Föderationen und verschiedenen Entitäten vertreten. Zunächst wird die Modelldomäne `Federation` gebildet. Das managed object `Federation` kann, wie in Abschnitt 4.2 beschrieben, verschiedene Ausprägungen annehmen. Die Repräsentation der Föderation wird in der Domäne `Federation` (vgl. Tabelle 4.28) vorgenommen, in der sämtliche Informationen zur Struktur und zu den Rollen definiert werden.

Domäne	*Federation*
Bezeichner	FED
Beschreibung	betrachtet Federation als managed object
Spezifikation	4.4.3

Tabelle 4.28.: Zusammenfassung der Domäne Federation

Als nächste Ausprägung wird die Modelldomäne `Inter-Federation` gebildet. Ebenso wie die Domäne `Federation` durchläuft die *Inter-Federation* den Lebenszyklus (vgl. Tabelle 4.29).

Domäne	*Inter-Federation*
Bezeichner	INT
Beschreibung	betrachtet Inter-Federation als managed object
Spezifikation	4.4.4

Tabelle 4.29.: Zusammenfassung der Domäne Inter-Federation

Jede Föderation besteht aus mehreren Entitäten, die in der Modelldomäne `Entity` zusammengefasst werden. Um Entitäten und ihre Prozesse zu unterstützen, wird die Domäne `Entity` (vgl. Tabelle 4.30) gebildet.

Domäne	*Entity*
Bezeichner	ENT
Beschreibung	Domäne von Entitäten, d. h. IdPs, SPs und AAs
Spezifikation	4.4.5

Tabelle 4.30.: Zusammenfassung der Domäne Entity

Die Zweckorientierung von Föderation erfordert ein adäquates Management von Mitgliedschaften. Die Mitgliedschaft von Entitäten kann zwar auch als Rolle aufgefasst werden, zu Modellierungszwecken werden jedoch alle betreffenden Informationen in der Domäne `Member` modelliert (vgl. Tabelle 4.31). Die Domäne ist für das Einreichen, Überprüfen, Akzeptieren, Ablehnen, Hinzufügen, Entfernen und Ändern von Mitgliedschaften in Föderationen und Inter-Föderationen verantwortlich.

Domäne	*Member*
Bezeichner	MEM
Beschreibung	Management von Mitgliedschaften zu FED und INT
Spezifikation	4.4.6

Tabelle 4.31.: Zusammenfassung der Domäne Member

Unabhängig von Föderationen, können Entitäten ihre Metadaten austauschen, wenn Nutzer einen Dienst verwenden möchten und sich IdP und SP bisher nicht technisch vertrauen. Um Metadaten austauschen zu können, müssen diese verwaltet werden, was in der Domäne `Metadata` geschieht. Die Domäne ist in der Tabelle 4.32 dargestellt.

Domäne	*Metadata*
Bezeichner	MET
Beschreibung	Management von Metadaten der Entitäten
Spezifikation	4.4.8

Tabelle 4.32.: Zusammenfassung der Domäne Metadata

Ob Identity Provider und Service Provider den automatischen Metadatenaustausch erlau-

ben, hängt mit dem eingehenden Risiko ab. Dieses wird in der Domäne `Trust` modelliert und verglichen.

Domäne	*Trust*
Bezeichner	TRU
Beschreibung	Management von Vertrauen zwischen Entitäten
Spezifikation	4.4.7

Tabelle 4.33.: Zusammenfassung der Domäne Trust

Damit SPs die Attribute der IdPs verstehen, müssen Identity Provider ihre Benutzerinformationen in ein SP-verständliches Format konvertieren. Die Ausprägung der Konvertierung wird im folgenden Kapitel näher erläutert. Für die Modellierung wird als Black Box die Domäne `Conversion Rule` verwendet, wie in Tabelle 4.34 zu sehen.

Domäne	*Conversion Rule*
Bezeichner	CR
Beschreibung	Management von Konvertierungsregeln
Spezifikation	4.4.9

Tabelle 4.34.: Zusammenfassung der Domäne Conversion Rule

Föderationen werden in dieser Arbeit als managed objects betrachtet. Sie werden nach Gesichtspunkten eines effizienten und effektiven Managements gruppiert, um u. a. die Verantwortlichkeiten und administrative Bereiche festzulegen. Hierbei spielen Policies und der Aufnahmeprozess in die Föderation eine Rolle. Diese Aspekte des Managements werden in der Domäne `Management` modelliert (vgl. Tabelle 4.35).

Domäne	*Management*
Bezeichner	MAN
Beschreibung	Domäne für Föderations- und Inter-Föderations-spezifisches Management
Spezifikation	4.4.11

Tabelle 4.35.: Zusammenfassung der Domäne Management

Das Management von Föderationen und Inter-Föderationen stellt ein komplexes und vielschichtiges Problem dar, welches in dieser Arbeit systematisch untersucht wird. Mit der Bereitstellung der Managementarchitektur wird das Ziel verfolgt, Föderationen (und Inter-Föderationen) als managed objects zu behandeln und den Metadatenaustausch über eine TTP zu automatisieren. Das in diesem Abschnitt zu entwickelnde Informationsmodell der Architektur legt die für das Management relevante Objekte und deren Beziehungen so fest, dass sie leicht erweiterbar sind. Basierend auf dieser allgemeinen Darstellung der Domänen werden in den nächsten Abschnitten die einzelnen Domänen detailliert diskutiert. Abbildung 4.8 umfasst die eben genannten Domänen im Inter-FIM-Informationsmodell zusammen. Zusätzlich wird die Domäne `Specification` erstellt, um die Gesamtheit der Spezifikationen

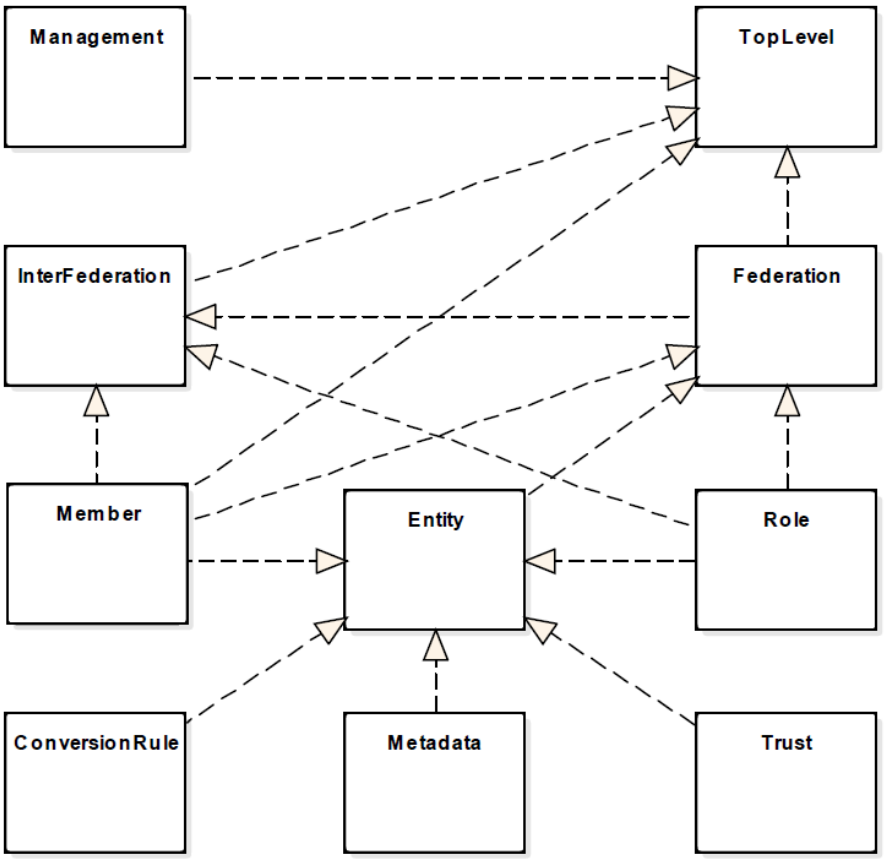

Abbildung 4.8.: Domänen des Inter-FIM Informationsmodells

zu beschreiben. Die Domänen `TopLevel` repräsentieren die allgemeinsten Klassen des Modells. Sie enthalten somit die Wurzelentitäten und die weitgehend abstrakten Oberklassen. Diese Oberklassen sowie die weiteren Domänen werden in den folgenden Abschnitten in ihrem Konzept dargestellt. Dabei wird aus Übersichtsgründen vor allem von FIM ausgegangen; das Modell lässt sich jedoch leicht auf Inter-FIM erweitern.

4.4.2. Die Domäne TopLevel

Die Domäne `TopLevel` repräsentiert die allgemeinste Klasse des Modells, indem alle Wurzel-Entitäten und weitgehend abstrakten Oberklassen enthalten sind. Sie liegt somit im Fokus zur Bereitstellung allgemeiner Entitäten. Die TopLevel `Domäne` beinhaltet abstrakte und generische Wurzelklassen, von denen die meisten Klassen sich ableiten lassen. Die Domäne ist in Abbildung 4.9 dargestellt.

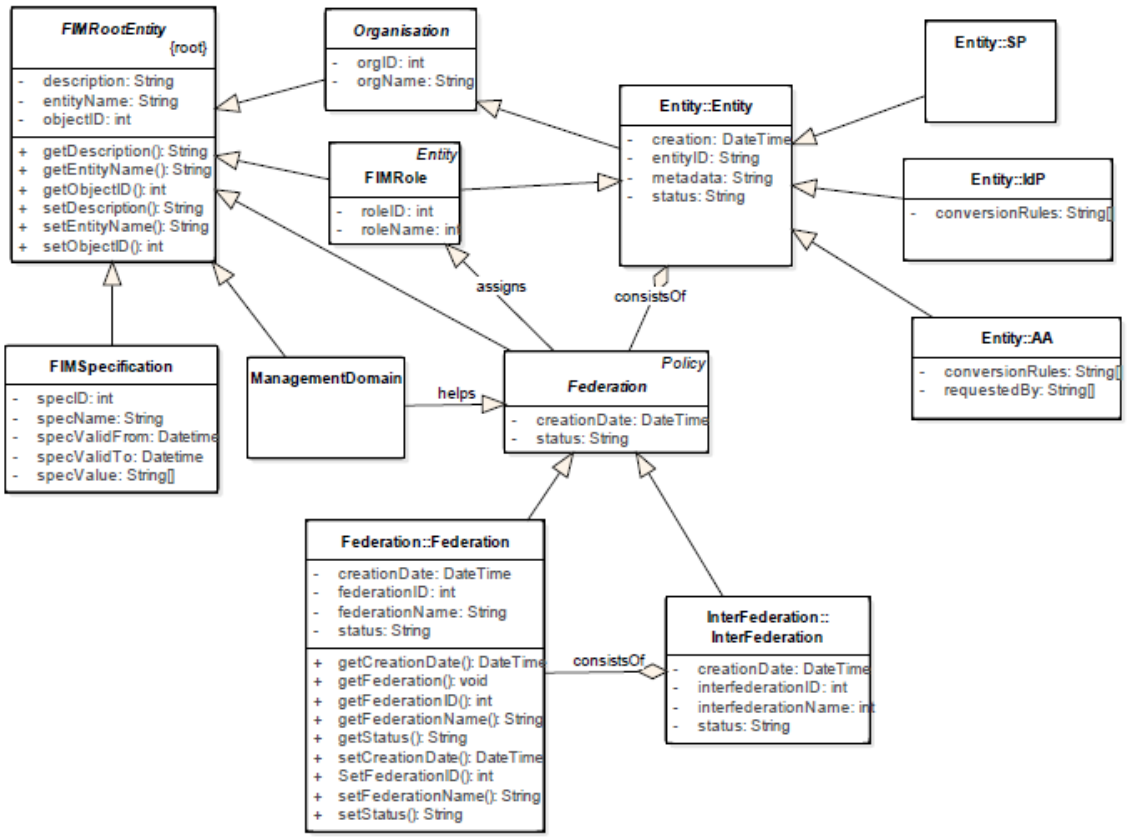

Abbildung 4.9.: Die Domäne TopLevel

Die Klasse `FIMRootEntity` wird als Wurzelklasse der Klassenhierarchie des Inter-FIM-Informationsmodells eingeführt. Dabei wird ein Satz von Attributen und Methoden definiert, die alle (Inter-)FIM-Entitäten benötigen:

- `objectID` wird zur eindeutigen Identifizierung von Objektinstanzen eingesetzt.
- `entityName` benennt die (Inter-)FIM-Entität.
- `description` beschreibt die (Inter-)FIM-Entität.

Über die Getter- und Setter-Methoden können diese Informationen gelesen und verändert werden.

Die `FIMRootEntity` wird durch fünf Klassen spezifiziert: `FIMSpecification`, `ManagementDomain`, `Federation`, `Role` und `Entity`, die über `Organization` an RootEntity angebunden ist. `Federation` ist eine abstrakte Klasse, die die beiden Klassen `Federation` und `InterFederation` enthält. `Entity` ist ebenfalls eine abstrakte Klasse, die durch die Klassen `IdP`, `SP`

und AA beschrieben wird. Diese Klassen werden in den nächsten Abschnitten weiter erläutert.

4.4.3. Die Domäne Federation

Die Domäne Federation beschreibt den Zusammenschluss von Entitäten zu einer Föderation. Die Klasse Federation ist bewusst generisch gehalten, spezialisiert die generischen Klasse Federation und stellt jedoch einfache Attribute zur Verfügung:

- federationID zur eindeutigen Identifizierung.
- federationName zur Beschreibung der Föderation.
- status zur Beschreibung des aktuellen Status bezüglich des Lebenszyklus über Vererbung.
- creationDate zur Beschreibung der Erstellung über Vererbung.

Zudem sind folgende Methoden vorhanden:

- getCreationDate(), um das Gründungsdatum der Föderation zu erhalten.
- getFederation(), um die Informationen über eine Föderation zu bekommen.
- getFederationID(), um die ID der Föderation abzufragen.
- getFederationName(), um den Namen der Föderation zu erhalten.
- getStatus(), um den Status der Föderation abzufragen.
- setCreationDate(), um das Gründungsdatum bei der Initiierung der Föderation zu setzen.
- setFederationID(), um die ID der Föderation bei der Initiierung zu setzen.
- setFederationName(), um einen Föderationsnamen zu schreiben.
- setStatus(), um einen Status zu speichern.

Die Klasse Federation ist wiederum aufgeteilt in zwei weitere Klassen, die die Föderation genauer beschreiben, wie in Abbildung 4.10 zu sehen. Die Klasse DynamicVirtualFederation stellt eine Spezialisierung der Klasse Federation für dynamische virtuelle Föderationen dar. Die Klasse FixedFederation ist das Gegenstück für Föderationen, die einen Aufnahmeprozess durchlaufen. Entsprechend hat diese Spezialisierung weitere Attribute und Methoden:

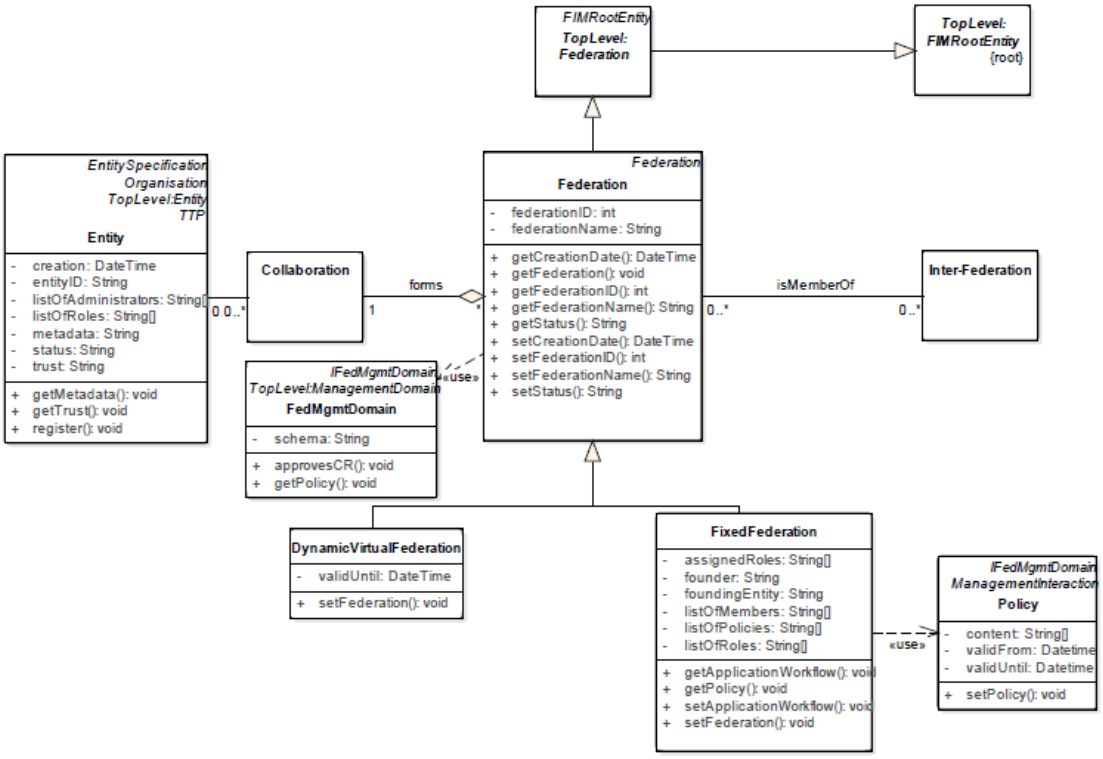

Abbildung 4.10.: Die Domäne Federation

- `assignedRoles` beschreibt die Rollen, die Mitgliedern gegeben wurden.
- `founder` bezeichnet den Gründer der Föderation.
- `foundingEntity` benennt die Entität, die für die Gründung der Föderation verantwortlich ist.
- `listOfMembers` enthält die Liste der Mitglieder.
- `listOfPolicies` enthält alle Policies der Föderation.
- `listOfRoles` beschreibt alle vorhandenen Rollen.
- `getApplicationWorkflow()` gibt den Aufnahmeprozess wieder, während `setApplicationWorkflow()` den Aufnahmeprozess festlegt.
- `getPolicy()` gibt die aktuell gültige Policy der Föderation wieder.
- `setFederation()` erstellt eine neue Föderation.

Feste Föderationen können über die Klasse `Policy` neue Policies erstellen. Dazu werden die Methode `setPolicy()` und die Attribute `validFrom`, `validUntil` und `content` bereitgestellt. Föderationen werden über die Klasse `FedMgmtDomain` verwaltet. Föderationen können Mitglieder in Inter-Föderationen sein und werden durch Entitäten und ihre Kollaborationen gegründet.

Äquivalent zu den Konvertierungsregeln werden im Datenmodell die Informationen zu `status`, `location` und `permissions` für Policies benötigt. Zudem werden Name der Policy, Datum der Erstellung sowie die zugehörige Föderation gespeichert, woraus sich folgende Informationen ergeben:

- `policyID` zur eindeutigen Identifizierung.
- `policyName`: Name der Policy.
- `status` zur Beschreibung des Status.
- `location` Ort der Policy. Die Policy kann entweder bei MdFIM abgelegt sein oder über eine URL aufgerufen werden können.
- `permissions` zur Beschreibung der Berechtigungen.
- `creationDate` zur Beschreibung der Erstellung der Policy.
- `federation`: Verknüpfung zur Föderation, die die Policy erstellt hat.

4.4.4. Die Domäne Inter-Federation

Wie in Abbildung 4.11 zu sehen, sind die Domänen `Federation` und `Inter-Federation` ähnlich aufgebaut. Im Gegensatz zu `Federation` bilden sich `Inter-Föderationen` meist aus Föderationen, was bei der Modellierung beachtet wurde.

4.4.5. Die Domäne Entity

Die Domäne `Entity` beschreibt, wie in Abbildung 4.12 zu sehen, Entitäten. Die Oberklasse `Entity` hat dabei die folgenden Attribute:

- `creation` hält den Tag der Registrierung fest, die über die Managementplattform validiert wird.
- `entityID` beschreibt eindeutig die Entität.
- `metadata` gibt die Speicherart wieder, die entweder eine URL oder ein Repository (bzw.

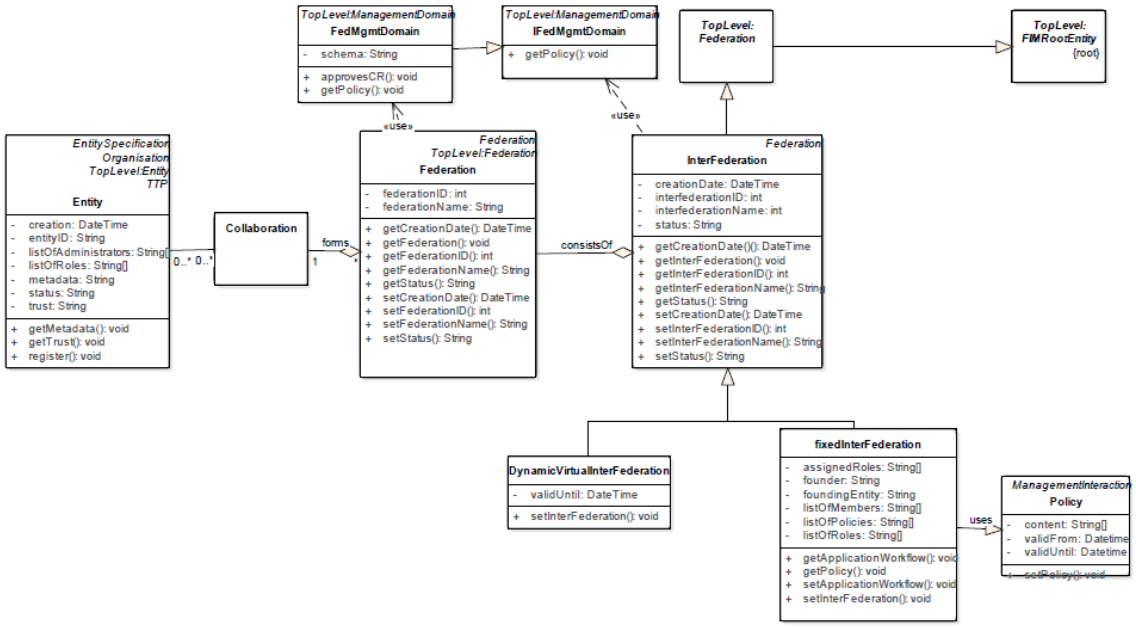

Abbildung 4.11.: Die Domäne Inter-Federation

Dateisystem) ist.

- `status` beschreibt den Status der Entität, wie registriert aber nicht validiert, validiert, ungültig oder am Ausscheiden.
- `listOfRoles` beschreibt die vergebenen Rollen.
- `listOfAdministrators` enthält eine Liste der Administratoren.
- `trust` beschreibt das Vertrauen als LoT/LoA oder anderen Wert für die Entität.
- `displayName` und `url` werden benötigt, um den Lokalisierungsdienst zu bedienen.

Über das Entitäten-Management wird die Entität durch die Administratoren verwaltet. Diese allgemeine Klasse `Entity` gliedert sich in drei weitere, spezifischere Klassen auf: `IdP`, `SP` und `AA`. Die Unterscheidung in Entitäten kann über ein weiteres Attribut, `entityType`, realisiert werden. Nachdem verschiedene Workflows der Einbindung von Attribute Authorities unterstützt, sollen verknüpfte AAs auch gespeichert werden. Während IdP und AA Konvertierungsregeln benötigen und daher die verwendeten Konvertierungsregeln durch das Attribut `conversionRule` festhalten, geben Service Provider in ihren Metadaten die Attribute an, die sie verwenden. Diese sind durch die Methode `requestedAttributes` herauszufinden. Alle Entitäten werden durch Metadaten beschrieben, die in einer eigenen Domäne dargestellt werden.

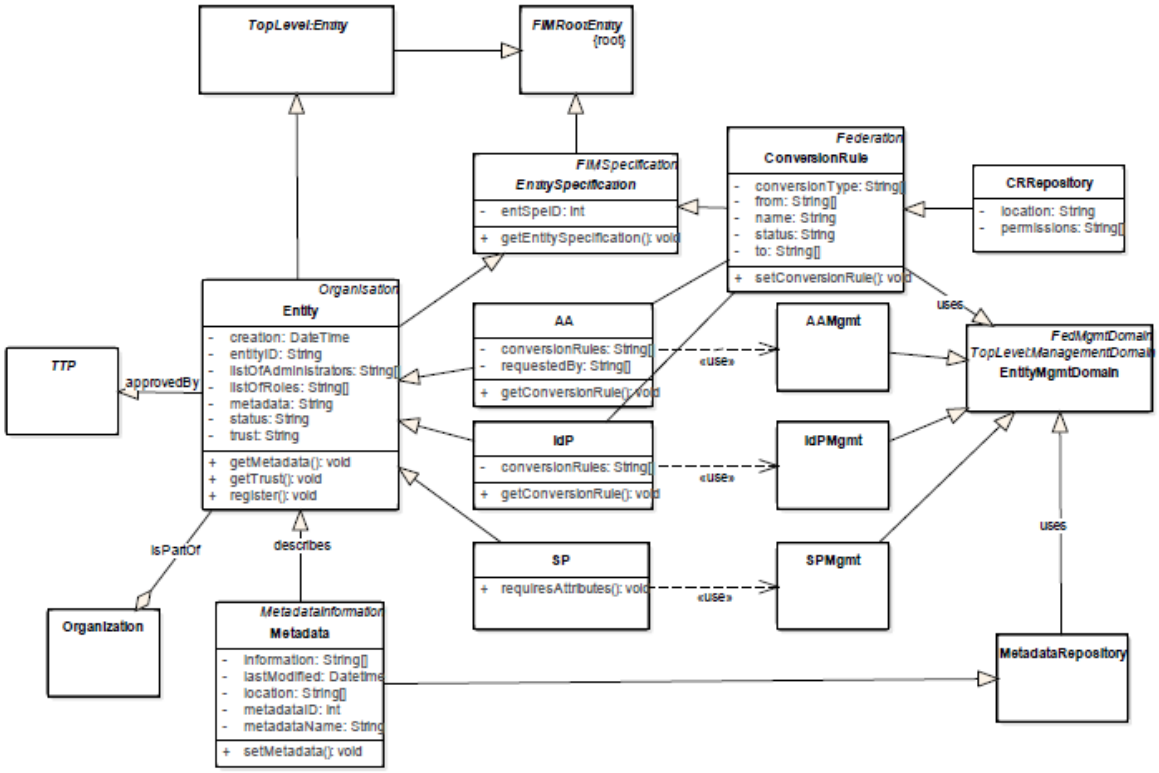

Abbildung 4.12.: Die Domäne Entity

4.4.6. Die Domäne Member

Die Domäne `Member` behandelt die Mitgliedschaft in Föderationen und Inter-Föderationen. Die Abbildung 4.13 zeigt vereinfacht die Mitgliedschaft in Föderationen. Entitäten bilden eine Föderation durch zwei Wege:

- Kollaboration, durch die Klasse `Collaboration` dargestellt.
- Bewerbung, durch die Klasse `Application` dargestellt.

Die Bildung von Föderationen über Bewerbungen geschieht über die entsprechende `Management` Domäne, die sich Policies zur Hilfe nimmt, um zu entscheiden, ob Entitäten den Anforderungen gerecht werden. Eine Föderation kann Entitäten zudem Rollen zuweisen, die sich im Laufe des Lebenszyklus ändern können.

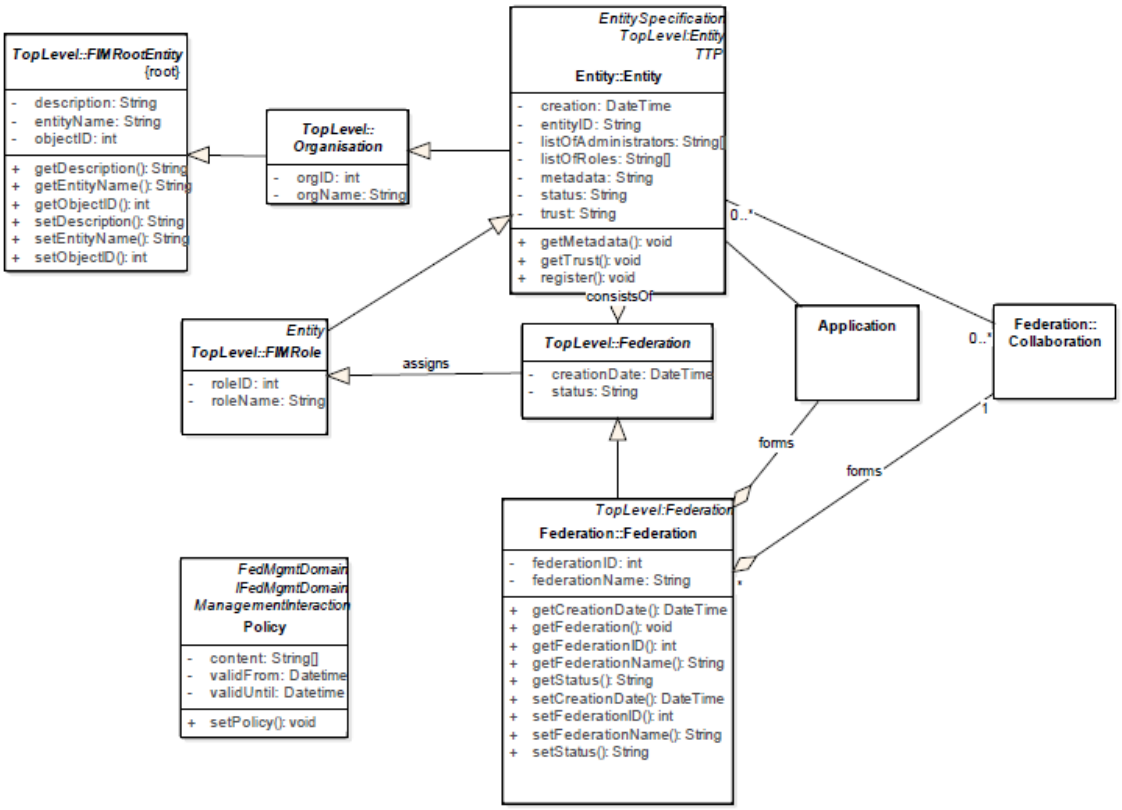

Abbildung 4.13.: Die Domäne Member

4.4.7. Die Domäne Trust

Die Domäne Trust modelliert, wie in Abbildung 4.14 zu sehen, Vertrauensbeziehungen. Entitäten, beschrieben durch die Klasse `Entity`, besitzen Vertrauensinformationen in Form von LoT, LoA oder anderen Kennzahlen, die durch die Klasse `Trust` verarbeitet werden. Die Klasse enthält dabei die Attribute

- `fromEntity`: Verweis auf den Startpunkt der Berechnung.
- `toEntity`: Verweis auf den Endpunkt der Berechnung.
- `status`: Status des Vertrauens.
- `creationDate`: Zeitstempel, wann das Vertrauen aufgebaut wurde.

Die Berechnung wird durch die Methode `calculateTrust()` durchgeführt. Die Vertrauensinformationen sind in den Metadaten der Entität enthalten und beschreiben somit das Vertrauen einer Entität, was ggf. durch eine Föderation überprüft wird. Die Hintergrundin-

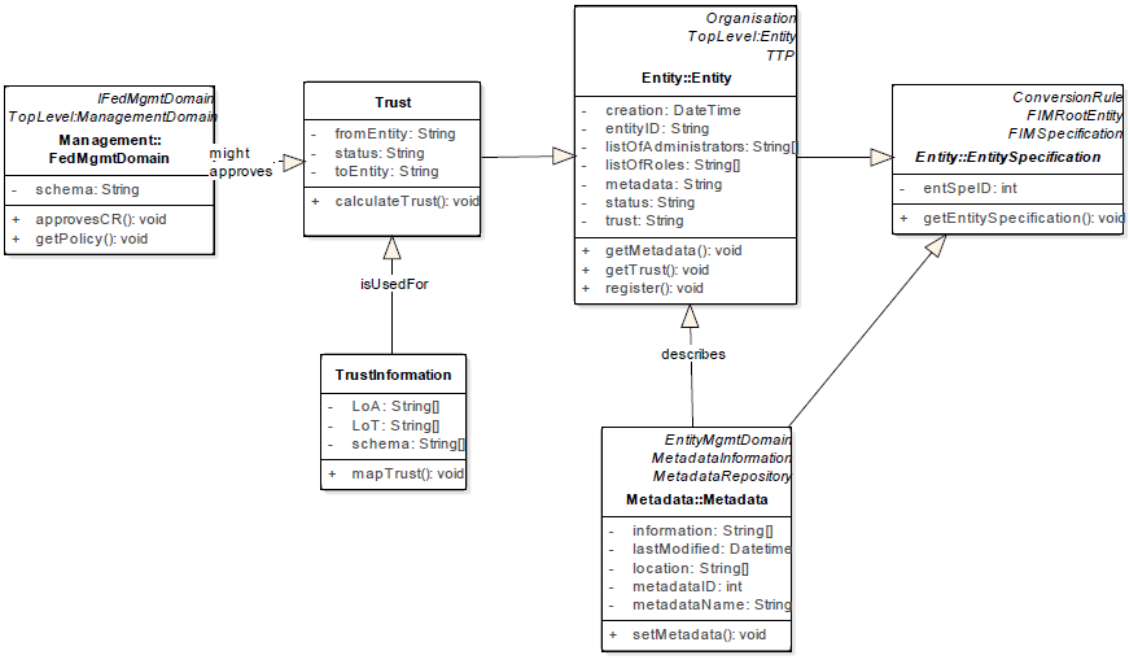

Abbildung 4.14.: Die Domäne Trust

formationen zur Berechnung des Vertrauens sind in der Klasse `TrustInformation` abgelegt. Dabei werden der jeweilige LoA und LoT sowie das zugehörige Schema gespeichert. Alternativ kann hier ein Verweis auf die IANA-Registry erfolgen. Die Methode `mapTrust` vergleicht, falls benötigt, die unterschiedlichen Vertrauensarten.

4.4.8. Die Domäne Metadata

Die Domäne `Metadata` beschreibt die Metadaten. Die Klasse `Metadata` ist hierbei die Hauptklasse mit den Attributen

- `information`, die alle Informationen über Metadaten enthält,
- `metadataID`, der eindeutigen ID der Metadaten,
- `metadataName`, dem Namen der Metadaten,
- `lastModified` mit dem Datum der letzten Aktualisierung,
- `location` mit dem Ort bzw. Art der Speicherung und
- `status` zur Beschreibung des aktuellen Status.

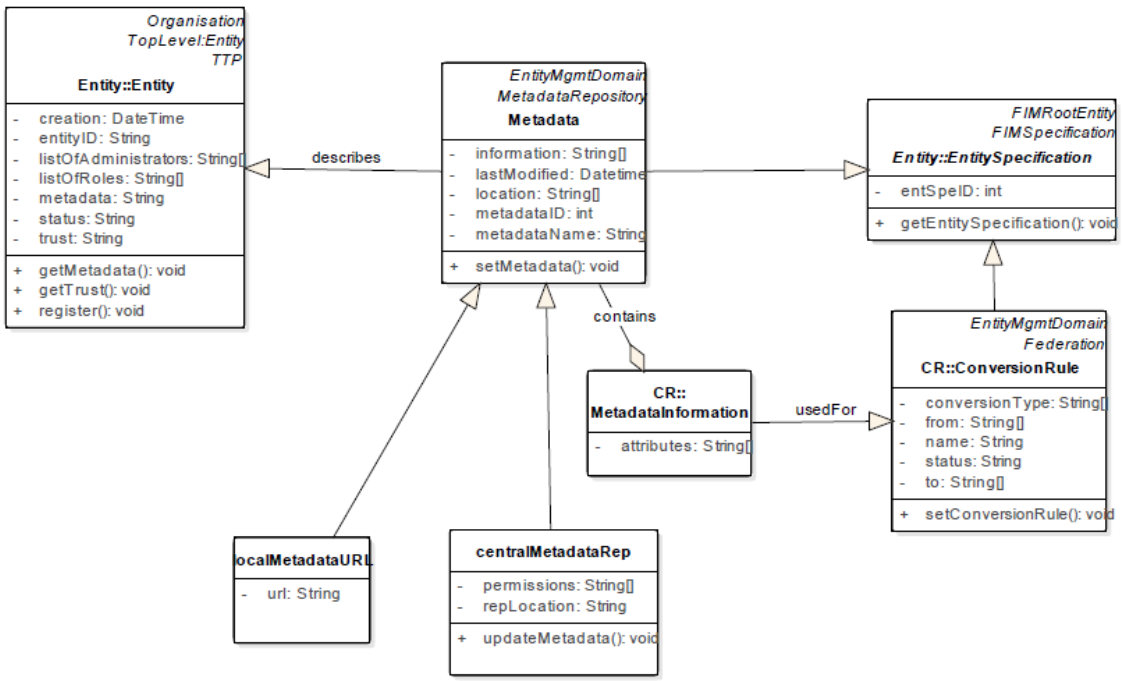

Abbildung 4.15.: Die Domäne Metadata

Die Klasse `Metadata` verwendet die Methode `setMetadata()`, um eine neue Version der Metadaten zu setzen. Das Attribut `location` verweist wiederum auf die beiden Spezialisierungen `localMetadataURL` und `centralMetadataRep`. `localMetadataURL` enthält das Attribute `url`, das auf die URL der gespeicherten Metadaten verweist. Die Klasse `centralMetadataRep` besteht aus den Attributen

- `permissions`, die die Berechtigungen für die Metadaten angibt und
- `repLocation`, die den Ort im zentralen Dateisystem beschreibt.

Bei der Verwendung des zentralen Dateisystems müssen die Metadaten aktualisiert werden können, um anschließend in der Webanwendung die verschiedenen Stände anzeigen und verwalten zu können. Die Aktualisierung wird über die Methode `updateMetadata()` angestoßen. Metadaten enthalten Informationen (`MetadataInformation`), die für Konvertierungsregeln benötigt werden, wie in der nachfolgenden Domäne erklärt. Metadaten beschreiben, wie in Abbildung 4.15 zu sehen, Entitäten.

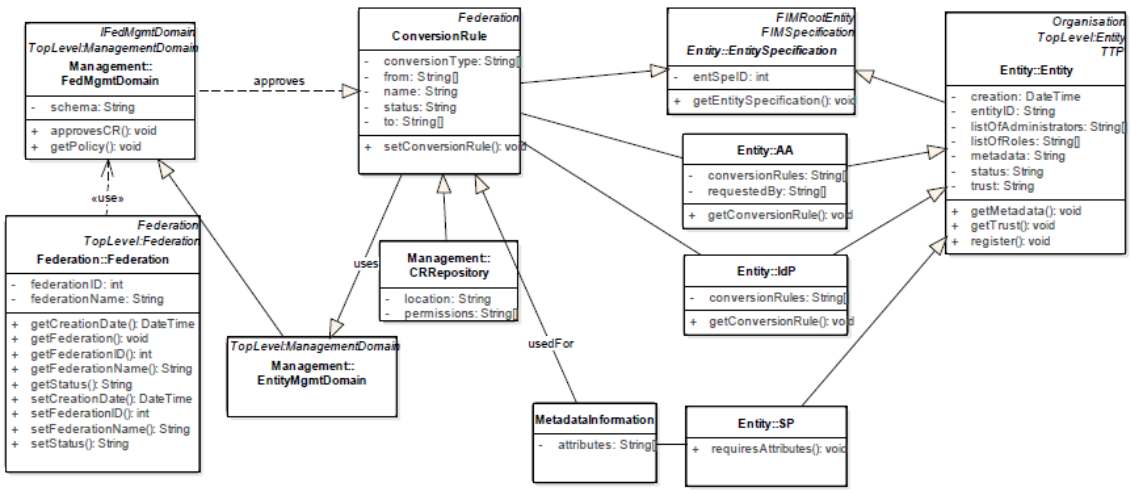

Abbildung 4.16.: Die Domäne Conversion Rule

4.4.9. Die Domäne Conversion Rule

Die Domäne `Conversion Rule` spezifiziert, visualisiert in Abbildung 4.16, Konvertierungsregeln und das Management von Konvertierungsregeln. Die Klasse `ConversionRule` besteht aus den Attributen

- `conversionType`, eine allgemeine Beschreibung der Art der Konvertierung.
- `from`, den Attributen, die zur Konvertierung verwendet werden.
- `name`, dem Namen der Konvertierung.
- `status`, dem Status der Konvertierung.
- `to`, den Attributen, die am Ende der Konvertierung entstehen sollen.

Über die Methode `setConversionRule()` kann eine neue Konvertierungsregel erstellt werden, die durch IdPs und AAs über `getConversionRule()` abgerufen werden kann. Die Informationen in den Metadaten der Service Provider wird für die Konvertierungsregeln verwendet, wie durch die Klasse `MetadataInformation` dargestellt. Die Konvertierungsregeln werden in einem Repository, siehe Klasse `CRRepository`, gespeichert. Die Klasse `CRRepository` besteht aus den Attributen `location` und `permissions`, um den Ort innerhalb des Dateisystems und den Berechtigungen zu beschreiben. Konvertierungsregeln können durch die Föderation akzeptiert werden, was durch die Klasse `FedMgmtDomain` visualisiert ist. Jede Föderation kann ein Schema festlegen (Attribut `schema`) und eine Konvertierungsregel akzeptieren (Methode `approvesCR`), was ihr einen besseren Status gewährt.

4.4.10. Die Domäne Role

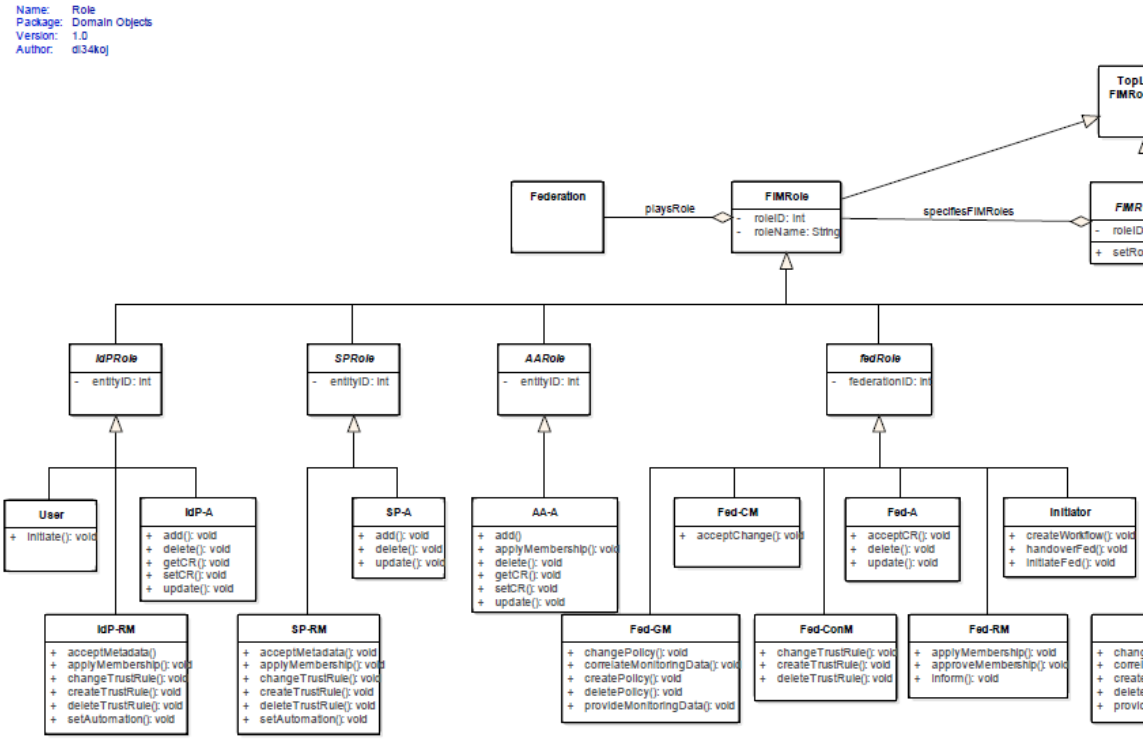

Abbildung 4.17.: Die Domäne Role 1/2

Die Domäne `Role` stellt die Modellierung der organisatorischen Domänen und deren zugehörigen Rollen dar. Die Abbildungen 4.17 und 4.18 zeigen, dass die zwei abstrakten Klassen `FIMRole` und `RoleSpecification` Spezialisierungen der `RootEntity` sind. Durch die Klasse `RoleSpecification` der Domäne `Specification` wird die Rolle spezifiziert. Die Klasse `FIMRole` enthält die Attribute `roleID` und `roleName`, um die Rolle eindeutig zu beschreiben. Die Teilnehmer von dynamischem FIM und Inter-FIM werden als Rollen modelliert, wodurch die folgenden Klassen existieren:

- `IdPRole` repräsentiert die IdP-Domäne mit den spezifischen Klassen `User`, `IdP-A` und `IdP-RM` sowie dem Attribut `entityID`.
- `SPRole` repräsentiert die SP-Domäne mit den spezifischen Klassen `SP-A` und `SP-RM` sowie dem Attribut `entityID`.
- `AARole` repräsentiert die AA-Domäne mit der spezifischen Klasse `AA-A` und dem Attribut `entityID`.
- `FedRole` repräsentiert die Fed-Domäne mit den spezifischen Klassen `Fed-RM`, `Fed-CM`, `Fed-ConM`, `Fed-A`, `Initiator` und `Fed-GM` sowie dem Attribut `fedID`.

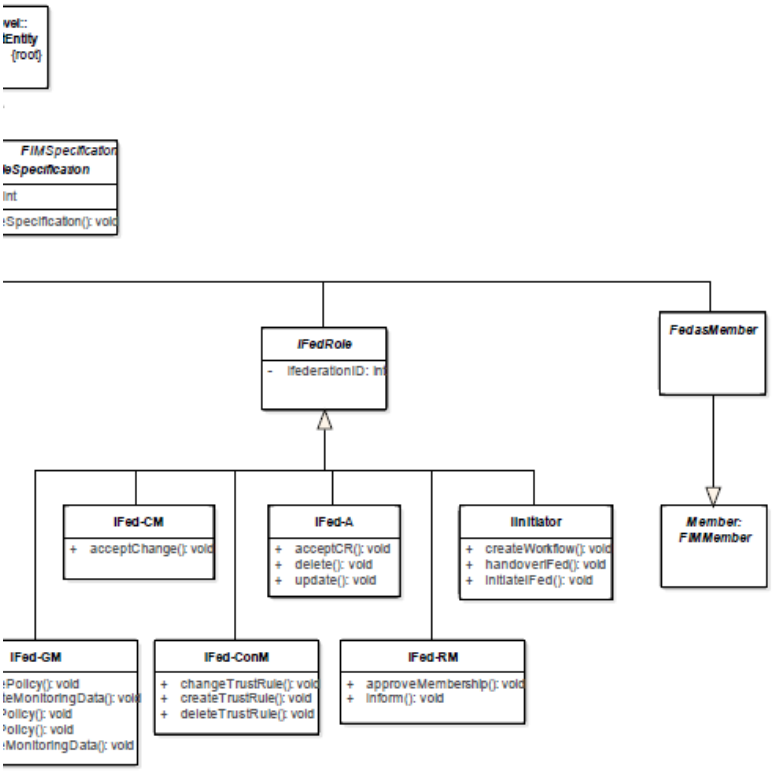

Abbildung 4.18.: Die Domäne Role 2/2

- `IFedRole` repräsentiert die IFed-Domäne mit den spezifischen Klassen `IFed-RM`, `IFed-CM`, `IFed-ConM`, `IFed-A`, `IInitiator` und `IFed-GM` sowie dem Attribut `ifedID`.

Die jeweiligen Rollen sind über die Klassen hinweg ähnlich modelliert, wodurch es folgende Methoden gibt:

- `initiate()` in der Klasse `User` initiiert den Metadatenaustausch.
- `add()`, `update()` und `delete()` in der Rolle Administrator (A). `add()` wird benötigt, um allgemeine Informationen hinzuzufügen, während `update()` diese Informationen ändert und `delete()` sie löscht. Die Klasse `AA-A` enthält zudem über die Methode `applyMembership()`. Die Klassen `IdP-A` und `AA-A verfügen` über weitere Methoden, um Konvertierungsregeln zu verwalten.
- Die Rollen Relationship Manager (RM) der Klassen `IdPRole` und `SPRole` besitzen die Methoden
  - `acceptMetadata()`,

  - `applyMembership()`,
  - `changeTrustRule()`,
  - `createTrustRule()`,
  - `deleteTrustRule()` und
  - `setAutomation()`.

  Durch die Methode `acceptMetadata()` werden Metadaten akzeptiert. `applyMembership()` ermöglicht die Mitgliedschaft in einer Föderation zu erwerben. Die Methoden `changeTrustRule()`, `createTrustRule()` und `deleteTrustRule()` sind generische Methoden, um mögliche Vertrauensinformationen zu verwalten. Hierbei können der benötigte LoA bzw. LoT ebenso gesetzt werden, wie Blacklists und Whitelists oder andere Vertrauenskonfigurationen. Über die Methode `setAutomation()` wird die Automatisierung des Metadatenaustausches konfiguriert.

- Die Rollen RM der Klassen `FedRole` und `IFedRole` verfügen über die beiden Methoden `approveMembership()` und `inform()`. Die Klasse `FedRole-RM` enthält zudem die Methode `applyMembership()`. Die Methoden zur Mitgliedschaft werden benötigt, um in einer Inter-Föderation aufgenommen werden zu können und mögliche Mitglieder zu akzeptieren oder abzuweisen. Die Mitglieder werden durch `inform()` über Änderungen der Föderation bzw. Inter-Föderation informiert.

- Die Klassen `Initiator` und `IInitiator` können die folgenden Methoden ausführen: `createWorkflow()`, `handoverFed()` bzw. `handoverIFed()` und `initiateFed()` bzw. `initiateIFed()`. Diese Methoden beschreiben die ursprüngliche Erstellung einer Föderation bzw. Inter-Föderation.

- Die Rolle ConM erhält die Methoden `changeTrustRule()`, `createTrustRule()` und `deleteTrustRule()`, um Regeln zum Vertrauen innerhalb einer Föderation oder Inter-Föderation festzulegen.

- CM der Klassen `Fed-CM` und `IFed-CM` verfügt über die Methode `acceptChange()`, um Änderungen zu akzeptieren oder abzulehnen.

- GM verfügt über die Methoden `changePolicy()`, `correlateMonitoringData()`, `createPolicy()`, `deletePolicy()` und `provideMonitoringData()`. Diese Methoden dienen zum einen dazu Policies zu verwalten und zum anderen Informationen über die Föderation bzw. Inter-Föderation bereitzustellen.

Auf Grund der Registrierung und Administration ist eine Art Benutzerverwaltung im Datenmodell sinnvoll. Es kann vorkommen, dass pro Entität mehrere Benutzer Zugriff benötigen, aber auch, dass ein Benutzer mehrere Entitäten verwaltet. Die Benutzer können unterschiedliche Rollen annehmen und folglich verschiedene Berichtigungen haben. Für weitere

Benachrichtigungen wird eine E-Mail-Adresse benötigt. Die Benutzerinformationen können durch FIM vom Identity Provider abgerufen werden, da im Hintergrund ein Service Provider eingebunden wird. Nachdem es sein kann, dass eine Organisation zwar einen SP betreibt, aber keinen IdP und da ein Identity Provider sich erst einmal registrieren muss, bevor er im Lokalisierungsdienst erscheint, ist eine Parallellösung von FIM und lokale Benutzerverwaltung sinnvoll. Anschließend, beim zweiten Fall, müssen intern der angelegte Nutzer und die Benutzerinformationen vom IdP verknüpft werden können. Hierfür wird ein eindeutiges Attribut benötigt, um die Verknüpfung durchzuführen. Für die lokale Benutzerverwaltung wird für ein gehashten Passwort zudem ein Salt verwendet, um das Erraten des Passworts zu erschweren. Zudem soll ein Zeitstempel der letzten Änderung gespeichert werden. Insgesamt zeigt sich folgendes Bild:

- `username`: Benutzername.
- `uniqueName`: `eduPersonTargetedID` oder ähnliches einmaliges Attribut zum Verknüpfen von Benutzerkonten.
- `password`: Gehashtes Passwort mit Salt.
- `givenName`: Vorname des Benutzers, optional.
- `surname`: Nachname des Benutzers, optional.
- `email`: E-Mail-Adresse des Nutzers für weitere Benachrichtigungen.
- `creationDate`: Zeitstempel, wann der Benutzer angelegt wurde.
- `lastModified`: Zeitstempel der letzten Änderung.
- `role`: Rollen, die der Nutzer in einem bestimmten Kontext hat.

4.4.11. Die Domäne Management

Die Domäne `Management` stellt die Sachverhalte dar, die für die Verwaltung von FIM und Inter-FIM nötig sind. Dazu zählen das Bereitstellen von Diensten und Benutzerinformationen, Verwalten von Föderation und Inter-Föderationen, Entitäten sowie das Bereitstellen von Policies. Ferner werden Metadaten und Konvertierungsregeln verwaltet, wie in Abbildung 4.19 zu sehen. Die Klasse `ManagementDomain` stammt hierbei von der `RootEntity`-Klasse ab und wird durch drei weitere Klassen spezifiziert: `IFedMgmtDomain`, `FedMgmtDomain` und `EntityMgmtDomain`. Die Klassen `IFedMgmtDomain` und `FedMgmtDomain` werden verwendet, um Föderationen bzw. Inter-Föderationen zu verwalten. Die Informationen der Mitglieder werden dabei zusammengefasst. Zudem erhalten die beiden Klassen Informationen über ihre Policies. Die Klasse `EntityMgmtDomain` erhält wiederum drei spezialisierte Klassen: `IdP`, `SP` und `AA`. Diese verwalten ihre Metadaten und Konvertierungsregeln sowie Trust-

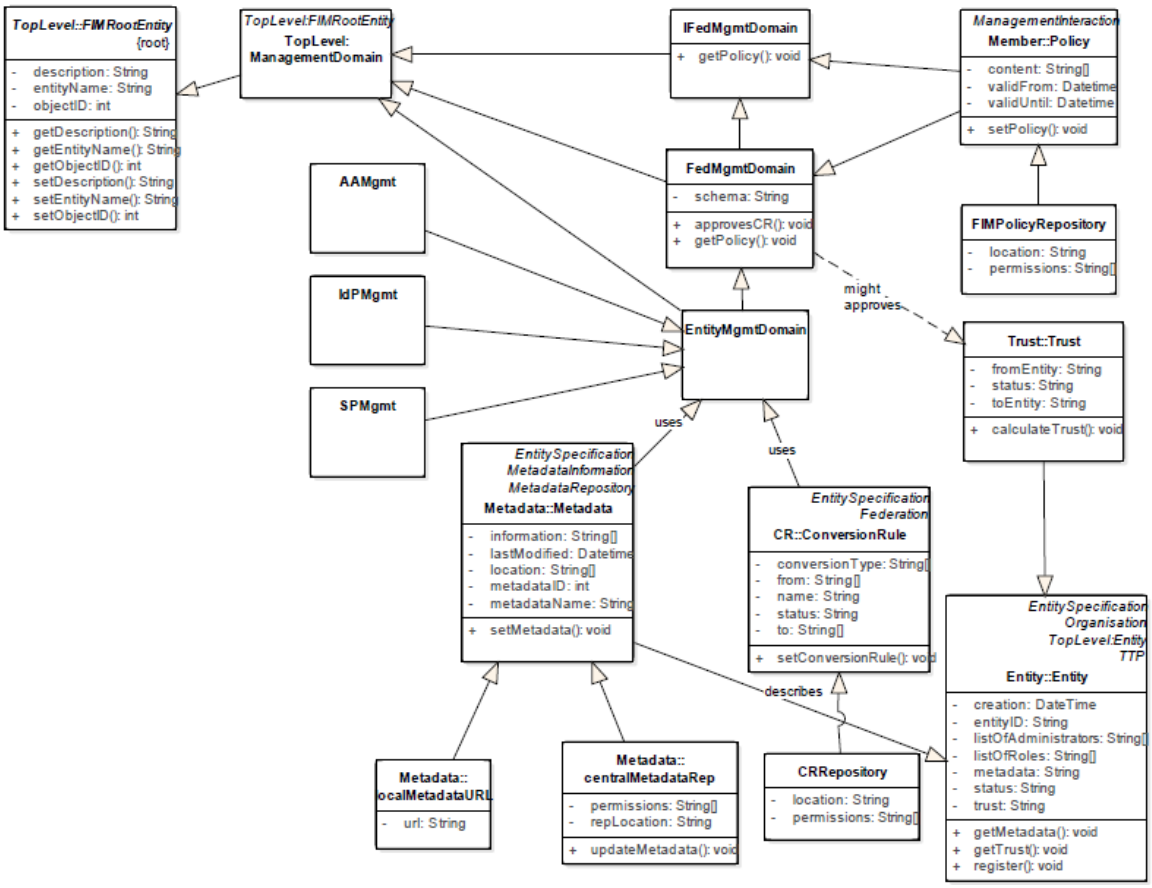

Abbildung 4.19.: Die Domäne Management

Informationen über diese Management-Domäne.

4.4.12. Die Domäne Specification

Die Domäne `Specification` stellt abschließend die gesamten Spezifikationen bereit, die für das Informationsmodell benötigt werden. Die abstrakte Klasse `FIMSpecification` wird über die folgenden Attribute definiert:

- `specID`: Eindeutige ID der Spezifikation.
- `specName`: Name der Spezifikation.
- `validUntil`: Gültigkeitsende der Spezifikation.
- `validFrom`: Gültigkeitsbeginn der Spezifikation.

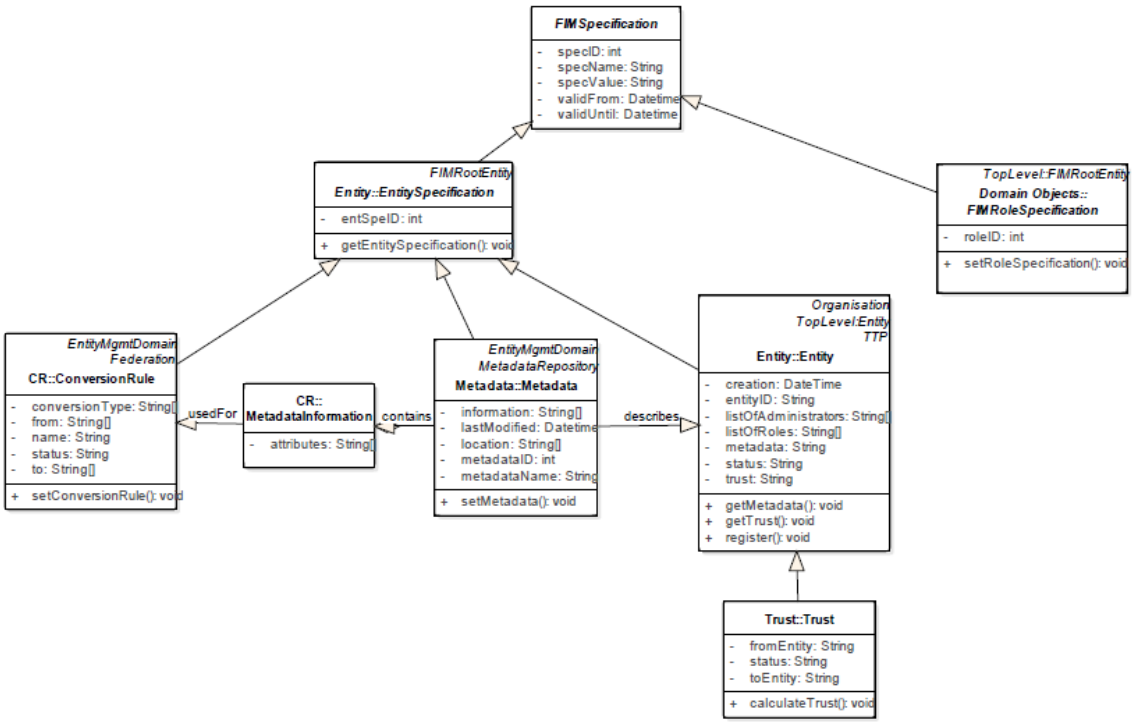

Abbildung 4.20.: Die Domäne Specification

- `specValue`: Wert der Spezifikation.

Die abstrakte Klasse wird durch zwei weitere Klassen spezifiziert: `EntitySpecification` und `RoleSpecification`. Von der Klasse `RoleSpecification` lassen sich alle Rollen des Informationsmodells ableiten. Die Klasse `EntitySpecification` spezifiziert alle weiteren Entitäten des UML-Modells, d. h. Entitäten des FIMs und Inter-FIMs in der Klasse `Entities`, Metadaten in der Klasse `Metadata`, Konvertierungsregeln in der Klasse `Conversion Rules` und indirekt Trust in der Klasse `Trust`.

Zur Beschreibung der Status einzelner Organisationen und Beziehungen wird ein Statuskonzept benötigt. Für den Provider ergeben sich folgende mögliche Status:

- `requested`: Provider hat sich registriert, aber seine Berechtigung noch nicht bestätigt.
- `valid`: Provider wurde bestätigt.
- `invalid`: Provider wurde nicht bestätigt.
- `deactivated`: Provider wurde auf Grund von deaktivierten Metadaten selbst deaktiviert.

Dementsprechend können Metadaten ebenfalls `valid` oder `invalid` sein. Wenn das enthaltene Zertifikat abgelaufen ist, wechselt der Status in `outdated`, bevor die Metadaten nach einer gewissen Zeitspanne deaktiviert werden. Eine Entität kann die Mitgliedschaft in einer Föderation beantragen (`requested`), die akzeptiert (`accepted`) oder abgelehnt (`declined`) wird. Zusätzlich ist es möglich, dass eine Entität aus einer Föderation ausgeschlossen (`excluded`) wird. Nachdem beim Austausch der Metadaten zwischen IdP und SP Fehler auftreten können, werden folgende beiden Status zusätzlich ermöglicht:

- `valid`: Der Austausch der Metadaten und die Konfiguration waren bei beiden Entitäten erfolgreich.
- `invalid`: Der Austausch der Metadaten und die Konfiguration waren bei mindestens einer Entität fehlerhaft und muss daher zurückgerollt werden.

Falls das Vertrauen nicht ausreicht, wird der Status `untrusted` gesetzt. Als Zwischenstatus ist folglich auch `trusted` möglich. Das beschriebene Informationsmodell mit den Domänen, Attributen und Methoden bildet die Basis für das im nächsten Abschnitt zu spezifizierende Kommunikationsmodell.

4.5. Kommunikationsmodell

Das Kommunikationsmodell liegt die Prinzipien und Konzepte zum Austausch von Managementinformationen zwischen den Akteuren fest und beschreibt damit auch, welche Mechanismen zum Austausch notwendig sind. Dabei sind laut Hegering et al. [HAN99] folgende Aspekte bei der Realisierung von besonderer Bedeutung:

- Welche Entitäten kommunizieren Managementinformationen?
- Wie sehen die Austauschformate für das Managementprotokoll aus?
- Wie sind Kommunikationsmechanismen für Managementinterventionen, für das Monitoring und für asynchrone Notifications zu spezifizieren?
- Welche zusätzlichen Dienste werden zur Unterstützung der Kommunikationsmechanismen angeboten?
- Wie werden die Managementmentprotokolle in die Architektur bzw. Protokollhierarchie von FIM/Inter-FIM eingebettet?

Zunächst werden die gerade genannten Fragen generisch beantwortet, bevor SAML-spezifische Aspekte betrachtet werden.

4.5.1. Generische Kommunikationsmechanismen

Verschiedene Nachrichten werden zwischen den Rollen ausgetauscht, um die Funktionen durchzuführen. Das FIM/Inter-FIM-Kommunikationsmodell basiert auf einem Manager-Agent-Ansatz. Die in dem Organisationsmodell und Informationsmodell beschriebenen Rollen nehmen je nach Bedarf die Rolle des Managers oder des Agents an, der über die Interaktionskanäle kommuniziert. Der Fokus der FIM/Inter-FIM-Architektur liegt auf dem Management von Föderationen und Inter-Föderationen. Auch wenn die Architektur prinzipiell protokollunabhängig sein soll, ist die Einbindung in standardisierte SAML-Workflows essentiell, damit Föderationen im R&E-Umfeld die Architektur einsetzen können. SAML basiert hauptsächlich auf HTTP (vgl. [SAML2Bind] [CHK+05]) und verwendet verschiedene Profiles. Folglich werden für das Management von Föderationen die folgenden Protokollinteraktionen mit entsprechenden Parametrisierung verwendet:

post: Um Managementinformationen zu senden.

get: Um Managementinformationen zu lesen.

set: Um Managementinformationen zu setzen.

query: Um Managementinformationen zu suchen.

create: Um Managementinformationen zu erstellen.

delete: Um Managementinformationen zu löschen.

update: Um Managementinformationen zu ändern.

register: Um Managementobjekte zu registrieren.

notify: Um andere Managementinstanzen zu informieren.

discover: Um Managementinstanzen zu identifizieren.

Die *ManagingRole* sendet den Request und erhält vom *Agent* eine Response. Um die Kompatibilität mit SAML und seinen Implementierungen zu wahren, werden die Nachrichten über HTTP verschickt, während längere Informationen in XML strukturiert formuliert sind; einfache Antworten sind im Body der HTTP-Nachricht zu finden. Die genauen Request- und Response-Formulierungen sind abhängig von der jeweiligen Funktion und werden hier nicht weiter diskutiert. Um die im Funktionsmodell noch zu beschreibenden Managementinformationen zu realisieren, müssen Basisdienste im Kommunikationsmodell definiert werden. Dazu zählt die *Authentifizierung und Autorisierung* bei der zentralen Komponente. Hierfür kann das Konzept des FIM angewandt werden. Nachdem Service Provider nicht unbedingt über einen Identity Provider verfügen müssen, ist es zudem wichtig lokale Benutzerkonten zu erlauben.

Ein *Informationsdienst* soll die beteiligten Managementinstanzen über Änderungen informieren können. Diese Informationen sollen entweder über synchrone (`PULL`) oder asynchrone (`PUSH`) Benachrichtigungsmechanismen realisiert werden. Der Inhalt der Nachrichten selbst soll ebenfalls als XML vorliegen.

Die Basisdienste sind allesamt generisch angelegt und verwenden die vorher definierten Kern-Operationen, wie `get` und `set`.

4.5.2. SAML-spezifische Kommunikationsmechanismen

Das FIM/Inter-FIM-Kommunikationsmodell orientiert sich an den Prinzipien des FIM/ Inter-FIM-Organisationsmodell und dessen Einbettung in FIM. Vorhandene Software soll weiter verwendet werden können. Neben dynamischen virtuellen Föderationen sollen auch festere Föderationen gebildet werden können. Das führt zu folgenden Notwendigkeiten:

- Unterstützung von losen wie auch festen Strukturen auf dem Interaktionskanal.
- Protokollierung der durchgeführten Aktionen und der Aktivitäten auf dem Interaktionskanal für Auditierung.
- Publizieren und Finden von Schnittstellen der Interaktionskanäle.
- Durchsetzen von Zugangsbeschränkungen zu Interaktionskanäle.
- Austausch von Metadaten, Vertrauensinformationen und Konvertierungsregeln.

Der letzte Aspekt wurde bereits als allgemeiner Basisdienst beschrieben, während die Protokollierung durch die SAML-Implementierungen gegeben ist und nur bei der zentrale Komponente der Managementarchitektur beachtet werden muss. SAML unterstützt sowohl feste wie auch lose Strukturen, wodurch dieser Aspekt ebenfalls lediglich bei der Implementierung der Managementarchitektur wichtig ist. Damit verschiedene Rollen überhaupt über einen Interaktionskanal kommunizieren können, müssen sie vorab Kenntnis voneinander haben. Eine Grundvoraussetzung hierfür ist die *Registrierung* bei einer Komponente der Managementarchitektur, die als Kernoperation `register` beschrieben ist. Die Registrierung ist Grundlage dafür, den eigenen Dienst zu publizieren und somit anderen Entitäten zugänglich zu machen. Ebenso können sich die weiteren Domänen, d. h. Entitäten bei Föderationen und Föderationen bei Inter-Föderationen, registrieren. Bei verteilten Strukturen ist es unabdingbar, dass sich die Lokalisierungsdienste gegenseitig kennen, also wiederum registriert sind. Abbildung 4.21 zeigt die beispielhafte Registrierung einer Entität bei der Managementplattform, eine TTP, und der anschließenden Registrierung bei einer Föderation.

Nachdem innerhalb der Managementarchitektur durch die Registrierung die unterschiedlichen Managementinstanzen bekannt sind, können diese im *Lokalisierungsdienst* angezeigt werden. Diese Funktionalität ist als Kernoperation `discovery` genannt. Hierfür kann ein

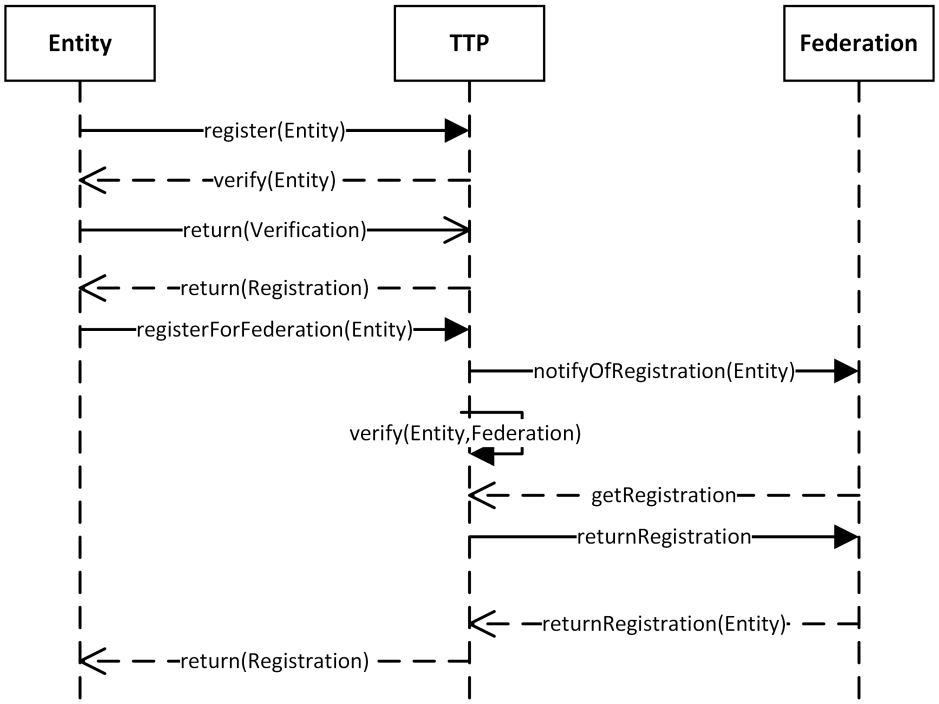

Abbildung 4.21.: Beispiel für die Anwendung von einem Registration Service

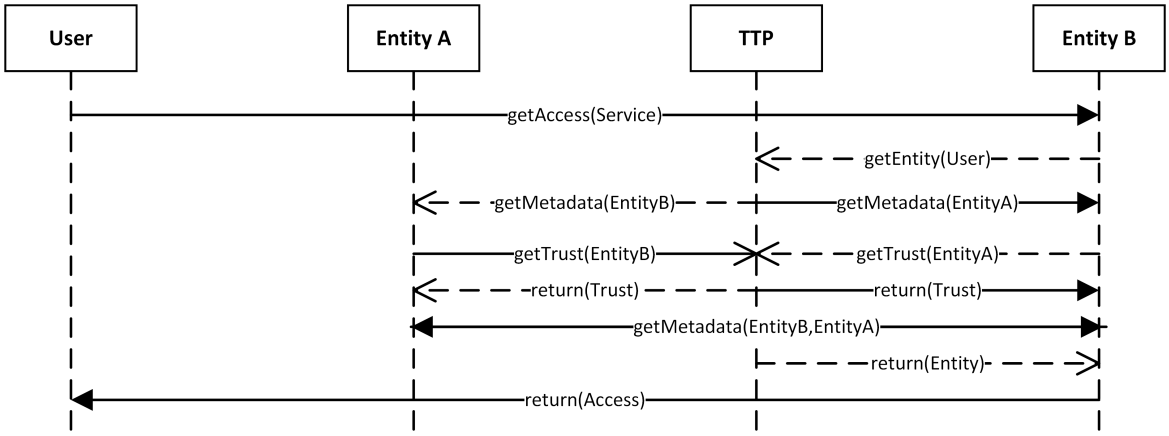

Abbildung 4.22.: Beispiel für die Anwendung von einem Lokalisierungsdienst mit Trust Configuration Service

erweiterter SAML-Discovery Service verwendet werden, der alle relevanten Teilnehmer anzeigt. Das ist wiederum die Voraussetzung für den *Austausch von Metadaten*, wodurch das technische Vertrauen hergestellt wird. Über die Kernoperation `notify` erhält die zentrale Komponente der Managementarchitektur Informationen über den Status des Austausches und der Integration der Metadaten. Falls eine Entität höhere Anforderungen an die Entität am anderen Ende des Interaktionskanals stellt, muss dies über eine Art Trust Configuration

Service eingestellt werden können. Auf die genaue Realisierung wird im folgenden Kapitel eingegangen. Ein möglicher Aspekt hierfür ist der *Austausch von Vertrauensinformationen*, wie auch in der Abbildung 4.22 dargestellt. Mehrere Managementinstanzen, die miteinander interagieren, können eine Föderation bilden, was durch `create` realisiert werden kann. Um die Benutzerinformationen in das passende Format zu transformieren, kann der *Austausch von Konvertierungsregeln* über einen Interaktionskanal relevant sein. Dies wird ebenfalls im nächsten Kapitel näher betrachtet.

In diesem Abschnitt wurden die Grundlagen des FIM/Inter-FIM-Kommunikationsmodells aufgezeigt. Ausgehend von der Spezifikation des Organisations- und Informationsmodells wurden die für die Interaktionskanäle erforderlichen Kern-Operationen identifiziert und darauf aufbauend zusätzliche Basisdienste in einer generischen Form beschrieben. Die Kommunikation zwischen Entitäten, Föderationen und der Trusted Third Party wird im Abschnitt 5.2.2 anhand von Protokollen und einer API genauer spezifiziert. Unter Verwendung der Informations- und Organisationsmodelle kann im nächsten Schritt das Funktionsmodell beschrieben werden.

4.6. Funktionsmodell

Das Funktionsmodell strukturiert gemäß [HAN99] das Management von FIM/Inter-FIM als Ganzes in mehrere Funktionsbereiche und legt allgemeine Managementfunktionen fest. Das Ziel ist dabei generische Funktionsbausteine für Anwendungen des Managements von Föderationen festzulegen. Im Abschnitt 4.1 wurden bereits gewünschte Funktionalitäten beschrieben. In diesem Abschnitt wird für jeden einzelnen Funktionsbereich die erwarteten Funktionalität und Dienste sowie die Managementobjekte zur Erbringung der Funktionalität definiert. Das Funktionsmodell ist ein Teilmodell der FIM/Inter-FIM-Managementarchitektur und adressiert die folgenden Fragen:

- In welche Funktionsbereiche kann das Management von FIM/Inter-FIM aufgeteilt werden?
- Welche generischen Managementfunktionen sind für die einzelnen Funktionsbereiche wichtig?
- Welche Aufrufkonventionen gelten für diese Funktionen?

Diese Fragen werden in den nachfolgenden Abschnitten betrachtet. Zunächst gilt es einzelne Funktionsbereiche festzulegen. Anschließend werden die Managementfunktionen definiert.

4.6.1. Festlegung der Funktionsbereiche

Eine Einteilung in die fünf Funktionsbereiche Fault, Configuration, Accounting, Performance, Security Management (FCAPS), wie beim Open Systems Interconnection (OSI)-Referenzmodell, ist hier nicht möglich, da die Arbeit den Fokus FIM/Inter-FIM besitzt. Äquivalent zum Organisationsmodell und dessen Domänen, werden hier die Funktionalitäten aus den Gesichtspunkten der unterschiedlichen Ebenen betrachtet:

- Die *Entity-Ebene* bestehend aus den Unterebenen AA-Ebene, IdP-Ebene und SP-Ebene.
- Die *Federation-Ebene*, die die Föderation betrifft.
- Die *Inter-Federation-Ebene* bezüglich der Inter-Föderation.

Diese Ebenen gliedern sich folgendermaßen auf:

Entity-Ebene: *Configuration Management*, *Metadata Management*, *Trust Management* und *Conversion Rule Management*. Für die SP-Ebene entfällt Conversion Rule Management. Das lokale Management wird auf Grund der Vorarbeit von Wolfgang Hommel [Hom07] (Abschnitte 4.3 und 4.4) nicht weiter betrachtet.

Federation- und Inter-Federation-Ebene: *Member Management*, *Policy Management*, *Configuration Management* und *Service Management*.

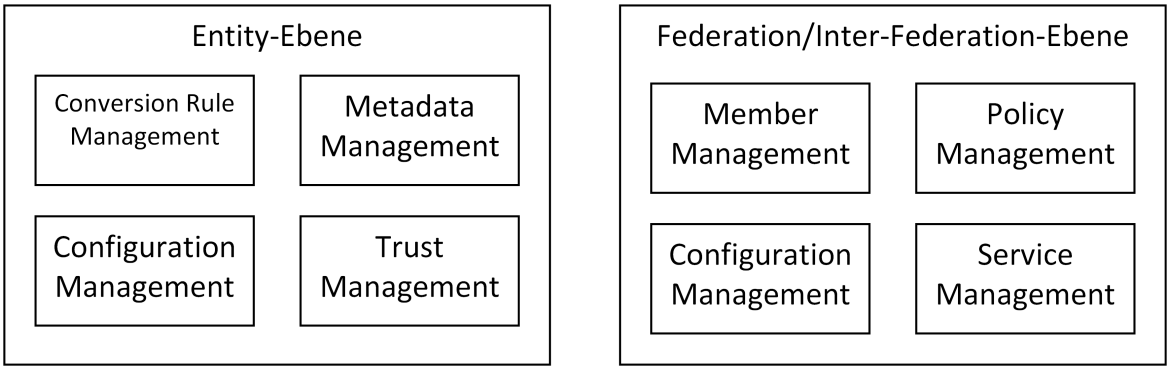

Abbildung 4.23.: Die Funktionsstruktur im FIM/Inter-FIM

Es sei angemerkt, dass hier vor allem die Funktionen betrachtet werden, die für ein effizientes skalierbares FIM/Inter-FIM über eine zentrale Komponente der Managementarchitektur, d. h. einer Managementplattform nötig sind. Es lässt sich jedoch das Modell leicht erweitern und an konkrete Gegebenheiten anpassen. Abbildung 4.23 stellt die Funktionsbereiche übersichtlich zusammen.

4.6.2. Festlegung der Managementfunktionen

In den folgenden Abschnitten werden die in Abbildung 4.23 genannten Funktionsbereiche genauer analysiert. Jeder Funktionsbereich wird zunächst allgemein erläutert, bevor er spezifiziert wird. Gemeinsam bilden sie die in der Managementplattform notwendigen Funktionen und Schnittstellen.

Funktion Configuration Management

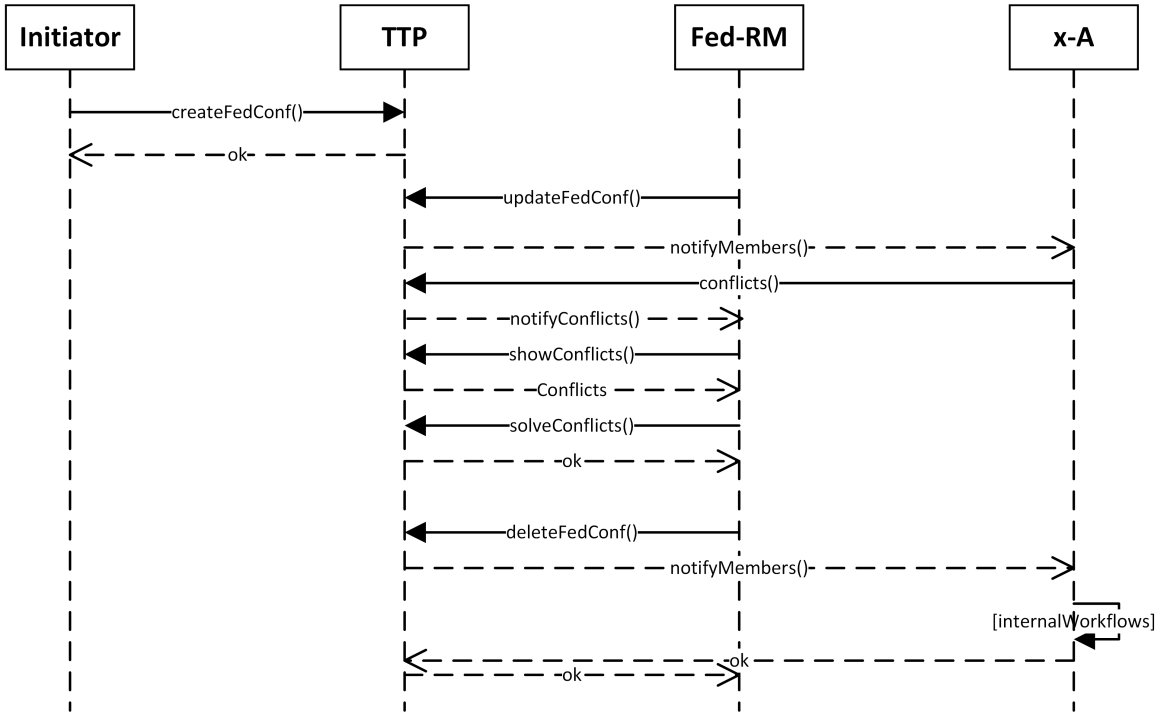

Abbildung 4.24.: Sequenzdiagramm für Configuration Management

Die Funktion Configuration Management behandelt die Konfiguration, also das *Erstellen*, *Ändern*, *Löschen* von Konfigurationen. Die Konfiguration bezieht sich hierbei auf drei Bereiche: Die Föderation und die Inter-Föderation sowie Entitäten. Die Konfiguration von Metadaten, Konvertierungsregeln und Trust bezüglich der Domänen der Entitäten wird in den jeweiligen Funktionen betrachtet.

In der fedDomain können Operatoren der Föderation die Konfiguration der Föderation erstellen, ändern und löschen. Die Konfiguration beruht hierbei auf dem Aufnahmeprozess für Entitäten und die Teilnahme an Inter-Föderationen, wie in Abbildung 4.24 dargestellt:

- Um eine Konfiguration zu erstellen, muss die Funktion `createFedConf()` durch den Initiator aufgerufen werden. Dabei werden die unterschiedlichen Ausprägungen der Fö-

deration angegeben. Zudem muss der Aufnahmeprozess festgelegt werden. Dies beinhaltet die Policies und sonstigen Voraussetzungen, wie Audits und LoA, die Entitäten erfüllen müssen, falls dies laut den Ausprägungen benötigt wird.

- Die eben genannte Konfiguration kann ebenfalls mit dem Aufruf `updateFedConf()` durch den Fed-RM geändert werden, bei dem die zu ändernden Werte mit den neuen Werten übergeben werden. Eine Änderung der Konfiguration muss zudem an die Mitglieder (AA-A, SP-A und IdP-A) über `notifyMembers()` übermittelt werden. Dies kann zu Konflikten der Mitgliedschaft führen, wenn aktuelle Mitglieder den neuen Anforderungen nicht entsprechen. Folglich muss Fed-RM diese Konflikte auflösen.

- Beim Löschen über `deleteFedConf()` durch den Fed-RM wird auch die Föderation aufgelöst. Folglich müssen auch hier die Mitglieder (AA-A, SP-A und IdP-A) informiert werden. Zugleich soll es eine kurze Übergangsphase geben, damit die Mitglieder ggf. eine neue Föderation gründen können. Zudem muss entschieden werden, was mit Konvertierungsregeln der Föderation geschieht. Wenn die Föderation sie nicht löscht, sollen Mitglieder den Besitz übernehmen können.

Entsprechend beschäftigt sich die Konfiguration der Inter-Föderationen mit dem Erstellen, Ändern und Löschen der Konfiguration. Die Konfiguration der Inter-Föderation legt fest, wie Föderationen und Entitäten in einer Inter-Föderation teilnehmen können:

- Äquivalent zu der fedDomain, muss für die interfedDomain eine Konfiguration erstellt werden können (`createIFedConf`). Diese Konfiguration muss die Ausprägungen einer Inter-Föderation widerspiegeln und den Aufnahmeprozess festlegen. Dabei ist entscheidend, ob nur Föderationen Mitglied werden können und damit indirekt die zugehörigen Entitäten oder ob es auch Entitäten (AA-A, SP-A und IdP-A) möglich ist, den Workflow anzustoßen bzw. als einzelne Entität an einer Inter-Föderation teilnehmen können. Zudem sind Policies und sonstige Voraussetzungen zu definieren.

- Die Konfiguration muss ebenfalls geändert werden können (`updateIFedConf()`). Die Mitglieder der Föderation müssen über die Änderung informiert werden über `notifyMembers()`. Falls es zu Konflikten mit Mitgliedschaften kommt, müssen diese Konflikte durch die Administratoren der Föderation gelöst werden.

- Das Löschen und somit Auflösen der Inter-Föderation ist ähnlich dem Auflösen der Föderation. Die Mitglieder, insbesondere Fed-RM, müssen nach dem Aufruf der Funktion `deleteIFedConf()` durch den IFed-RM informiert werden. Durch eine Übergangsphase haben die Mitglieder die Möglichkeit zu reagieren.

Funktion Metadata Management

Die Funktion Metadata Management umschließt das *Hochladen*, *Ändern*, *Löschen*, *Austauschen*, *Konfigurations des Automatisierungsgrades* von Metadaten sowie *Notifikationen* und

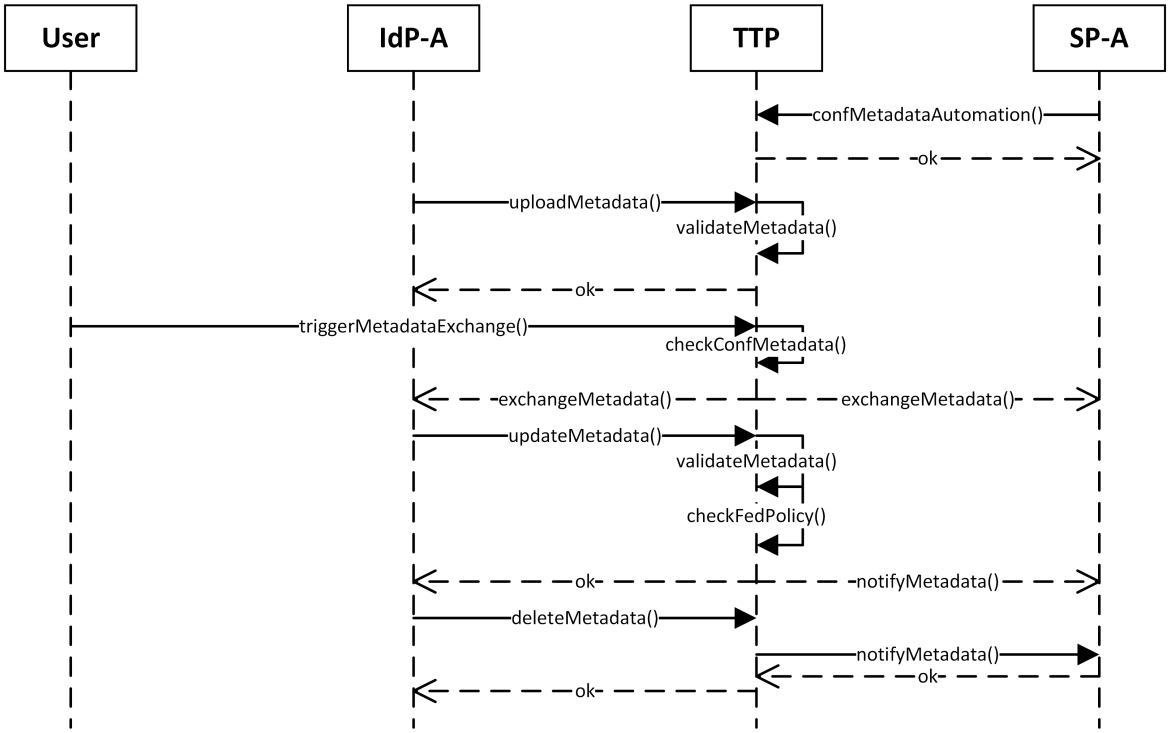

Abbildung 4.25.: Sequenzdiagramm für Metadata Management

Protokollierung. Diese Funktion betrachtet somit die Domänen der Entitäten und ist in Abbildung 4.25 visualisiert.

Metadaten hochladen: Der Administrator der Entität, d. h. AA-A, SP-A oder IdP-A, lädt die Metadaten seiner Entität über `uploadMetadata()` hoch, nachdem seine Entität akzeptiert wurde. Dies geschieht im Rahmen der Registrierung. Alternativ kann er auch einen Ort angeben, wo seine Metadaten öffentlich verfügbar sind (`setMetadata-Location()`). Dabei werden die Metadaten validiert (`validateMetadata()`). Wenn sie nicht korrekt sind, beispielsweise wenn das XML falsch geformt ist, werden die Metadaten nicht akzeptiert. Der Administrator muss den Fehler beseitigen und dann seine neuen Metadaten hochladen. Diese Status sollen gleichzeitig protokolliert werden.

Metadaten ändern: Bei Änderungen von Metadaten lädt der Administrator der Entitäten (AA-A, SP-A oder IdP-A) einen neuen Metadatensatz zur TTP hoch oder er ersetzt die Metadaten beim öffentlich verfügbaren Ort außerhalb der TTP. Äquivalent zum ersten Hochladen der Metadaten müssen die Metadaten durch `validateMetadata()` validiert werden. Falls Föderationen Anforderungen bezüglich der Metadaten haben, müssen die Metadaten diesbezüglich ebenfalls überprüft werden (`checkFedPolicy()`). Nach der Änderung der Metadaten müssen alle Entitäten benachrichtigt werden, die technische Vertrauensbeziehungen haben (`notifyMetadata()`). Folglich sollen diese Vertrauensbeziehungen in der TTP gespeichert werden.

Metadaten löschen: Beim Löschen von Metadaten über `deleteMetadata()` werden die Metadaten deaktiviert. Als Resultat müssen hier ebenfalls die Entitäten mit Vertrauensbeziehungen benachrichtigt werden (`notifyMetadata()`). Durch das Löschen wird die technische Vertrauensbeziehung nach einer kurzen Übergangsphase aufgelöst (`deleteTrust()`), wenn keine neuen Metadaten hochgeladen werden.

Metadaten austauschen: Der Aufbau des technischen Vertrauens beinhaltet den Austausch von Metadaten (`exchangeMetadata()`). Zwei Entitäten tauschen die Metadaten über die TTP aus, um anschließend dem Nutzer einen Service zur Verfügung zu stellen. Falls eine oder beide Entitäten bestimmte Voraussetzungen erfüllt haben möchten, müssen diese zunächst überprüft werden. Diese Voraussetzungen können beispielsweise Trust, was im nächsten Kapitel erklärt wird, oder bestimmte Kategorien sein. Beispielsweise kann ein IdP bestimmen, dass er nur SPs aus R&E akzeptiert. Möchte ein Nutzer einen kommerziellen SP nutzen, wird dieses technische Vertrauen nicht hergestellt. Der Nutzer muss hierüber informiert werden (`notifyUser()`).

Konfiguration des Automatisierungsgrades: Dieses Beispiel kann durch die Konfiguration des Automatisierungsgrades weiter geführt werden. Wenn der IdP beispielsweise konfiguriert hat, dass alle Service Provider, die nicht aus R&E stammen, vorher durch IdP-A manuell akzeptiert werden müssen, erhält der IdP-A nun eine Nachricht und kann dann entscheiden, ob er den SP akzeptiert oder ihn ablehnt. Folglich muss der Automatisierungsgrad konfiguriert werden können (`confMAutomation()`). Diese Konfiguration kann auch geändert und gelöscht werden. Beim Metadatenaustausch wird diese Konfiguration überprüft (`checkConfMetadata()`) und gegebenenfalls wird der Administrator informiert (`notifyMetadataConf()`).

Notifikationen: Die Auflistung zeigt bereits, dass es verschiedene Notifikationen gibt. Administratoren erhalten Notifikationen, wenn sich Metadaten geändert haben oder sie gelöscht werden (`notifyMetadata()`). Wenn ein Metadatenaustausch durch die Konfiguration des Automatisierungsgrades das manuelle Eingreifen des Administrators erfordert, wird er über eine weitere Notifikation darüber informiert (`notifyMetadataConf()`). Falls ein Fehler beim Metadatenaustausch auftrat, ist es ebenfalls wichtig, die Administratoren zu informieren (`notifyMetadataError()`). Neben den Administratoren erhält auch der Nutzer Informationen über `notifyUser()`, wenn beispielsweise das technische Vertrauen nicht hergestellt werden konnte. Hierbei ist es essentiell, dass der Nutzer eine verständliche Fehlermeldung erhält.

Protokollierung: Alle relevanten Aktionen bezüglich des Metadatenaustausches und dem Hochladen von Metadaten sollen protokolliert werden. Dies beinhaltet den Status der Metadaten (`setMetadataStatus()`), dem Aufbau vom technischen Vertrauen mit dem Status der Vertrauensbeziehung (`setTrustStatus()`) und den Gründen für Fehler beim Aufbau der Vertrauensbeziehung (`setTrustError()`).

Funktion Trust Management

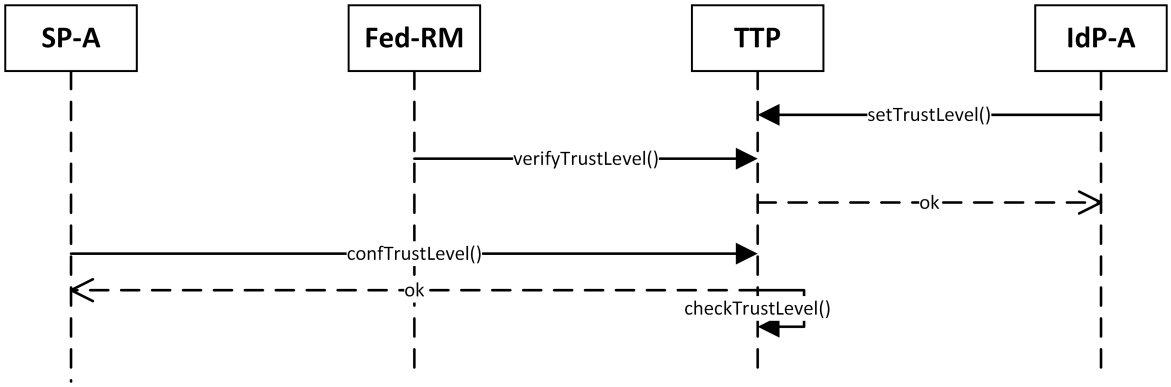

Abbildung 4.26.: Sequenzdiagramm für Trust Management

Neben dem technischen Vertrauen werden hier weitere Aspekte des Vertrauens angesprochen. LoA und LoT werden im nächsten Kapitel genauer näher definiert.
Eine Entität (AA-A, SP-A oder IdP-A) erhält einen Vertrauensgrad, beispielsweise einen LoA (`setTrustLevel()`). Gegebenenfalls muss dieser Vertrauensgrad durch eine Föderation oder über ein Audit bestätigt werden (`verifyTrustLevel()`). Falls eine Entität nur einen oder mehrere bestimmte Vertrauensgrade akzeptiert (`confTrustLevel()`), muss der Vertrauensgrad mit dieser Konfiguration abgeglichen werden (`checkTrustLevel()`). Dies ist in Abbildung 4.26 abgebildet. Wenn der Vertrauensgrad nicht ausreicht, müssen sowohl Administrator (`notifyTrustError()`) als auch Nutzer (`notifyUser()`) informiert werden. Im einfachsten Fall erstellt der Administrator (AA-A, SP-A oder IdP-A) eine Art Blacklist oder Whitelist (`confListMetadata()`), die für den Abgleich verwendet wird (`checkListMetadata()`). Diese Liste kann genauso wie der Vertrauensgrad geändert oder gelöscht werden.

Funktion Conversion Rule Management

Damit Benutzerinformationen in das Format des SPs konvertiert werden, müssen Konvertierungsregeln eingesetzt werden. Konvertierungsregeln werden im nächsten Kapitel detaillierter betrachtet. Für dieses Kapitel ist es wichtig zu sehen, dass Konvertierungsregeln

- erstellt (`createConvRule()`),
- geändert (`changeConvRule()`),
- gelöscht (`deleteConvRule()`) und
- ausgetauscht (`downloadConvRule()`) werden können.

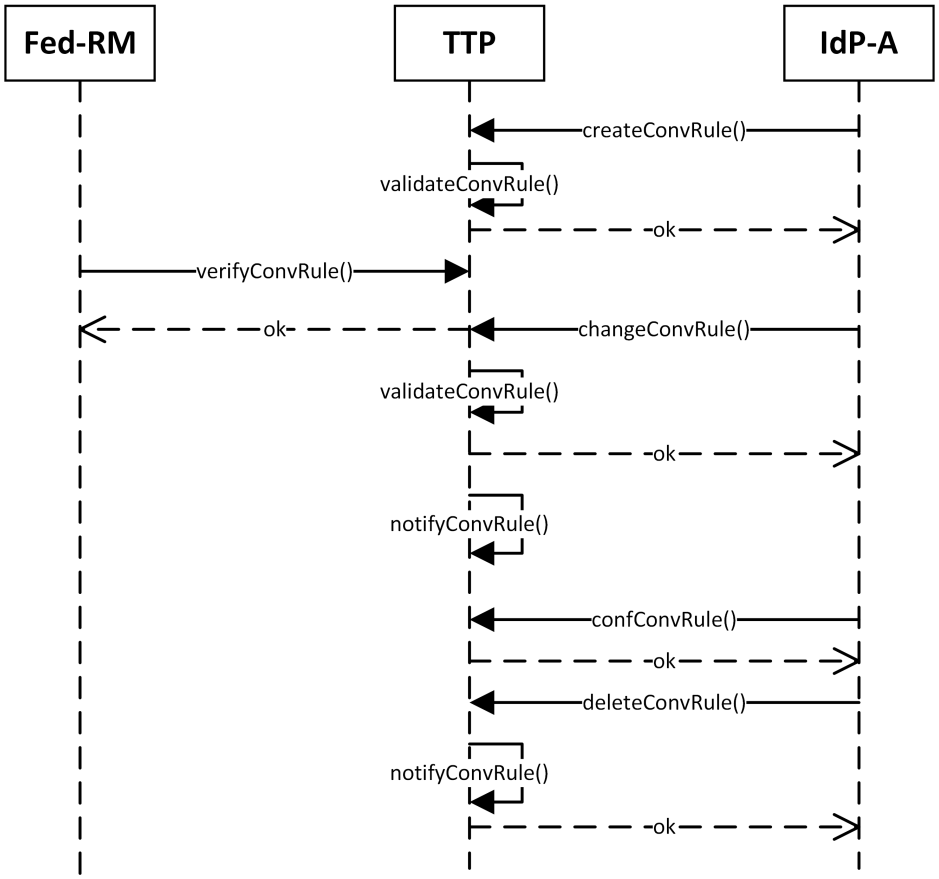

Abbildung 4.27.: Sequenzdiagramm für Conversion Rule Management

Hierbei werden die Konvertierungsregeln validiert (`validateConvRule()`). Der Grad der Automatisierung kann konfiguriert werden (`confConvRule()`). Entsprechend der Implementierung müssen Anpassungen an die Regel gemacht werden. Bei Änderungen oder falls ein geringer Grad an Automatisierung gewünscht ist, muss der Administrator (IdP-A oder AA-A) informiert werden (`notifyConvRule()`). Föderationen können Konvertierungsregeln akzeptieren (`verifyConvRule()`) und somit ihnen eine höhere Priorität geben. Falls der Eigentümer sich löscht, soll es eine Möglichkeit geben die Eigentümerschaft zu übernehmen (`setOwnershipConvRule()`). Äquivalent zu den vorherigen Funktionen müssen alle wichtigen Aktionen protokolliert werden. Ein beispielhaftes Sequenzdiagramm in Abbildung 4.27 veranschaulicht diese Aufrufe. Falls keine Konvertierungsregel verfügbar ist, müssen sowohl Nutzer (`notifyUser()`) als auch Administrator (`notifyConvRule()`) informiert werden.

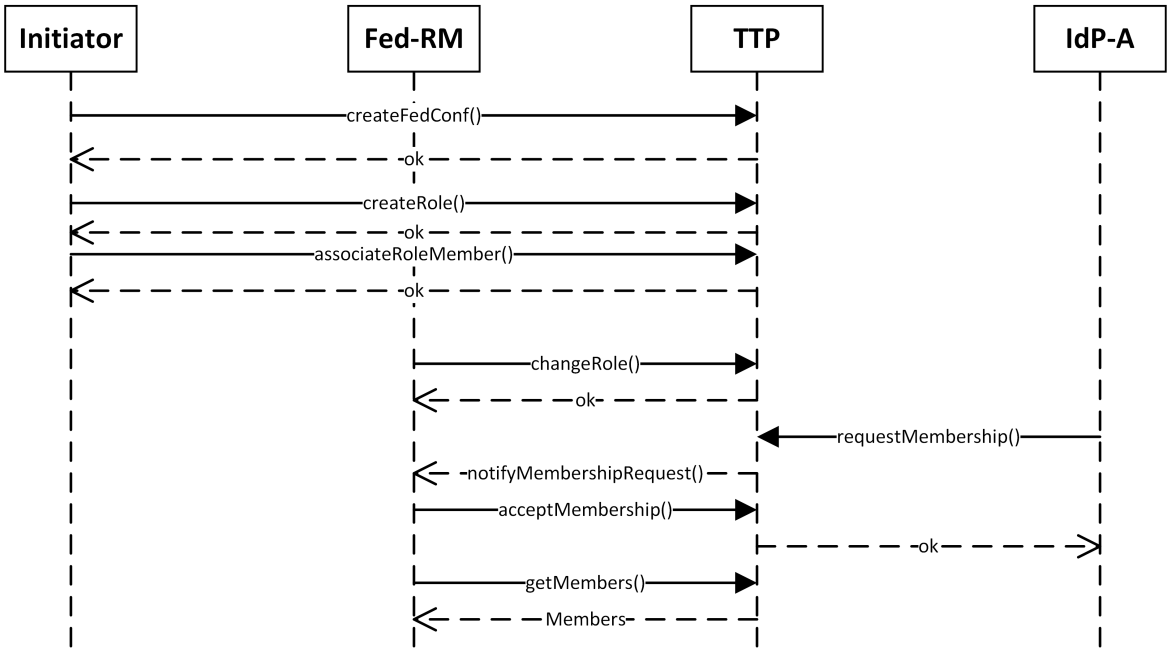

Abbildung 4.28.: Sequenzdiagramm für Member Management

Funktion Member Management

Damit feste Föderationen ihre Mitglieder verwalten können, sind verschiedene Funktionalitäten möglich:

- Abfrage von Mitglieds- und Rolleninformationen,
- Antrag auf Mitgliedschaft, Mitglieder akzeptieren und ablehnen, sowie Mitglieder aus der Föderation entlassen,
- Eigene und andere Mitgliedschaften ändern und löschen,
- Rollen verwalten, inklusive Erstellen, Ändern und Löschen von Rollen sowie dem Assoziieren von Rollen und Mitgliedern,
- Notifikationen bei Änderungen,
- Aufnahmeprozess erstellen, ändern und löschen sowie
- Protokollierung von wichtigen Aktionen.

Eine Auswahl ist in Abbildung 4.28 dargestellt.

Die Abfrage von Mitglieds- und Rolleninformationen verläuft über einfache Getter:

- `getMember()` und
- `getRoles()`.

Die Mitgliedschaft in einer Föderation bzw. Inter-Föderation kann durch einen Administrator einer Entität (AA-A, SP-A, IdP-A) oder einem Fed-RM gestellt werden (`requestMembership()`). Entsprechend werden durch den Fed-RM bzw. IFed-RM Mitglieder akzeptiert (`acceptMembership()`) oder abgelehnt (`denyMembership()`), nachdem sie eine Notifikation erhalten haben (`notifyMembershipRequest()`). Die Mitgliedschaft wird in der Datenbank der TTP gespeichert, ebenso wie die Status dazwischen. Die eigene Mitgliedschaft kann durch `changeMembership()` geändert bzw. durch `deleteMembership()` gelöscht werden. Diese Aufrufe können durch Fed-RM und IFed-RM für ihre Mitglieder ebenfalls ausgeführt werden, wodurch die Mitglieder informiert werden müssen (`notifyMembershipChange()`). Wenn ein Administrator die Mitgliedschaft in einer Föderation bzw. ein Fed-RM die Mitgliedschaft in einer Inter-Föderation ändert, muss der zuständige RM ebenfalls darüber informiert werden (`notifyMembershipChange()`).

Neue Rollen können durch `createRole()` erstellt werden. Änderungen (`changeRole()`) und Löschen (`deleteRole()`) resultieren in Notifikationen an Mitgliedern, die diese Rollen ausfüllen (`notifyRoleChange()`). Wenn Mitglieder Rollen ausfüllen, müssen Rollen und Mitglieder miteinander assoziiert werden (`associateRoleMember()`). Dasselbe gilt für die eigene Verwaltung, wenn zusätzliche Rollen benötigt werden.

Damit AA-A, SP-A und IdP-A Mitglied in einer Föderation werden, muss vorab der Aufnahmeprozess erstellt werden. Dies wurde bereits in der Funktion Configuration Management durch den Aufruf `createFedConf()` beschrieben. Die Konfiguration kann entsprechend geändert und gelöscht werden.

Wie gerade gesehen, sind verschiedene Notifikationen notwendig:

- `notifyMembershipRequest()`,
- `notifyMembershipChange()` sowie
- die Notifikation `notifyMembers()`, wenn der Aufnahmeprozess bei einer Föderation bzw. Inter-Föderation geändert wird.

Ebenso werden alle wichtigen Aktionen protokolliert und in der Datenbank der TTP gespeichert.

Dynamische virtuelle Föderationen müssen durch Mitglieder auch gegründet werden. In diesem Fall kreiert ein Mitglied, (IdP-RM oder SP-RM), eine Föderation über `createFed()`. Dabei muss die Konfiguration beschrieben werden (`createFedConf()`). Diese enthält ver-

mutlich keine oder nur geringe Anforderungen und auch keinen speziellen Aufnahmeprozess. Dies kann auch automatisch nach Überschreiten von vermehrten Kooperationen geschehen. Über `requestMembership()` kann eine SPDomain, IdPDomain oder AADomain eine Mitgliedschaft anfordern. Nachdem keine Anforderungen bestehen, wird die Mitgliedschaft kurz darauf automatisch akzeptiert (`acceptMembership()`). In diesem Fall ist die Erstellung von Rollen umso wichtiger, da Mitglieder nun RM, A und weitere Rollen übernehmen können. Ein voll automatisierter Mechanismus ist ebenfalls denkbar.

Funktion Policy Management

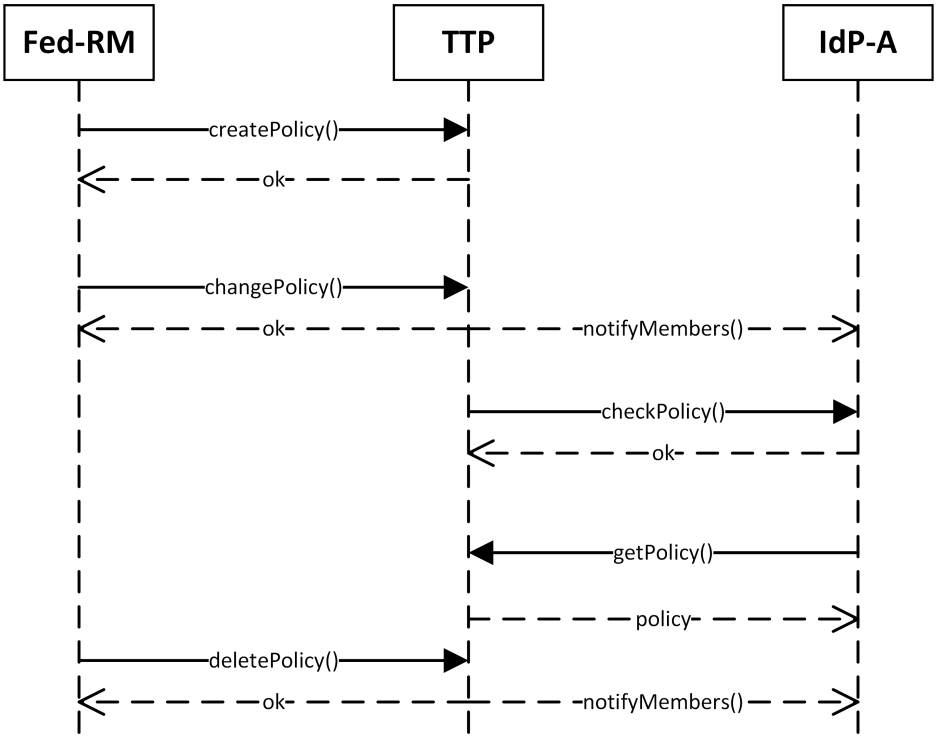

Abbildung 4.29.: Sequenzdiagramm für Policy Management

Ein essentieller Part des Aufnahmeprozesses ist der Abgleich mit Policies. Die Policies sollen möglichst in einem generischen Format gespeichert werden, um einen automatischen Abgleich zu erlauben. Zum initialen Speichern müssen die Policies durch Fed-RM bzw. IFed-RM hochgeladen werden (`createPolicy()`). Bei Änderungen (`changePolicy()`) muss die geänderte Policy gegenüber den bestehenden Mitgliedschaften abgeglichen und Konflikte aufgelöst werden. Zugleich müssen die Mitglieder kontaktiert werden (`notifyMembers()`). Dies ist auch beim Löschen einer Policy (`deletePolicy()`) relevant. Über einen Getter können Policies, beispielsweise durch SP-A, IdP-A und AA-A, abgerufen werden (`getPolicy()`). Ein Abgleich mit einer Entität geschieht über `checkPolicy()` und ist Teil der Überprüfung einer Entität beim Beitritt zu einer Föderation. Dieselbe Funktion kann beim Abgleich einer Föde-

ration gegenüber einer Inter-Föderation eingesetzt werden. Ein Sequenzdiagramm zu einem beispielhaften Lebenszyklus im Policy Management ist in der Abbildung 4.29 dargestellt.

Funktion Service Management

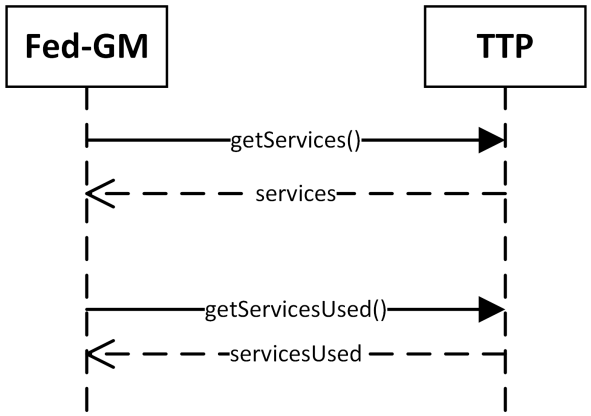

Abbildung 4.30.: Sequenzdiagramm für Service Management

Funktionen sowie Inter-Föderationen benötigten für Fed-GM bzw. IFed-GM eine Übersicht über die angebotenen Dienste und deren Nutzung. Diese Übersicht kann letztlich durch den Nutzer verwendet werden, um Dienste zu nutzen. Die Abfrage über die Dienste in einer Föderation bzw. Inter-Föderation ist über `getServices()` für die entsprechenden Rollen möglich. Über `getServicesUsed()` können zusätzliche Informationen über die Nutzung abgerufen werden. Dies ist im Sequenzdiagramm der Abbildung 4.30 dargestellt. Weitere Abfragen für Statistiken sind denkbar und leicht erweiterbar.

Zusammenfassung

In den vorangegangenen Abschnitten wurden die Funktionen Configuration Management, Metadata Management, Trust Management, Conversion Rule Management, Member Management, Policy Management und Service Management besprochen. Jeder dieser Funktionsbereiche besteht aus mehreren einzelnen Funktionen, die teils wiederverwendet werden. Die Tabellen 4.36 und 4.37 geben einen Überblick über alle genannten Funktionen und ihre Verwendung.

Diese Funktionen bilden die wichtigsten Funktionen für FIM/Inter-FIM über eine Managementplattform. Die Anzahl an einzelnen Funktionen zeigt, dass für ein Konzept für dynamischen Metadatenaustausch über eine TTP viele Funktionen und Funktionsbereiche beachtet werden müssen. Über die noch zu definierende API und den ebenfalls noch zu definierenden Protokollen können die unterschiedlichen Funktionen aufgerufen werden. Dies

Funktion	Configuration Management	Metadata Management	Trust Management	Conversion Rule Management	Member Management	Policy Management	Service Management
createFedConf()	+	-	-	-	-	-	-
updateFedConf()	+	-	-	-	-	-	-
notifyMembers()	+	-	-	-	-	+	-
deleteFedConf()	+	-	-	-	-	-	-
createIFedConf()	+	-	-	-	-	-	-
updateIFedConf()	+	-	-	-	-	-	-
deleteIFedConf()	+	-	-	-	-	-	-
uploadMetadata()	-	+	-	-	-	-	-
setMetadataLoc()	-	+	-	-	-	-	-
validateMetadata()	-	+	-	-	-	-	-
checkFedMetadata()	-	+	-	-	-	-	-
notifyMetadata()	-	+	-	-	-	-	-
deleteMetadata()	-	+	-	-	-	-	-
deleteTTrust()	-	+	-	-	-	-	-
exchangeMetadata()	-	+	-	-	-	-	-
notifyUser()	-	+	+	+	-	-	-
confMAutomation()	-	+	-	-	-	-	-
checkConfMetadata()	-	+	-	-	-	-	-
notifyMetadataConf()	-	+	-	-	-	-	-
notifyMetadataError()	-	+	-	-	-	-	-
setMetadataStatus()	-	+	-	-	-	-	-
setTrustStatus()	-	+	-	-	-	-	-
setTrustError()	-	+	-	-	-	-	-
setTrustLevel()	-	-	+	-	-	-	-
confTrustLevel()	-	-	+	-	-	-	-
checkTrustLevel()	-	-	+	-	-	-	-
confListMetadata()	-	-	+	-	-	-	-
checkListMetadata()	-	-	+	-	-	-	-
notifyTrustError()	-	-	+	-	-	-	-

Tabelle 4.36.: Überblick über die Funktionen und Funktionsbereiche im FIM/Inter-FIM über eine TTP 1/2

Funktion	Configuration Management	Metadata Management	Trust Management	Conversion Rule Management	Member Management	Policy Management	Service Management
createConvRule()	-	-	-	+	-	-	-
changeConvRule()	-	-	-	+	-	-	-
deleteConvRule()	-	-	-	+	-	-	-
downloadConvRule()	-	-	-	+	-	-	-
validateConvRule()	-	-	-	+	-	-	-
confConvRule()	-	-	-	+	-	-	-
notifyConvRule()	-	-	-	+	-	-	-
verifyConvRule()	-	-	-	+	-	-	-
setOwnershipConvRule()	-	-	-	+	-	-	-
getMember()	-	-	-	-	+	-	-
getRoles()	-	-	-	-	+	-	-
requestMembership()	-	-	-	-	+	-	-
acceptMembership()	-	-	-	-	+	-	-
denyMembership()	-	-	-	-	+	-	-
notifyMembershipRequest()	-	-	-	-	+	-	-
deleteMembership()	-	-	-	-	+	-	-
notifyMembershipChange()	-	-	-	-	+	-	-
createRole()	-	-	-	-	+	-	-
changeRole()	-	-	-	-	+	-	-
deleteRole()	-	-	-	-	+	-	-
notifyRoleChange()	-	-	-	-	+	-	-
associateRoleMember()	-	-	-	-	+	-	-
createPolicy()	-	-	-	-	-	+	-
changePolicy()	-	-	-	-	-	+	-
deletePolicy()	-	-	-	-	-	+	-
getPolicy()	-	-	-	-	-	+	-
checkPolicy()	-	-	-	-	-	+	-
getServicesUsed()	-	-	-	-	-	-	+
getServices()	-	-	-	-	-	-	+

Tabelle 4.37.: Überblick über die Funktionen und Funktionsbereiche im FIM/Inter-FIM über eine TTP 2/2

wird im folgenden Kapitel näher spezifiziert. Basierend auf der Prozess-Klassifikation von Markus Garschhammer et al. [GHH+02] erfolgt ein kurzer Abgleich der in dieser Arbeit zu konzipierende Lösung mit der Abdeckung möglicher Aspekte:

Contract Management: Die losen Verträge zwischen IdP und SP werden in der Datenbank anhand der im nächsten Kapitel beschriebenen Workflows gespeichert. Diese Verträge kommen durch den dynamischen Metadatenaustausch zustande. Tatsächliche, schriftliche SLAs können durch die TTP nicht abgeschlossen, jedoch initiiert werden.

Provisioning: Rechtemanagement anhand von Rollen, die in der Datenbank gespeichert sind. Vergleiche hierbei die in diesem Kapitel spezifizierten Rollen in den vorhandenen Domänen und den passenden Funktionen.

Accounting Management: Die bereits lokal implementierten Lösungen werden durch die Managementplattform nicht geändert. Folglich erfolgt lokal bereits die Protokollierung. Die Managementplattform muss diese Funktionalität nachbilden, die auch für Statistiken weiter verwendet werden kann.

Problem Management: Die lokalen Lösungen werden durch die Managementplattform nicht geändert, jedoch ist ein übergeordnetes Problem Management eine sinnvolle Erweiterung. In diesem Abschnitt wurde dies bereits durch die Notifikationen an die Nutzer und Administratoren bzw. Resource Manager dargestellt.

Security Management: Die Auswirkungen auf das Security Management werden in Abschnitt 4.8 näher erläutert. Ferner ist eine interorganisationale Lösung wünschenswert.

Customer Care: Die lokal implementierten Lösungen werden durch die Managementplattform nicht geändert. Über die Föderationsverwaltung der Managementplattform können Föderationsverwaltungen ihre Mitglieder verwalten.

Change Management: Die Auswirkungen auf das Change Management werden nachfolgend in Abschnitt 4.8 beschrieben.

4.7. Integration in bestehende Umgebung

Nach der Betrachtung von Organisationsmodell, Funktionsmodell, Kommunikationsmodell und Informationsmodell wird basierend auf diesen Informationen im folgenden Abschnitt die Integration der entwickelten Architektur in die bestehende Umgebung gezeigt. Dieser Schritt dient zugleich zur Vorbereitung der Migration. Bei der Integration werden nicht nur die technischen Komponenten betrachtet, sondern auch die Anpassung interner Abläufe. Da bestehende Software erweitert wird, bleiben interne technische Schnittstellen großteils bestehen. Anschließend werden Security Management und Change Management genauer betrachtet.

4.7.1. Integration für Entitäten

Neben der *Installation der Erweiterung* und der grundlegenden *Konfiguration* sind folgende Aspekte für alle Entitäten relevant:

- Die Entität muss festlegen, welche *Anforderungen* an andere Entitäten gestellt werden. Dies kann beispielsweise Datenschutz, aber auch andere Aspekte des Vertrauens betreffen. Dazu müssen die Verantwortlichen den jeweiligen Schutzbedarf festlegen. Dieser bezieht sich auf das Risiko, welches durch falsche Daten bzw. Verwendung entsteht. Das Risiko lässt sich durch Eintrittswahrscheinlichkeit und Auswirkung abschätzen. So ist ein Terminplaner mit weniger Risiken behaftet als ein kommerzieller Dienst, für den eine Minuten-genaue Abrechnung erfolgt und der sensible Daten verarbeitet.

- Wenn eine Entität bestimmte andere Entitäten kennt, mit denen sie keine Metadaten austauschen möchte, da sie ihnen nicht vertraut, dann muss sie diese in einer *Blacklist* eintragen. Wenn dies für eine bestimmte Kategorie an Entitäten gilt, ist dies ebenfalls festzulegen. Sind hingegen andere Entitäten sicher, können diese in *Whitelists* stehen. Diese Konfiguration ist auch über verschiedene *Vertrauenswerte*, wie LoA und LoT, möglich.

- Nach der Festlegung der eben genannten Punkte, müssen diese in der lokalen *Konfiguration* beschrieben werden, damit sie angewandt werden können.

- Der Grad der *Automatisierung* ist ebenfalls in der Konfiguration festzulegen, d. h. inwiefern die Abläufe automatisiert werden sollen. Dies kann beispielsweise vollkommene Automatisierung sein, wenn das Risiko gering ist, beispielsweise bei der ausschließlichen Verwendung von Whitelists oder bei externen IdPs, wie Google. Ist das Risiko höher, kann beispielsweise der Administrator informiert werden. Stimmt er dem Metadatenaustausch zu, erfolgt dieser asynchron.

- Der Administrator muss seinen Dienst bei der Managementplattform *anmelden* und seine Berechtigung nachweisen. Zudem muss die entsprechende Software mit der oben beschriebenen Erweiterung betrieben werden. Dazu sind entsprechende Konfigurationsschritte notwendig. Änderungen müssen ebenfalls der TTP mitgeteilt werden.

- Um bei Föderationen teilzunehmen, muss die *Mitgliedschaft* bei der entsprechenden Föderation beantragt werden. Die Policies sind öffentlich bekannt und können daher vor der Beantragung eingesehen werden. Die Föderationsverwaltung überprüft die Vorgaben und entscheidet über den Beitritt.

- Wenn die Entität außerhalb der Föderationen Gruppen bilden will, kann der Administrator bei Bedarf virtuelle Föderationen über die Managementplattform gründen bzw. automatisiert gründen lassen.

Für Identity Provider ist speziell Folgendes zu beachten:

- Die Automatisierung soll sich auch für Konvertierungsregeln einstellen lassen. Die Anpassungen bezüglich Attribute werden im nächsten Kapitel besprochen.

Der folgende Aspekt ist für Service Provider relevant:

- Wenn der SP einen *Embedded Discovery Service* verwendet, muss dieser für die Nutzung mit TTP angepasst werden.

4.7.2. Integration für Föderationen und Inter-Föderationen

Durch den Einsatz einer TTP als Managementplattform können fedDomain und interfedDomain ihre Föderation bzw. Inter-Föderation über die Managementplattform verwalten. Für die Föderationsverwaltung ändert sich durch die Schnittstelle bei der Managementplattform Folgendes:

- Wenn die Operateure, insbesondere Fed-A, der Föderation eine eigene TTP administrieren wollen, muss die TTP zunächst *installiert* und konfiguriert werden.

- Nach der Entscheidung, welche TTP verwendet werden soll, muss sich die Föderation bei der Managementplattform *registrieren*. Dies geschieht durch den Initiator.

- Anschließend an die Autorisierung müssen der *Aufnahmeprozess und die Policies* festgelegt und eingetragen werden. Dies ist notwendig, damit es einen einheitlichen Prozess für IdPs und SPs gibt. Neben Policies können u. a. auch Audits vorgeschrieben werden. Diese Konfiguration wird ebenfalls durch den Initiator vorgenommen.

- Die *Verwaltung* von Entitäten geschieht über das Web-Interface der Managementplattform. Dazu zählt die Aufnahme und Ablehnung von Entitäten, aber auch die Aktualisierung von Policies sowie der Ausschluss von Entitäten.

- Um das Aufsetzen einer neuen TTP zu erleichtern, soll der *Import* bestehender Mitgliedschaften über Föderationsmetadaten möglich sein. Das bedeutet, dass die Entitäten automatisch registriert und in die Föderation aufgenommen werden.

- Nachdem die Föderationsverwaltung konfiguriert und angepasst ist, kann die Föderation ggf. die Teilnahme an einer *Inter-Föderation* beantragen.

Folglich wird die Verwaltung der Föderation nicht mehr rein lokal, sondern über die Schnittstelle der Managementplattform geschehen.

4.8. Schnittstellen

Im Folgenden werden die Schnittstellen zu *Security Management* und *Change Management* betrachtet. Zunächst wird die Sicherheitsinfrastruktur erläutert, um mögliche Angriffe und Maßnahmen zu verstehen. Basierend auf der Betrachtung der Komponenten werden verschiedene Angreifermodelle vorgestellt, um anschließend die wichtigsten Schutzmaßnahmen zusammen zu fassen. Dabei wird davon ausgegangen, dass ein LoA bzw. LoT verwendet werden kann, wie im nächsten Kapitel näher spezifiziert. Die Erweiterung der bisherigen Software hat verschiedene Auswirkungen auf das Change Management, die ebenfalls analysiert werden.

4.8.1. Sicherheitsinfrastruktur und Security Management

In diesem Abschnitt werden die Sicherheitsmaßnahmen und Angriffsszenarien, die zusätzlich zu den in Wolfgang Hommels Dissertation [Hom07] beschriebenen existieren, diskutiert. Das bedeutet, dass die Maßnahmen und Angriffe durch die Erweiterung ermöglicht werden, wie in [Pöh16b] aufgezeigt. Die nachfolgende Betrachtung erfolgt nach dem Riskomanagement-bezogenen Ansatz. Die Grundlage hierfür bilden vorhandene *Assets*, das Schutzbedürfnis und die Sicherheitsbetrachtung der einzelnen Komponenten. Die Sicherheitseigenschaften von SAML [`SAMLSecure`] [HPM05] wird an dieser Stelle nicht betrachtet, da dieser Aspekt in anderen Arbeiten bereits betrachtet wurde. Basierend darauf erfolgt die Beschreibung untergliedert in die verschiedenen *Maßnahmen* und *Angreifermodelle*.

Laut [HAN99] umfasst Security Management neben der Risikoanalyse die folgenden Aspekte:

- Security Policy,
- Authentisierung,
- Zugriffskontrolle,
- Verschlüsselung,
- Datenintegrität,
- Monitoring und
- Reporting.

Wichtig im Zusammenhang mit der beschriebenen Architektur ist die Einhaltung der Vertraulichkeit, der Integrität und der Verfügbarkeit. Um diese zu gewährleisten, muss bekannt sein, wie hoch das *Risiko* ist. Das Risiko wird durch die *Eintrittswahrscheinlichkeit* und

die *Auswirkung* ermittelt. Ist mindestens einer der beiden Faktoren hoch, sollen Gegenmaßnahmen getroffen werden. Bei geringer Eintrittswahrscheinlichkeit und geringer Auswirkung kann gegebenenfalls das Risiko vernachlässigt werden.

Assets und Schutzbedürfnis

Im Risikomanagement-orientiertem Ansatz werden zunächst Assets und der Schutzbedarf ermittelt, bevor Schwachstellen aufgezeigt werden. Basierend auf Angriffen werden Maßnahmen erläutert und die Kosten betrachtet. Die Angriffe werden nachfolgend aufgeteilt in die verschiedenen Angreifertypen und Angriffsgründe näher analysiert. Assets sind beispielsweise

- Benutzerinformationen, die zwischen Identity Provider und Service Provider ausgetauscht werden und beim IdP gespeichert sind.
- Dienste mit Dienstinformationen, also SPs und die bei SPs vorhandenen Daten. Ebenso sind Identity Provider hierbei ein Asset.
- TTP mit Software und Informationen, die ebenfalls ein Dienst mit Dienstinformationen ist.
- der Dienst, durch den das technische Vertrauen ausgetauscht wird.
- Vertrauen zwischen zwei Entitäten, wodurch ein Nutzer einen Dienst verwenden kann.
- Strom, Server, Virtuelle Maschinen (VMs), weitere Hardware, Infrastruktur und Administratoren hinter den Diensten.

Diese Assets haben einen besonderen Schutzbedarf.

Betrachtung der Komponenten und deren Schwachstellen

Für die *Risikobetrachtung* ist es wichtig, zu wissen, wie die Architektur aufgebaut ist und folglich welche Komponenten Schwachstellen haben.

- Bei Identity Provider und Service Provider wird eine *erweiterte lokale Software* eingesetzt. Bei IdPs läuft diese normalerweise auf einem *Webserver*, wie z. B. Apache Tomcat oder Jetty. Über diese Software ist ein lokales I&AM angebunden. Auf Seiten des SPs wird eine Software eingesetzt, um FIM für einen lokal installierten Dienst anzubieten. Dieser ist in den meisten Fällen eine Webanwendung, die wiederum auf einem Webserver läuft. Durch die Erweiterung werden Metadaten automatisch ausgetauscht und die Konfiguration angepasst. Auf Seiten des IdPs erfolgt die *Konfiguration* der Konvertierungsregeln und Filterregelungen automatisch. Durch die Automatisierung erlaubt der

SP dem Nutzer ohne vorherige Prüfung die Nutzung eines Dienstes. Folglich kann, neben bisher vorhandenen Angriffsvektoren, die *Automatisierung* Angriffe erleichtern, um beispielsweise auf einen Dienst zugreifen zu können oder Metadaten eines Angreifers einzuschleusen. Hier müssen folglich Schwachstellen betrachtet werden.

- Die größte Änderung der Architektur erfolgt durch die *zentrale Komponente TTP*. Diese wird im nächsten Kapitel näher vorgestellt. Um die Sicherheit zu gewährleisten, wird sie hier kurz eingeführt. Die TTP erweitert den Lokalisierungsdienst um eine Managementplattform, die über eine *Webanwendung* realisiert wird. Über diese Webanwendung können sich Entitäten registrieren, Fed-RM ihre Föderation verwalten und auch Inter-Föderationen gegründet werden. Die Webanwendung läuft auf einem *Webserver* und speichert Daten in einer *Datenbank* bzw. in einem *Repository*. Darunter liegt zumindest ein *virtueller Server*. Zudem automatisiert dieser Dienst den Metadatenaustausch. So wird der ursprüngliche Request des SPs zwischengespeichert und ein neuer Request an den IdP generiert, um den Nutzer zu authentifizieren. Anschließend werden die Metadaten ausgetauscht und automatisch durch die Erweiterung bei IdP bzw. SP integriert. Da die für das dynamische FIM/Inter-FIM benötigten Daten auf der TTP gespeichert sind, und damit die Auswirkung im Fall eines Angriffs groß ausfallen kann, muss die TTP besonders geschützt werden. Hier werden ebenfalls Schwachstellen betrachtet.

Bei der Analyse der Schwachstellen für die oben genannten Komponenten ergibt sich folgendes Bild:

Server: Die auf dem Server installierte Software ist eine mögliche Schwachstelle, durch die zusätzliche Berechtigungen erlangt werden können. Passwörter können eine weitere Schwachstelle darstellen, wenn sie öffentlich zugänglich oder leicht herauszufinden sind. Leitungen bzw. Kommunikationswege können ebenfalls eine Schwachstelle bilden, genauso wie die Verfügbarkeit.

Virtuelle Maschine: Zusätzlich zur Komponente Server Verfügbarkeit können mögliche offene Ports genutzt werden, um Schwachstellen auszunutzen. Der Hypervisor kann theoretisch ebenso angegriffen werden. Einige bekannten Angriffe auf virtuelle Maschinen basieren auf den Seitenkanal, durch den eine virtuelle Maschine den Speicherinhalt einer anderen VM aus dem Cache lesen kann. Dies wird zum Diebstahl von Chiffrierungsschlüsseln, zur Identifizierung einer physischen Maschine aus einer VM in der Cloud und zum direkten Datenaustausch zwischen virtuellen Maschinen über die physische Ebene ausgenutzt.

Webserver: Durch die Implementierung vorhandene Schwachstellen können ausgenutzt werden. Zudem ist die Verfügbarkeit eine Schwachstelle.

Webanwendung: Die Webanwendung an sich kann durch Programmierfehler und vorhandene Fehler in verwendeter Software Schwachstellen enthalten, beispielsweise durch nicht durchgeführtes Prüfen der Eingaben.

SAML-Implementierung bei Entitäten: Äquivalent zu Wolfgang Hommel [Hom07] (Abschnitt 4.7).

Erweiterungen bei Entitäten: Durch die Automatisierung können Metadaten ohne vorherige Prüfung in die lokale Konfiguration eingetragen werden, was durchaus eine Schwachstelle darstellen kann. So ist die Herausgabe von zu vielen Benutzerinformationen bei falscher Konfiguration auf Seiten des IdPs möglich. Service Provider können den Dienst für Nutzer freigeben, die normalerweise nicht dazu berechtigt wären. Programmierfehler bei den Erweiterungen können weitere Schwachstellen darstellen.

Datenbank: Die Integrität der Daten kann eine Schwachstelle darstellen, genauso wie die Benutzerrechte, wenn beide nicht genügend gesichert sind. Ferner ist die Verfügbarkeit der Datenbank entscheidend.

TTP-Lokalisierungsdienst: Wichtig beim Lokalisierungsdienst sind die Verfügbarkeit des Dienstes sowie Integrität und Authentizität der Daten. Dies sind ebenso die Schwachstellen des Lokalisierungsdienstes.

TTP-Webanwendung: Neben den Schwachstellen einer Webanwendung, bietet die TTP-Webanwendung spezielle Schwachstellen und somit Angriffsarten.

- Durch die Registrierung einer gefälschten Entität können entweder Benutzerinformationen abgegriffen oder Dienste unerlaubt benutzt werden.
- Durch das Vortäuschen eines höheren Vertrauens können ebenfalls Dienste unerlaubt genutzt werden.
- Durch das Erschleichen von höheren Berechtigungen können andere Entitäten ausgeschlossen oder Angreiferentitäten eingeschleust werden.
- Durch das Einbringen von Schadcode kann die Datenbank ausgelesen und verändert werden. Dies kann beispielsweise dazu dienen von den Daten finanziell zu profitieren, mehrere Metadatenaustausche anzustoßen, Entitäten auszuschließen oder hinzuzufügen, Berechtigungen zu ändern oder das Vertrauen einzelner Entitäten zu ändern.

SAML-Kommunikation: Übliche Angriffe auf SAML sind Kollisionsangriffe, Denial of Service (DoS), Man-in-the-Middle, Replay Angriffe und Session Hijacking (vgl. [`SAMLSecure`] [HPM05]). Durch die Verwendung von zusätzlichen Bindings, wie dem Artifact Binding, können weitere Schwachstellen auftreten.

Erweiterte Kommunikation: Durch den automatischen Austausch von Metadaten können Metadaten von Entitäten ausgetauscht werden, die keine technische Vertrauensbeziehung aufbauen möchten.

Angreifermodell

Im Folgenden werden verschiedene Angreifermodelle betrachtet und diese auf die gerade genannten Komponenten und Angriffe übertragen. Die Angreifer sind nach der Position aufgeteilt, d. h. Entitäten allgemein, Service Provider im speziellen, Föderation, Nutzer und Externe, wobei sich Externe in Internet und Besucher/Einbrecher untergliedern lassen. Dabei werden Motivation und Zielsetzung des Angreifers, Fähigkeit des Angreifers, Aufwand für den Angriff und Rückverfolgbarkeit analysiert. Es werden die wahrscheinlichsten bzw. offensichtlichsten Angriffe betrachtet; es können jedoch weitere Angriffe insbesondere auf die Webanwendungen erfolgen, die jedoch durch geeignete Maßnahmen während der Programmierung verhindert werden. Zudem werden Angriffe, die von SAML nicht betrachtet werden, ebenfalls nicht analysiert.

Entitäten als Angreifer

Angriffe, die von sowohl SP als auch IdP durchgeführt werden können, sind nachfolgend dargestellt.

Der Identity Provider kann als Angreifer gegenüber dem SP, und umgekehrt, oder der TTP agieren, wenn z. B. seine Systeme kompromittiert wurden oder ein Administrator seine Berechtigungen missbraucht. Zudem kann es Innentäter geben. Diese Angriffsarten werden im Folgenden beschrieben. Ein Administrator kann seine Berechtigung missbrauchen, wenn er Identitäten vortäuscht, die in Realität nicht existieren oder ihnen höhere Berechtigungen gibt, um die Nutzung von Diensten zu erschleichen.

Motivation/Zielsetzung: SP schaden, indem Täter Dienste nutzt, die er normalerweise nicht nutzen kann oder zusätzliche Metadatenaustausche anstößt. Ein SP kann wiederum IdPs schaden, indem möglichst viele Benutzerdaten abgerufen werden.

Fähigkeit: Administrator.

Aufwand: Machbar.

Rückverfolgbarkeit: Ja.

Es ist zudem möglich, dass ein IdP-Administrator, genauso wie ein SP-Administrator, möglichst viele Metadaten oder andere Daten an die TTP überträgt. Dies kann beispielsweise auf den Plattenplatz, auf die Bandbreite oder die Central Processing Unit (CPU)-Zeit zielen und somit ein DoS-Angriff sein. Um viele Metadaten zu übertragen, muss der IdP-Administrator entweder als Nutzer angemeldet sein oder eine mögliche Schwachstelle der TTP ausnutzen. Bei einem Angriff gegen den eigenen IdP ist der Administrator als Innentäter zu sehen.

Motivation/Zielsetzung: TTP angreifen, möglicherweise eigene Entität schaden.

Fähigkeit: Relativ gering.

Aufwand: Machbar.

Rückverfolgbarkeit: Ja, durch Authentifizierung, bei Schwachstelle schwieriger.

Durch das Erschleichen von höheren Berechtigungen bei der TTP können andere Entitäten ausgeschlossen oder Angreiferentitäten eingeschleust werden. Weitere Manipulationen, wie Höhe des Vertrauens, sind möglich. Hierfür ist mehr Aufwand nötig, um eine höhere Berechtigung zu erlangen, zudem ist die Wahrscheinlichkeit eher gering.

Motivation/Zielsetzung: Andere Entitäten ausschließen.

Fähigkeit: Experte.

Aufwand: Etwas höher.

Rückverfolgbarkeit: Ja, durch Logfiles.

Durch das Einbringen von Schadcode, über die Webanwendung oder am Server, in die TTP kann die Datenbank ausgelesen und verändert werden. Dies kann beispielsweise dazu dienen von den Daten finanziell zu profitieren, mehrere Metadatenaustausche anzustoßen, Entitäten auszuschließen oder hinzuzufügen, Berechtigungen zu ändern oder das Vertrauen einzelner Entitäten zu ändern. Ebenso können die TTP selbst oder die dort abgelegten Dateien verändert werden. Dazu muss Schadcode eingebracht werden. Bei einer Webanwendung ist dies beispielsweise durch Eingabefelder möglich, die nicht die Eingabe und Ausgabe validieren.

Motivation/Zielsetzung: Verschieden.

Fähigkeit: Je nachdem.

Aufwand: Machbar.

Rückverfolgbarkeit: Schwieriger.

Ferner kann ein Administrator versuchen, eine Session einer anderen Entität zu übernehmen, d. h. Session Hijacking, um die Daten einer anderen Entität zu verändern. Jeder Administrator muss sich für die Registrierung der Entität bei der Webanwendung anmelden. Weitere Managementfunktionen stehen über die Webanwendung der TTP zur Verfügung. Wenn ein Angreifer die Session eines Administrators übernimmt, kann er dementsprechend mit dessen Berechtigungen Daten manipulieren. Hierfür ist jedoch ein gezielter Angriff notwendig.

Motivation/Zielsetzung: Andere Entitäten schaden.

Fähigkeit: Experte.

Aufwand: Etwas höher.

Rückverfolgbarkeit: Schwieriger.

Ein Innentäter bei einer Entität kann ebenfalls den eigenen IdP bzw. SP schaden, indem er Software, Hardware stört, oder die Daten bei der TTP ändert. Dies kann beispielsweise dann der Fall sein, wenn der Innentäter die Organisation der Entität bald verlässt.

Motivation/Zielsetzung: Eigene Entität schaden.

Fähigkeit: Relativ gering.

Aufwand: Machbar.

Rückverfolgbarkeit: Kommt drauf an.

Die Daten einer anderen Entität können auch dann geändert werden, wenn der Angreifer über Social Engineering an die Zugangsdaten gelangt. Der Angreifer täuscht hierfür beispielsweise eine Notsituation vor, für die er unbedingt die Zugangsdaten benötigt. Alternativ können die Zugangsdaten, wenn sie beispielsweise am Bildschirm kleben oder ein solcher Zettel aus dem Müll gefischt wird, über andere Wege zum Angreifer gelangen, die er dann ausnutzt.

Motivation/Zielsetzung: Andere Entitäten schaden.

Fähigkeit: Menschenkenntnis.

Aufwand: Machbar.

Rückverfolgbarkeit: Schwierig.

Auch wenn die Wahrscheinlichkeit, dass ein IdP als Angreifer fungiert, relativ gering ist, müssen diese möglichen Angriffe betrachtet werden.

Service Provider als Angreifer

Nachfolgend wird ein expliziter Angriff durch Service Provider betrachtet.

Ebenso wie im bisherigen FIM-Szenario, kann das Abrufen möglichst umfangreicher personenbezogener Daten eine Missbrauchsmöglichkeit durch den SP sein. Diese ist möglich, wenn zu freizügig konfigurierte ARPs ausgenutzt werden. Dieser Effekt wird durch die Automatisierung des Metadatenaustausches und der Konfiguration verstärkt. Dazu muss jedoch auch der Nutzer seinen Consent geben.

Motivation/Zielsetzung: Personenbezogene Daten sammeln.

Fähigkeit: Administrator.

Aufwand: Machbar, durch Automatisierung weniger Zeitaufwändig.

Rückverfolgbarkeit: Ja, durch Logfiles.

Die Wahrscheinlichkeit von SPs als Angreifer ist vorhanden, wenn es insbesondere kostenpflichtige Dienste sind, ansonsten ist die Wahrscheinlichkeit eher gering.

Nutzer als Angreifer

Der Nutzer kann als Angreifer fungieren, wenn er einer Entität schaden, weitere Dienste nutzen oder seine Fähigkeiten ausprobieren will. Über das Erlangen zusätzlicher Berechtigungen oder das Abstreiten der Dienstnutzung hinausgehend, können Nutzer eine große Anzahl an Metadaten-Austausche anstoßen und damit einen oder mehrere Dienste verlangsamen bis hin zum Absturz bringen. Dieser DoS-Angriff kann, wenn sich mehrere Nutzer absprechen oder verschiedene Benutzerkonten besitzen, auch verteilt erfolgen. Ein verteilter Angriff ist ebenfalls durch Schadcode durch Nutzer möglich, wenn auch für externe Angreifer wahrscheinlicher.

Motivation/Zielsetzung: Dienst oder TTP verlangsamen oder zum Absturz bringen, beispielsweise um Vorlesung praktisch ausprobieren oder um Frust über einen Dienst abzureagieren.

Fähigkeit: Gering.

Aufwand: Machbar.

Rückverfolgbarkeit: Ja, durch Authentifizierung.

Denkbar ist zudem, dass ein Nutzer mehrere Benutzerkonten übernimmt und damit einen Maskerade-Angriff betätigt. Diese Angriffe sind nicht speziell durch die Erweiterung ermöglicht. Um an mehrere Benutzerkonten zu gelangen, muss er entweder eine Schwachstelle beim IdP ausnutzen oder über Social Engineering an die Benutzerkonten gelangen.

Motivation/Zielsetzung: Dienst nutzen, der sonst nicht nutzbar ist, oder anderen Nutzer schaden.

Fähigkeit: Mit Wissen

Aufwand: Höher.

Rückverfolgbarkeit: Schwieriger.

Äquivalent zu IdP und SP, kann ein Nutzer Schadcode in die TTP einbringen.

Motivation/Zielsetzung: Daten ändern, auslesen oder löschen. Verschiedene Zielsetzung, wie finanzieller Vorteil, Entitäten schädigen oder Dienste nutzen, wofür kein genug großes Vertrauen vorhanden ist.

Fähigkeit: Experte, zumal Weboberfläche Authentifizierung erfordert und Endnutzer ein Benutzerkonto bräuchten; alternativ Script-Kid, welches den Stoff der Vorlesung ausprobieren möchte.

Aufwand: Höher.

Rückverfolgbarkeit: Möglich durch Logfiles.

Ebenso sind Angriffe über Social Engineering möglich, beispielsweise um einen IdP, SP oder eine TTP zu übernehmen oder an das Benutzerkonto eines anderen Nutzers zu gelangen.

Motivation/Zielsetzung: Entität oder TTP schaden, Dienst nutzen, der sonst nicht genutzt werden kann oder anderem Nutzer schaden.

Fähigkeit: Menschenkenntnis.

Aufwand: Machbar.

Rückverfolgbarkeit: Schwierig.

Föderation als Angreifer

Durch die Erweiterung und der Schnittstelle für Föderation ist es möglich, dass eine Föderationsverwaltung einen Angriff startet. Dieser ist entweder gegen eine Entität oder Föderation gerichtet, alternativ kann ein finanzieller Vorteil die Motivation sein. Zusätzlich kommen Innentäter in Frage, die gegen die eigene Föderation vorgehen. Der Administrator einer Föderation kann beispielsweise einen IdP oder SP in die Föderation aufnehmen, um andere Entitäten anzugreifen, u. a. durch vermehrte Metadatenaustausche. Die Schädigung kann durch finanzielle Anreize oder anderen Beweggründen bedingt sein.

Motivation/Zielsetzung: Schädigung einer Entität.

Fähigkeit: Administrator mit entsprechenden Berechtigungen.

Aufwand: Machbar bis höher.

Rückverfolgbarkeit: Ja, durch Logfiles und Datenbank.

Eine andere Zielsetzung kann das Sammeln von Benutzerdaten sein, um beispielsweise daraus einen finanziellen Vorteil zu erhalten. Der Angreifer bringt dadurch eine weitere Entität in die Föderation und bewirbt sie. Dazu muss der Endnutzer jedoch den Dienst wählen und seinen Consent zum Datenversand geben. Ebenso muss der IdP schlecht konfiguriert

sein.

Motivation/Zielsetzung: Finanzieller Vorteil.

Fähigkeit: Administrator mit entsprechenden Berechtigungen.

Aufwand: Höher.

Rückverfolgbarkeit: Ja, durch Logfiles und Datenbank.

Ein Administrator kann seine Berechtigungen missbrauchen, um Entitäten auszuschließen. Dies ist z. B. möglich, wenn er hierfür Geld bekommt oder aus anderen Beweggründen der Entität schaden möchte.

Motivation/Zielsetzung: Entität schaden bzw. finanzieller Vorteil.

Fähigkeit: Administrator mit entsprechenden Berechtigungen.

Aufwand: Gering.

Rückverfolgbarkeit: Ja, durch Datenbank.

Durch das Erschleichen von höheren Berechtigungen in der TTP können ganze Föderationen ausgeschlossen oder torpedieren werden, indem ihre Daten geändert werden. Dazu muss der Angreifer höhere Berechtigungen bei der TTP erlangen, was durch mögliche Schwachstellen machbar ist.

Motivation/Zielsetzung: Schaden einer Föderation.

Fähigkeit: Administrator mit Wissen.

Aufwand: Höher.

Rückverfolgbarkeit: Ja, durch Datenbank.

Durch das Einbringen von Schadcode kann die Datenbank ausgelesen und verändert werden. Dies kann beispielsweise dazu dienen von den Daten finanziell zu profitieren, mehrere Metadatenaustausche anzustoßen, Entitäten auszuschließen oder hinzuzufügen, Berechtigungen zu ändern oder das Vertrauen einzelner Entitäten zu ändern. Zudem kann die TTP kompromittiert werden, z. B. um weiteren Schadcode zu laden oder Nutzer bzw. Entitäten zu schaden, wenn mobiler Schadcode eingebaut ist.

Motivation/Zielsetzung: Daten ändern, auslesen oder löschen. Verschiedene Zielsetzung, wie finanzieller Vorteil, Entitäten schädigen oder Dienste nutzen, wofür kein genug großes Vertrauen vorhanden ist.

Fähigkeit: Experte.

Aufwand: Höher.

Rückverfolgbarkeit: Möglich durch Logfiles.

Ein weiterer möglicher Angriff ist Social Engineering, wo versucht wird das Benutzerkonto einer anderen Föderation zu übernehmen oder einzelne Entitäten zu schädigen.

Motivation/Zielsetzung: Entität, andere Föderation oder TTP schaden.

Fähigkeit: Menschenkenntnis.

Aufwand: Machbar.

Rückverfolgbarkeit: Schwierig.

Wenn die Föderation als Betreiber der TTP auftritt, sind weitere Angriffsvektoren denkbar. Somit ist es für Administratoren der TTP möglich Ressourcen zu missbrauchen, um beispielsweise Daten zu stehlen oder zu verändern, wie bereits oben genannt. Ein Innentäter kann auch Schadcode auf den Server spielen, um gewollt Daten zu verändern, zu löschen oder einen Dienstausfall hervorzurufen. Mobiler Code kann in die Webanwendung bzw. auf den Webserver eingebracht werden, um teilnehmende Entitäten oder Nutzer anzugreifen.

Motivation/Zielsetzung: Neben finanziellen Vorteilen kann auch ein größtmöglicher Schaden ein Ziel sein.

Fähigkeit: Administrator mit entsprechenden Berechtigungen oder Wissen.

Aufwand: Machbar.

Rückverfolgbarkeit: Möglich durch Logfiles, wenn diese nicht geändert wurden.

Zudem kann die Infrastruktur der TTP angegriffen werden, z. B. durch einfaches Stecker ziehen, einem Brand oder Wasser. Ein Innentäter kann hierbei versuchen die TTP zu deaktivieren und ggf. einen größtmöglichen Schaden anzurichten.

Motivation/Zielsetzung: Neben finanziellen Vorteilen kann auch ein größtmöglicher Schaden ein Ziel sein.

Fähigkeit: Jeder mit Zutritt.

Aufwand: Kommt drauf an, wie die Infrastruktur abgesichert ist.

Rückverfolgbarkeit: Je nachdem, wie der Zutritt gesichert ist.

Ein Innentäter eines Betreibers kann durch Schwachstellen oder entsprechenden Berechtigungen die Daten der TTP exportieren und weiter verkaufen.

Motivation/Zielsetzung: Finanzielle Vorteile.

Fähigkeit: Administrator.

Aufwand: Kommt drauf an.

Rückverfolgbarkeit: Je nachdem möglich durch Logfiles, wenn diese nicht geändert wurden.

Insgesamt ist ein Innentäter, beispielsweise ein Mitarbeiter, dem gekündigt wurde, am schwierigsten zu ermitteln.

Externer Angreifer

Als nächstes werden externe Angreifer betrachtet. Es kommen sowohl Angriffe, die gegen die TTP gerichtet sind, aber auch Angriffe gegen einzelne Service Provider oder IdPs in Frage. Als Angreifer können Konkurrenten, Skript-Kids, aber auch professionelle Hacker auftreten, die beispielsweise aus einem anderen Land stammen. Während Skript-Kids verhältnismäßig einfach normalerweise abzuwehren sind, sind Hacker, die Geld und Zeit haben, eine größere Gefahr. Besonders kritisch ist der Lokalisierungsdienst, die TTP, durch die Anzahl an Daten und durch ihre exponierte Lage. Dies ist insbesondere der Fall, da durch die Erweiterung mehr Daten über SPs, IdPs und ihre Kooperationen gespeichert sind. Mögliche Angriffe betreffen beispielsweise:

- Einbringen von mobilem Code oder Schadcode auf der TTP, um beispielsweise Daten auszulesen, Eingaben mitzuprotokollieren oder Entitäten anzugreifen.
- SQL Injections, um Informationen aus der Datenbank abzurufen oder Daten zu verändern. Dies kann z. B. die Informationen zum Metadatenaustausch zwischen IdPs und SPs oder die Löschung von Konkurrenten betreffen.
- Replay-Angriffe durch das Einspielen mitgehörter Kommunikationsnachrichten, mit dem Ziel an weitere Informationen zu gelangen oder ungeplante Aktionen anzustoßen.
- DoS bzw. Distributed Denial of Service (DDoS) Angriffe, um die Verfügbarkeit der TTP oder einzelner Entitäten einzuschränken.
- Angriffe auf die Benutzerverwaltung.
- Session Hijacking.

Diese und weitere Angriffsvektoren werden im Folgenden genauer beschrieben.

Durch das Einbringen von Schadcode oder durch Ausnutzung von anderen Lücken kann

die Datenbank ausgelesen und verändert werden. Dies kann beispielsweise dazu dienen von den Daten finanziell zu profitieren, mehrere Metadatenaustausche anzustoßen, Entitäten auszuschließen oder hinzuzufügen, Berechtigungen zu ändern oder das Vertrauen einzelner Entitäten zu ändern. Die Datenbank der TTP kann beispielsweise durch SQL Injections verändert werden, wodurch die Integrität der Daten nicht mehr sichergestellt ist. Durch Veränderung der Berechtigungen kann die Vertraulichkeit leiden.

Motivation/Zielsetzung: Finanzieller Vorteil oder Profilbildung.

Fähigkeit: Experte.

Aufwand: Höher.

Rückverfolgbarkeit: Möglich durch Logfiles.

Durch Einbringen von Schadcode auf der TTP können teilnehmende Nutzer oder Entitäten infizieren werden. Zudem kann es möglich sein, dass dadurch die TTP angegriffen wird. Daten können durch Schadcode ebenso manipuliert werden.

Motivation/Zielsetzung: Entität schaden oder TTP schaden, ggf. finanzieller Vorteil oder Ressourcen für andere Aktivitäten durch infizierte Rechner.

Fähigkeit: Mit Vorwissen, alternativ erfolgreicher Skript-Kid.

Aufwand: Höher.

Rückverfolgbarkeit: Möglich durch Logfiles und Monitoring.

Durch das Erschleichen von Berechtigungen oder dem Ausnutzen von Schwachstellen können Entitäten ausgeschlossen und somit ihnen geschadet werden. Dazu muss ein externer Angreifer ein Benutzerkonto bei der TTP erhalten oder über Schwachstellen in die TTP gelangen und möglicherweise root-Rechte erlangen.

Motivation/Zielsetzung: Entität schaden.

Fähigkeit: Experte, zumal kein Benutzerkonto auf der TTP vorhanden.

Aufwand: Höher.

Rückverfolgbarkeit: Möglich durch Logfiles und Monitoring.

Ferner ist es möglich, dass ein externer Angreifer mehrere Benutzerkonten übernimmt und damit einen Maskerade-Angriff betätigt.

Motivation/Zielsetzung: Dienst nutzen, der sonst nicht nutzbar ist, oder anderen Nutzer

schaden.

Fähigkeit: Mit Wissen.

Aufwand: Höher.

Rückverfolgbarkeit: Schwieriger.

Zudem können externe Angreifer Metadaten für andere Entitäten auszutauschen, um den Dienst TTP oder eine Entität zeitweilig unnutzbar zu machen. Diese und weitere DoS bzw. DDoS Angriffe richten sich gegen die Verfügbarkeit.

Motivation/Zielsetzung: Entität schaden oder TTP schaden.

Fähigkeit: Mit Vorwissen, alternativ erfolgreicher Skript-Kid.

Aufwand: Machbar.

Rückverfolgbarkeit: Möglich durch Logfiles und Monitoring.

Äquivalent zum vorherigen Abschnitt, d. h. Entität als Angreifer, kann sich ein vermeintlicher Service Provider bei der TTP registrieren, um möglichst umfangreiche personenbezogene Daten zu erhalten. Der Name des vermeintlichen SPs sollte dabei ähnlich zu einem beliebten SP sein. Nutzer müssen auch hier dem Dienst vertrauen und ihre Daten freigeben.

Motivation/Zielsetzung: Finanzieller Vorteil oder Profilbildung.

Fähigkeit: Mit Vorwissen.

Aufwand: Höher.

Rückverfolgbarkeit: Möglich durch Logfiles.

Ferner ist es möglich, dass sich ein Externer einen IdP registriert. Auf dem IdP hat er mindestens einen Nutzer, um Dienste von SPs zu nutzen. Dazu muss der Angreifer allerdings erst einmal einen IdP aufsetzen und SPs finden, die ihn akzeptieren.

Motivation/Zielsetzung: Finanzieller Vorteil, Dienstnutzung.

Fähigkeit: Mit Vorwissen.

Aufwand: Höher.

Rückverfolgbarkeit: Möglich durch Logfiles.

Ein Externer kann versuchen die erweiterte Kommunikation zwischen IdP, SP und TTP

abzuhören, um Nutzerinformationen zu erhalten. Eine Aufzeichnung kann auch dazu dienen über Replay mehr Informationen zu bekommen.

Motivation/Zielsetzung: Finanzieller Vorteil oder Profilbildung.

Fähigkeit: Mit Vorwissen.

Aufwand: Machbar.

Rückverfolgbarkeit: Ggf. möglich durch Logfiles.

Ein Externer kann versuchen eine Föderation zu registrieren, die ähnlich zu einer bekannten Föderation klingt. Dies kann das Ziel haben, möglichst viele IdPs und SPs in die eigene Föderation zu locken, Informationen zu erhalten und gleichzeitig eine andere Föderation zu schädigen. Falls der Angreifer ein Zertifikat, Audit oder andere Vorgaben hat, die nur gegen Bezahlung möglich sind, hat der Angriff auch einen finanziellen Hintergrund.

Motivation/Zielsetzung: Finanzieller Vorteil, Informationen, andere Föderation schaden.

Fähigkeit: Mit Vorwissen.

Aufwand: Höher.

Rückverfolgbarkeit: Teils durch Datenbank.

Ein Externer kann, genauso wie alle anderen Arten von Angreifern, über Social Engineering an Benutzerkonten gelangen, um damit einer TTP, Föderation, einzelnen Entitäten oder Nutzern zu schädigen.

Motivation/Zielsetzung: Entität, Föderation, Nutzer oder TTP schaden.

Fähigkeit: Menschenkenntnis.

Aufwand: Machbar.

Rückverfolgbarkeit: Schwierig.

Wie bei jeder Anwendung, kann ein externer Angreifer Schwachstellen bei Server, virtueller Maschine oder der Webanwendung als Einfallstor nutzen. Die Ziele hierfür sind unterschiedlich, wie finanzieller Vorteil, Schaden oder Dienstnutzung.

Motivation/Zielsetzung: Unterschiedlich.

Fähigkeit: Experte.

Aufwand: Höher.

Rückverfolgbarkeit: Ggf. möglich durch Logfiles.

Als letzten Vektor seien Besucher und Einbrecher genannt, die beispielsweise physischen Schaden anrichten können. Dies ist u. a. durch Feuer oder Kabel ziehen möglich.

Motivation/Zielsetzung: Schaden oder finanziellen Vorteil.

Fähigkeit: Jeder, der sich physischen Zutritt verschaffen kann.

Aufwand: Kommt drauf an, wie die Infrastruktur abgesichert ist.

Rückverfolgbarkeit: Je nachdem, wie der Zutritt gesichert ist.

Externe Angreifer, wenn nicht Skript-Kids, können mehr Schaden anrichten, da sie meist über andere Mittel oder anderes Wissen verfügen. Zudem ist ihre Motivation eine andere als bei Teilnehmern der TTP.

Schutzmaßnahmen und Kosten

Damit Angriffe entweder durch Maßnahmen vorab vereitelt werden oder im Nachhinein gefunden werden, müssen mögliche *Maßnahmen* zusammengestellt und bewertet werden. Basierend auf den eben ausgeführten Angriffen, werden verschiedene Maßnahmen genannt. Im Anschluss werden die Maßnahmen noch genauer aufgegliedert. Dabei ist zu beachten, dass die Maßnahmen die Daten beim Transport, bei der Speicherung und der Verarbeitung schützen sollen. Mögliche Maßnahmen basierend auf den verwendeten Systemen sind die Folgenden:

Server: Zugriff über ein Gateway-System, offene und nicht benötigte Ports schließen, kryptographische Prüfsummen, Intrusion Detection System (IDS), Protokollierung und Monitoring sowie feingranulare Zugriffsrechte können beispielsweise als Schutz dienen.

Virtuelle Maschine: Bei virtuellen Maschinen sind die Maßnahmen ähnlich wie bei reellen Servern.

Webserver: Sichere Konfiguration ist hierbei entscheidend. Beispiele dafür sind in vielen Fällen das root-Passwort ändern und dafür ein zweites Benutzerkonto einrichten, tendenziell eher mit Schlüssel als mit Passwort anmelden, Version des Webservers und Serverinformationen verstecken und Timeout-Werte anpassen.

Webanwendung: Bei Webanwendungen sind Authentifizierung, Autorisierung, Ein- und Ausgabevalidierung, Session-Management, Fehlerbehandlung und Protokollierung wichtig. Ferner sollen z. B. eine sichere Anbindung von Hintergrundsystemen, eine sichere Systemarchitektur und feingranulare Zugriffsrechte eingesetzt werden.

SAML-Implementierung bei Entitäten: Keine weiteren Maßnahmen als bei Wolfgang Hommel [Hom07] (Abschnitt 4.7) beschrieben.

Erweiterungen bei Entitäten: Vertrauen, Consent, sichere Implementierung, Attributfilter; LoA/LoT kann durch Föderationen oder externe unabhängige Organisationen bestätigt werden.

Datenbank: Auch hier sind Zugriffskontrolle, verschlüsselte Passwörter bzw. Verschlüsselung vertraulicher Daten, kryptographische Prüfsummen, Logging und Sicherheitsgateways mögliche Sicherheitsmaßnahmen gegen die eben genannten und weiteren möglichen Angriffen. Möglichst keine personenbezogenen Daten auf TTP speichern, wichtige bzw. personenbezogene Daten verschlüsseln, TTP möglichst gut absichern, Monitoring.

TTP-Lokalisierungsdienst: Der Lokalisierungsdienst soll einzig die Entitäten der TTP anzeigen können. Veränderungen beim Lokalisierungsdienst müssen bemerkt werden. Zudem soll die TTP den Metadatenaustausch mit Entitäten, die nicht registriert sind, verbieten. Ein weiterer Hinweis auf Angriffe liefern Logdateien, die den Metadatenaustausch mitprotokollieren sollen.

TTP-Webanwendung: Neben möglichen Angriffen auf eine Webanwendung, bietet die TTP-Webanwendung spezielle Ziele und Angriffsarten. Folglich benötigt die TTP-Webanwendung spezielle Schutzmaßnahmen und muss gesondert betrachtet werden.

- Um keine falschen Entitäten aufzunehmen, soll die TTP die Registrierung validieren, beispielsweise über ein Zertifikat, eine E-Mail-Adresse oder dem Erstellen einer Seite auf dem Webserver.
- Vertrauen überprüfen, protokollieren.
- Angriffe sollen auch durch restriktive Zugriffsrechte vermieden werden. Zudem werden alle wichtigen Aktionen, wie Hinzufügen und Ausschließen von Entitäten protokolliert.
- Ein- und Ausgabevalidierung sollen helfen Angriffe über die Webanwendung zu vermeiden. Bei größeren oder wichtigen Veränderungen der Datenbank sollen Historientabellen beschrieben werden mit dem ursprünglichen Status und den geänderten Werten.

SAML-Kommunikation: IdP und SP müssen eine sichere Konfiguration aufweisen. So sollen IdPs nicht alle Benutzerinformationen ausgeben, sondern nur die, die benötigt werden. Hierbei ist zu beachten, dass keine internen sensiblen Daten versendet werden. Daneben gibt es u. a. Nutzerauthentifizierung und aktive Sessions, Verwendung von TLS bzw. SSL, Signaturen, einer kurzen Validitätsdauer der Assertions und die ID eines Requests.

Erweiterte Kommunikation: Nutzerauthentifizierung vor Austausch.

Im Folgenden werden mögliche Maßnahmen, unterschieden in organisatorische und technische Maßnahmen, aufgezeigt. Daneben werden die Maßnahmen in präventive, detektive und reagierende Maßnahmen untergliedert. Gegebenenfalls werden die oben genannten Maßnahmen ergänzt, um ein möglichst vollständiges Bild zu erhalten.

Organisatorischen und technischen Maßnahmen

Die folgenden organisatorischen und technischen Maßnahmen sind bei einer Architektur für verteilten dynamischen Metadatenaustausch relevant.

Als *organisatorische Maßnahme* kann beispielsweise die Schulung von Mitarbeitern gelten. Zudem soll es Regelungen geben, wer welche Berechtigungen auf Servern und in Serverräumen erhält und wie Lofiles ausgewertet werden sollen. Ein zusätzlicher Security Incident Response Prozess kann bei Security Incidents im Bereich FIM helfen, Schwachstelle zu entdecken, den möglichen Angriff zu analysieren, den Soll-Zustand wieder herzustellen und alles zu dokumentieren. Um die Level of Assurance und Level of Trust sinnvoll einsetzen zu können, muss organisationsintern geklärt werden, welchen LoA/LoT die kooperierenden Entitäten benötigen, also welcher Schutzbedarf in der Organisation vorhanden ist, und welchen LoT/LoA die Organisation selbst bieten kann. Weitere interne Maßnahmen können Audits und die Implementierung eines Vier-Augen-Prinzipes sein. Bezüglich der TTP können interne Regelungen zur Überprüfung der gemachten Informationen, wie Zertifikate, oder der Berechtigung des Administrators helfen.

Neben den organisatorischen Maßnahmen können *technische Maßnahmen* dynamisches FIM absichern. Die technischen Maßnahmen sind mengenmäßig überlegen. Allgemein geltende technische Maßnahmen sind beispielsweise Firewalls, Intrusion Detection Systeme, Logdateien, automatische Rekonfigurierung und Virenscanner. Zugriff über ein Gateway-System, kryptografische Prüfsummen, Protokollierung und Monitoring sowie restriktive, feingranulare Zugriffsrechte können als weiterer Schutz dienen. Ferner sollen z. B. eine sichere Anbindung von Hintergrundsystemen und eine sichere Systemarchitektur eingesetzt werden.
Bei Webanwendungen helfen technische Maßnahmen, wie Zufallszahlen als SessionID, Tokens, Validierung der Eingabe und Ausgabe sowie Session Timeouts, die bei der Implementierung der Webanwendung zu beachten sind. Zudem gelten die folgenden Maßnahmen:

- Berechtigung der Administrator validieren. Dies kann u. a. bei der Registrierung geschehen, beispielsweise über ein Zertifikat, eine E-Mail-Adresse oder dem Erstellen einer Seite auf dem eigenen Webserver. Zudem können Teilnahmen an Föderationen oder Audits eine gewisse Sicherheit gewährleisten.

- Verwendung von Hashes, und bei Passwörtern Salt, um sie schwieriger zu erraten bzw. die Integrität nachweisen zu können.

- Veränderungen beim Lokalisierungsdienst müssen bemerkt werden.

- Zudem soll die TTP den Metadatenaustausch mit Entitäten, die nicht registriert sind,

verbieten.

- Bei größeren oder wichtigen Veränderungen der Datenbank sollen Historientabellen mit dem ursprünglichen Status und den geänderten Werten beschrieben werden.

- Neben einer sicheren Implementierung der TTP und der Erweiterungen, soll durch ein Schwellenwertverfahren bei der TTP festzustellen sein, wenn ein IdP oder SP überdurchschnittlich viele Metadaten austauscht bzw. überdurchschnittlich viele Attribute anfordert oder versendet.

- Eine Anwendung des LoA-/LoT-Framework soll technisch helfen nur vertrauenswürdige Entitäten als Kooperationspartner auszuwählen.

- Beim dynamischen Metadatenaustausch sollen sich Nutzer bei der TTP authentifizieren, um unnötigen Metadatenaustausch zurückverfolgen zu können.

- SP und IdP müssen ihre Konfiguration anpassen, so dass nicht jeder IdP bzw. SP den Dienst nutzen kann. Für Identity Provider ist es wichtig, dass nicht alle Benutzerinformationen versendet werden. Nutzer sollen der Weitergabe ihrer Daten über uApprove oder ein anderes Consent-Modul zustimmen.

Präventiven, detektierenden, reagierenden Maßnahmen

Diese und weitere Maßnahmen lassen sich in die Kategorien präventiv, detektiv und reagierend aufteilen. Die meisten technischen Maßnahmen sind gleichzeitig präventiv, auch wenn es detektierende und reagierende Maßnahmen gibt.

Präventive Maßnahmen sind Maßnahmen, die im Vorfeld getroffen werden, um einen Angriff zu vereiteln oder zumindest zu erschweren. Dazu zählen Mitarbeiterschulungen und technische Maßnahmen, wie Firewalls. Um einen Angriff auf die TTP zu erschweren, müssen hohe Sicherheitsmaßnahmen angewandt werden. So können Firewalls, Anti-Viren-Software und die Verteilung der TTP auf mehrere Sicherheitszonen, wie in Wolfgang Hommels Arbeit [Hom07] (Abschnitt 4.7.2) beschrieben, helfen. Restriktive Zugriffsrechte und gegebenenfalls der Zugriff auf die TTP über ein Gateway-System Angriffe von Anfang an vereiteln. Ferner gelten die folgenden beispielhaften präventiven Maßnahmen:

- Ein- und Ausgabe-Validierung in der Webanwendung,

- Session Management,

- Überprüfung der Berechtigung des Administrators und

- korrekte Konfiguration.

Fast alle weiteren technischen Maßnahmen, die oben genannt wurden, sind präventive Maßnahmen. Sie werden nicht weiter ausgeführt, um Wiederholungen zu vermeiden.

Neben den präventiven Maßnahmen, können *detektive Maßnahmen* einen aktuellen Angriff bemerken. Dies ist beispielsweise durch Intrusion Detection Systeme und weitere Monitoring-Systeme möglich. Schwellenwertverfahren und Logfiles bieten weitere Informationen über mögliche Angriffe. Ein Vier-Augen-Prinzip kann ebenfalls einen aktuellen Angriff erkennen und ihn vereiteln.

Als *reagierende Maßnahme* können der Security Incident Response Prozess sowie die automatische Rekonfiguration betrachtet werden. Audits und Backups gelten ebenfalls als reagierende Maßnahme. Als eine Art Backups sind die Historientabellen zu betrachten, die den ursprünglichen Status und die geänderten Werte enthalten und somit das Wiedereinspielen der ursprünglichen Werte erlauben.

Kosten

Um die Maßnahmen zu bewerten, müssen die damit verbundenen Kosten bekannt sein. Basierend darauf, der Eintrittswahrscheinlichkeit und der Auswirkung des Schadens wird entschieden, ob die Maßnahme durchgeführt wird oder nicht. Eintrittswahrscheinlichkeit und Auswirkung sind teils schwierig zu beurteilen. Nachfolgend wird grob abgeschätzt, welche Kosten benötigt werden, um eine sichere TTP zu gewährleisten. Dabei wird davon ausgegangen, dass sie sicher implementiert wurde.

- Durch die Erweiterung wird zusätzliche Zeit bei IdP und SP benötigt, um LoA/LoT zu wählen, und um die Erweiterung zu konfigurieren. Falls LoA bzw. LoT Audits vorsieht, können hier weitere Kosten entstehen.

- Föderationen müssen ihren Aufnahmeprozess festlegen, diesen in der TTP konfigurieren und Mitglieder verwalten. Ebenso wie bei IdP und SP sind es somit hauptsächlich Personalkosten.

- Dies ist auf Seiten der TTP anders. Die Kommunikation zwischen Entitäten und einer TTP muss sicher sein, d. h. wichtige Informationen verschlüsseln. Ein Nutzer muss sich erst authentifizieren, bevor Metadaten ausgetauscht werden.

- Die Webanwendung der TTP inklusive Datenbank und Ablagesystem müssen sicher implementiert werden. Dies beinhaltet Unit-Tests und weitere Tests bei der Implementierung, was wiederum Zeit kostet.

- Der Server der TTP muss zudem, wie oben beschrieben, abgesichert werden, was ebenso Zeit kostet. IDS und weitere technischen Maßnahmen können bei Verwendung von kommerziellen Produkten ebenfalls kosten. Monitoring und Verbesserungen sind begleitende Maßnahmen für den Betrieb.

Bisher bestehende Schnittstellen, die Wolfgang Hommel [Hom07] (Abschnitt 4.7) bereits beschrieb, bleiben durch die Erweiterung bestehen und sollen weiterhin genutzt werden, um ein effektives und effizientes Security Management zu gewährleisten. Neben der Auditierbar-

keit sind u. a. Ein- und Ausgabe-Validierung, Authentifizierung und Session Management, Validierung der Integrität der gespeicherten Daten und Authentisierung von Bedeutung. Mögliche Schwellenwertverfahren und neuronale Netze können zudem darüber entscheiden, ob ein möglicher Angriff erfolgt oder nicht. Ein weiterer, entscheidender Aspekt ist die Konfiguration, beispielsweise von LoA und LoT. Zusätzlich können Nutzer die Herausgabe ihrer Daten durch das Consent Management, wie bei uApprove, einschränken.

4.8.2. Change Management

Unter *Change Management* wird in Anlehnung an Information Technology Infrastructure Library (ITIL) ein transparenter Prozess verstanden, der standardisierte Methoden zur Durchführung von Änderungen verwendet. ITIL unterscheidet zwei Arten von Veränderungen:

- *Pre-authorized Changes* oder auch Standard-Changes genannt sind kleinere Änderungen, die relativ häufig anfallen, nach einem bereits bekannten Muster ablaufen und geringe Risiken bergen. Sie bedürfen keiner gesonderten Genehmigung.

- ITIL versteht als zweite Art *größere Änderungen*, Normal-Changes, deren Auswirkungen komplex oder unbekannt bei der Stellung des Änderungsantrags sind. Diese Änderungen müssen durch ein so genanntes Change Advisory Board (CAB) analysiert und genehmigt werden.

Zudem existieren Emergency Changes, um im Notfall schnell reagieren zu können. Allgemein sollen Änderungen nur geplant und koordiniert durchgeführt werden. Dabei sind die Risiken zu beachten. Im Unterschied zur organisationsinternen Betrachtung von ITIL, können Änderungen im FIM-Umfeld auch organisationsübergreifende Auswirkungen haben, wie bereits von Wolfgang Hommel [Hom07] (Abschnitt 4.8) aufgezeigt. Es ist möglich, dass, abhängig von der Organisationsform der Föderation, Änderungen gegen den Willen einer Organisation beschlossen und umgesetzt werden. Dies betrifft insbesondere die Terminplanung für Änderungen, Forward Schedule of Changes (FSC) genannt. Dasselbe gilt inzwischen auch föderationsübergreifend in Inter-Föderationen. Folglich muss die Auswirkung von Änderungen unterschieden werden in

- Änderungen, die die eigene Organisation betreffen.

- Änderungen, die eine gesamte Föderation betreffen.

- Änderungen, die mehrere Föderationen betreffen.

Änderungen können laut Wolfgang Hommel Metadaten, ARPs, Konfiguration und Föderationszusammensetzung betreffen. Ferner sind Änderungen an den folgenden Aspekten möglich:

- benötigte Attribute,
- technisches Vertrauen zwischen zwei Entitäten,
- LoA bzw. LoT der eigenen Entität, wie im nächsten Kapitel aufgezeigt,
- mindestens benötigte LoA bzw. LoT und
- Policies sowie Aufnahmeprozess der Föderationen.

Die folgende Betrachtung der Änderungen am Change Management durch die Erweiterung um eine TTP erfolgt sowohl aus Sicht der Entitäten als auch aus Sicht der Föderationsverwaltung.
Aus Sicht der einzelnen Entität ist für die *Einführung der neuen Architektur* ein CAB notwendig, da eine größere Änderung erfolgt, deren Auswirkungen komplex sind. So ist die Erweiterung zu installieren. Ferner soll die Entität festlegen, welche Mindestanforderung sie an andere Entitäten stellt und welche Anforderungen sie erfüllen kann. Änderungen am Level of Assurance oder Level of Trust sollen, da sie teilweise komplexe Auswirkungen auf die zukünftige Verwendung von Diensten bzw. zukünftige Nutzer durch das CAB analysiert und genehmigt werden. Zudem erfolgen zukünftig kleine Änderungen, so genannte pre-authorized Changes, automatisch. Diese kleineren Änderungen umfassen die Erstellung von technischem Vertrauen zwischen zwei Entitäten und die entsprechende Anpassung der lokalen Konfiguration inklusive ARPs auf Seiten des IdPs. Geänderte Metadaten gelten ebenso als kleine Änderungen, die keiner gesonderten Genehmigung bedürfen. Falls ein SP die Attribute, die er benötigt, an geänderte Anforderungen anpasst, erfolgt diese Änderung zwar nicht häufig, jedoch sind die Änderungen überschaubar. Daher ist nicht unbedingt ein CAB notwendig. Wenn die Anzahl der angefragten Attribute sich stark verändert, ist eine Genehmigung durch ein CAB trotzdem sinnvoll. Geänderte Teilnahmen an föderationsähnlichen Strukturen haben teilweise ebenfalls komplexe Änderungen, insbesondere wenn Policies akzeptiert werden müssen, so dass diese Änderungen durch das CAB ebenso genehmigt werden sollen. Virtuelle Föderationen ohne Policies können dahingegen als kleinere Änderungen angesehen werden, die beispielsweise bei wechselnden Projekten relativ häufig anfallen können. Wichtig ist bei allen pre-authorized Changes, dass die Änderungen auditierbar und nachvollziehbar sind.

Für Föderationsverwaltungen ändert sich Folgendes: Die Einführung muss ebenfalls durch ein CAB genehmigt werden. Dasselbe gilt für die Erstellung von und Änderungen an Policies sowie dem Aufnahmeprozess in die Föderation. Dies kann gleichzeitig mit der Einführung geschehen. Änderungen an Policies, die Auswirkungen auf die gesamte Föderation haben, sollten möglichst durch ein gemeinsames CAB aus allen Teilnehmern und der Föderationsverwaltung selbst beschlossen werden. Ebenso hat die Teilnahme an einer Inter-Föderation Auswirkung auf alle Teilnehmer und soll durch ein gemeinsames CAB beschlossen werden.

Etablierte Prozesse bleiben durch die Erweiterung um die TTP bestehen, jedoch werden viele Abläufe automatisiert. Diese zuvor manuell getätigten Abläufe sind pre-authorized Changes, die keiner gesonderten Genehmigung bedürfen. Die Einführung der TTP muss für

jede Entität und Föderationsverwaltung durch ein Change Advisory Board analysiert und genehmigt werden. Änderungen, die komplexe Auswirkungen haben, sind ebenfalls durch das CAB zu genehmigen und lassen sich nicht automatisieren. Dazu zählen auch die ausgeklammerten schriftlichen SLAs.

4.9. Bewertung

Zur Bewertung der Architektur und zur Verdeutlichung der noch zu entwickelnden Werkzeuge wird der Anforderungskatalog aus Kapitel 2 herangezogen. Anforderungen, die bereits von SAML-Implementierungen erfüllt werden und an denen keine Änderungen vorgenommen wurden, werden nicht weiter betrachtet, da keine Verschlechterung möglich ist.

Die essentiellen Anforderungen wurden wie folgt erfüllt:

[FA-Föderation]: Priorität 1

- Beschreibung: Die Bildung und Verwaltung mehrerer Föderationen, egal ob national oder international, in einem oder in mehreren Sektoren, muss unterstützt werden.
- Bewertung: Durch die Möglichkeit der Bildung von parallelen Föderationen und dynamischen virtuellen Föderationen wird diese Anforderung vollständig erfüllt.

[FA-Grenzüberschreitend]: Priorität 1

- Beschreibung: Die Benutzerinformationen müssen über nationale und Sektor-Grenzen hinweg versendet werden können.
- Bewertung: Durch die in dieser Arbeit konzipierte Architektur und die Möglichkeit der Bildung dynamischer virtueller Föderationen über nationale (Föderations-) Grenzen hinaus wird diese Anforderung ebenfalls erfüllt.

[FA-Integration]: Priorität 1

- Beschreibung: Die Integration in die lokale Umgebung der IdPs und SPs muss ohne erheblichen Aufwand geschehen.
- Bewertung: Die Integration wurde im Abschnitt 4.7 betrachtet. Nachdem vor allem eine Erweiterung der lokalen Software installiert und konfiguriert werden muss, ist die Integration ohne großem Aufwand möglich. Zusätzlich soll die Erweiterung konfiguriert werden.

[FA-Konfiguration]: Priorität 1

- Beschreibung: Der Administrator muss die Möglichkeit haben die Integration der Metadaten und den Vertrauensaustausch zu konfigurieren.

- Bewertung: Diese Anforderung wird durch die Konfigurationsmöglichkeiten erfüllt. Diese werden im nächsten Kapitel noch näher beschrieben.

[FA-Langlebigkeit]: Priorität 1

- Beschreibung: Die Architektur der Föderationen muss eine möglichst langfristige und dauerhafte Lösung ermöglichen.

- Bewertung: Da die erweiterte Architektur auf bestehende Lösungen aufbaut und mehrere Arten an Föderationen erlaubt (siehe Klassifikation von Föderationen), wird diese Anforderung als erfüllt angesehen.

[NFA-Dokumentation]: Priorität 1

- Beschreibung: Die Spezifikation des Föderationskonzeptes muss offen gelegt werden.

- Bewertung: Durch diese Arbeit und Veröffentlichungen ist diese Anforderung erfüllt.

[NFA-Koexistenz]: Priorität 1

- Beschreibung: Es muss möglich sein, dass eine Entität mehreren Föderationen zugehört.

- Bewertung: Diese Anforderung wird durch das Konzept dieser Arbeit unterstützt.

[NFA-Skalierbarkeit]: Priorität 1

- Beschreibung: Das System muss skalierbar für eine beliebige Anzahl an Teilnehmern und dynamischen virtuellen Föderationen sein. Skalierbarkeit hat hierbei unterschiedliche Bedeutungen, die allesamt erfüllt werden sollen. Zum einen soll das System für eine beliebige Anzahl an Teilnehmern und Föderationen skalieren. Zum anderen soll die Skalierbarkeit bezüglich des Metadatenaustausches verbessert werden. Hierbei werden die Anzahl an manuellen Schritten und die Größe des Metadatensatzes beachtet.

- Bewertung: Nachdem nur benötigte Metadaten ausgetauscht werden, ist die in dieser Arbeit konzipierte Lösung gut skalierbar. Zudem lässt das Konzept beliebig viele Teilnehmer und Föderationen zu.

[SEC-Authentifizierung]: Priorität 1

- Beschreibung: Benutzer müssen sich vor der Nutzung eines Dienstes authentifizieren können.

- Bewertung: Diese Anforderung wird durch SAML erfüllt. Zudem muss sich der Nutzer authentifizieren, um unnötigen Metadatenaustausch zu vermeiden.

[ORG-Validierung]: Priorität 1

- Beschreibung: Registrierte Organisationen müssen validiert und überprüft werden können. Dies kann beispielsweise über eine Instanz der Föderation geschehen.

- Bewertung: Eine solche Validierung erfolgt bei der Registrierung. Dieser Vorgang wird im Kapitel 5 näher beschrieben. Weitere Überprüfungen durch externe Instanzen werden unterstützt.

[DSA-Datenschutz]: Priorität 1

- Beschreibung: Die Datenschutzrichtlinien und Datenschutzgesetze müssen eingehalten werden können.

- Bewertung: In SAML kann grundsätzlich die Einhaltung der Datenschutzrichtlinie als `EntityCategory` in den Metadaten angegeben werden. Eine automatische Überprüfung ist nicht möglich, wodurch dies im Rahmen dieser Arbeit nicht weiter betrachtet wurde.

Folglich sind die meisten essentiellen Anforderungen, so weit technisch möglich, umgesetzt worden. Weitere essentielle Anforderungen werden durch zusätzlich konzipierte Werkzeuge erfüllt, wie im nächsten Kapitel zu sehen. Bei den wichtigen Anforderungen ist der Stand wie folgt:

[FA-Attributswahl]: Priorität 2

- Beschreibung: Der Nutzer hat durch eine Webanwendung die Auswahl, welche Attribute er welchem Dienst zur Verfügung stellt, äquivalent zu UMA.

- Bewertung: Durch die Arbeit ist eine Attributswahl, äquivalent zu uApprove.jp, nicht zusätzlich konzipiert worden. Jedoch ist es möglich anstelle von uApprove uApprove.jp zu verwenden, um eine Attributswahl zu ermöglichen.

[FA-Automatisierung]: Priorität 2

- Beschreibung: Der Vertrauensaufbau soll on demand und automatisch geschehen können, um dynamisch auf geänderte Anforderungen der Nutzer reagieren zu können.

- Bewertung: Diese Anforderung wird durch die Architektur mit der Möglichkeit

zur Automatisierung der bisher manuellen Schritte erfüllt. Die Automatisierung wird im nächsten Kapitel genauer beschrieben.

[FA-Dynamik]: Priorität 2

- Beschreibung: Die Architektur der Föderationen soll sich dynamisch an geänderte Anforderungen anpassen können, um FIM in dynamischen Umgebungen einsetzen zu können. Als zusätzlicher Aspekt soll die Dynamik der Metadaten betrachtet werden, um deren Skalierbarkeit zu verbessern.

- Bewertung: Diese Anforderung wird durch die Architektur und die Schnittstelle zur Föderationsverwaltung erfüllt.

[FA-Fehlermanagement]: Priorität 2

- Beschreibung: Die Behandlung von Fehlern soll unterstützt werden.

- Bewertung: Innerhalb der Konzeption wurde auf Fehlermanagement durch aussagekräftige Fehlermeldungen und der Information über Fehler Wert gelegt.

[FA-Initiierung]: Priorität 2

- Beschreibung: Der Nutzer soll den Austausch von Daten und somit Aufbau des Vertrauensverhältnisses initiieren können, um u. a. die Wartezeit zu verringern.

- Bewertung: Der Nutzer triggert den Metadatenaustausch, folglich ist diese Anforderung umgesetzt. Dies wird im nächsten Kapitel näher erläutert.

[FA-LoA]: Priorität 2

- Beschreibung: Dem SP soll angezeigt werden, welche Datenqualität der IdP liefern kann. Zugleich soll die Einteilung in eine bestimmte Klasse transparent erfolgen

- Bewertung: Diese Einteilung ist durch SAML grundsätzlich gegeben (vgl. [SAMLAC] [KCM+05]) und soll in Kapitel 5 genauer definiert werden.

[FA-LoT]: Priorität 2

- Beschreibung: Die Vertrauenswürdigkeit von SPs soll überprüft und geeignet dargestellt werden.

- Bewertung: Diese Einteilung ist durch SAML grundsätzlich gegeben und soll in Kapitel 5 genauer definiert werden.

[FA-Metadaten]: Priorität 2

- Beschreibung: Die Metadaten enthalten Informationen über die Organisationen

und sollen passend generiert werden.

- Bewertung: Diese Anforderung wird durch SAML großteils erfüllt. Durch das neue Konzept müssen Metadaten nur einmal spezifiziert werden, um für mehrere Föderationen zu gelten. Dies hat den Vorteil, dass Entitäten, die Kooperationen mit Entitäten in mehreren Föderationen haben, bei Änderungen nur einen Metadatensatz austauschen müssen.

[FA-Reichweite]: Priorität 2

- Beschreibung: Die Lösung soll alle benötigten Identity Provider und Service Provider umfassen.

- Bewertung: Nachdem bisherige feste Föderationen durch die Architektur aufgelöst werden, wird die Reichweite vergrößert; siehe dynamische virtuelle Föderationen und Inter-Föderationen.

[FA-Rollen]: Priorität 2

- Beschreibung: Organisationen sollen parallel in mehreren Rollen agieren können.

- Bewertung: Diese Anforderung wird bereits durch SAML erfüllt. Die Architektur erlaubt zudem das Agieren in mehreren Rollen.

[NFA-Implementierungsunabhängigkeit]: Priorität 2

- Beschreibung: Das System soll unterschiedliche Implementierungen akzeptieren.

- Bewertung: Das hier beschriebene Konzept ist allgemein gültig. Auch wenn die TTP, wie in der Implementierung zu sehen, auf dem Shibboleth Centralized Discovery Service aufbaut und die Erweiterungen der IdP bzw. SP Software ebenfalls auf Shibboleth basiert, ist es möglich Erweiterungen für SimpleSAMLphp und andere Implementierungen von SAML zu konzipieren.

[NFA-Performanz]: Priorität 2

- Beschreibung: Das System soll performant sein, beispielsweise auf Anfragen in akzeptabler Zeit reagieren.

- Bewertung: Nachdem der Metadatenaustausch synchron erfolgt zur Anfrage des Nutzers, wird von einer guten Performanz ausgegangen.

[NFA-Protokollunabhängigkeit]: Priorität 2

- Beschreibung: Das System soll unterschiedliche Protokolle akzeptieren.

- Bewertung: Das hier beschriebene Konzept ist allgemein gültig. Auch wenn die TTP, wie in der Implementierung zu sehen, auf dem Shibboleth Centralized Discovery Service aufbaut, ist es möglich andere Protokolle, wie OpenID Connect, einzusetzen.

[SEC-Auditing]: Priorität 2

- Beschreibung: Das System und seine Teilkomponenten sollen auditierbar sein.
- Bewertung: Die Nachvollziehbarkeit auf Seiten der Entitäten wurde nicht geändert. Die TTP protokolliert den Verbindungsaufbau mit, wodurch auch diese Komponente auditierbar ist.

[SEC-Automatisierung]: Priorität 2

- Beschreibung: Bei der Automatisierung des Datenaustausches soll die Sicherheit gewährleistet werden.
- Bewertung: Die Sicherheit des in dieser konzipierten Ansatzes wurde im vorherigen Abschnitt betrachtet. Durch entsprechende Maßnahmen kann die Sicherheit gewährleistet werden.

[SEC-Initiierung]: Priorität 2

- Beschreibung: Die Initiierung des Vertrauensaufbaus soll sicher sein.
- Bewertung: Durch die Kontrollinstanz bei der TTP, Überprüfung der Zertifikatsinformationen und organisatorische Möglichkeiten, ist eine sichere Initiierung möglich.

[SEC-Integration]: Priorität 2

- Beschreibung: Die Integration soll aus Sicht der Sicherheit betrachtet werden.
- Bewertung: Die Integration und die Sicherheit wurden in diesem Kapitel betrachtet.

[ORG-Automatisierung]: Priorität 2

- Beschreibung: Die Automatisierung der Verbindungserstellung hat Auswirkungen auf die Organisation, die beachtet werden sollen.
- Bewertung: Die Automatisierung wurde insbesondere auf Hinblick des Change Managements im vorherigen Abschnitt betrachtet.

[ORG-Föderation]: Priorität 2

- Beschreibung: Die Bildung und Verwaltung nach verschiedenen Föderationsmodellen soll passend durch die Lösung unterstützt werden.

- Bewertung: Diese Anforderung wird durch die Unterstützung unterschiedlicher Föderationsmodelle erfüllt.

[ORG-Konfiguration]: Priorität 2

- Beschreibung: Auf organisatorischer Ebene betrifft die Konfiguration die internen Abläufe.

- Bewertung: Die Erweiterungen der IdP bzw. SP Software lässt sich konfigurieren und kann somit angepasst werden. Ansonsten baut die Erweiterung auf bestehende, eingesetzte Software auf, die Möglichkeiten zur Konfiguration aufweist. Die Konfiguration wird im nächsten Kapitel noch detaillierter beschrieben.

[ORG-LoA]: Priorität 2

- Beschreibung: Die Verlässlichkeitsklasse betrifft interne Abläufe und die Konfiguration.

- Bewertung: Der LoA soll für Service Provider entsprechend den internen Anforderungen konfiguriert und angepasst werden. Gleichzeitig ist es wichtig, dass der LoA der IdPs ohne erheblichen Mehraufwand ermittelt werden kann. Dies wird, zusätzlich zum bereits beschriebenen Change Management, im nächsten Kapitel näher betrachtet. SAML unterstützt die Anwendung des LoAs.

[ORG-LoT]: Priorität 2

- Beschreibung: Das Vertrauen in den SP betrifft interne Abläufe sowie die Konfiguration.

- Bewertung: Der LoT soll für Identity Provider entsprechend den internen Anforderungen konfiguriert und angepasst werden. Gleichzeitig ist es wichtig, dass der LoT der SPs ohne erheblichen Mehraufwand ermittelt werden kann. Dies wird, zusätzlich zum bereits beschriebenen Change Management, ebenfalls im nächsten Kapitel betrachtet. Level of Trust kann, ähnlich wie Level of Assurance, in SAML angegeben werden.

[ORG-Migration]: Priorität 2

- Beschreibung: Für die Migration sollen geeignete Lösungen bereitgestellt werden.

- Bewertung: Ein entsprechendes Konzept wurde in diesem Kapitel erstellt.

[ORG-Supportprozesse]: Priorität 2

- Beschreibung: Die Lösung soll Schnittstellen zu den organisationsinternen Supportprozessen, wie dem Service Desk und dem Change Management, aufweisen.

- Bewertung: Durch die Erweiterung können bestehende Schnittstellen weiter verwendet werden, wie im Abschnitt 4.8 dargestellt.

[DSA-LoT]: Priorität 2

- Beschreibung: Durch bzw. trotz der Funktion Level of Trust soll der Datenschutz eingehalten werden.

- Bewertung: LoT soll nur eine bessere Einschätzung des SPs erlauben, nachdem aktuell kein Mittel hierfür besteht. Dieser LoT soll, wie in diesem Kapitel veranschaulicht, dazu dienen, vorab zu prüfen, ob der SP den Anforderungen entspricht. Somit wird diese Anforderung besser erfüllt als bisher. Level of Trust wird im nächsten Kapitel beschrieben.

Folglich werden fast alle wichtigen Anforderungen bereits mit der in diesem Kapitel konzipierten Lösung erfüllt. Der Einsatz einer verbesserten Version von uApprove.jp kann zu einer größeren Entscheidungsfreiheit der Nutzer führen. Die Festlegung von Level of Assurance und Level of Trust sowie die Möglichkeiten der Attributskonvertierungen werden im folgenden Kapitel betrachtet. Bei den unwichtigen Anforderungen ergibt sich folgendes Bild:

[FA-Entscheidungshilfe]: Priorität 3

- Beschreibung: Der Benutzer bekommt eine Entscheidungshilfe, welche Attribute von welcher Art von Dienst benötigt werden und kann daraufhin seine [FA-Attributswahl] anpassen.

- Bewertung: Eine Unterstützung für den Nutzer ist zwar sinnvoll, jedoch können Informationen durch IdP oder SP auch die Bedeutung eines Attributs in Kombination mit uApprove.jp erklären. Nachdem eine Attributswahl nicht speziell konzipiert wurde, aber uApprove.jp eingesetzt werden kann, wurde diese Anforderung nicht weiter betrachtet.

[FA-Homeless]: Priorität 3

- Beschreibung: Nutzer ohne Heimatorganisation können die Dienste trotzdem nutzen. Eine entsprechende Schnittstelle zu einem Dienst für heimatlose Nutzer oder die direkte Integration eines solchen Dienstes muss gegeben sein.

- Bewertung: Da es benutzerfreundlicher ist, keine zusätzlichen IdPs für Nutzer ohne Heimatorganisation zu verwenden und dafür die Verwendung von IdPs außerhalb der eigentlichen Föderation zu erlauben, wurde kein Homeless-IdP in die Lösung integriert, sondern die Reichweite erhöht. Nichtsdestotrotz kann auf Grund der Architektur ein Homeless-IdP registriert werden, um Nutzer ohne Hei-

matorganisation zu verwalten.

[FA-SelfAsserted]: Priorität 3

- Beschreibung: Der Nutzer hat die Möglichkeit eigene Attribute, beispielsweise für die Profilbildung, zu speichern.
- Bewertung: Diese Anforderung kann durch eine Verknüpfung mit UMA umgesetzt werden. Nachdem UMA aktuell noch entwickelt wird und keine klaren Schnittstellen bekannt sind, wurde dieses Konzept nicht weiter betrachtet. Es wird jedoch an einer Umsetzung in SAML gearbeitet, wodurch eine Einbindung in Zukunft möglich sein soll.

[FA-SLA]: Priorität 3

- Beschreibung: Die Anzahl der benötigten Verträge kann reduziert bzw. dem Nutzerkreis angepasst werden.
- Bewertung: Durch den bedarfsgerechten Austausch von Metadaten müssen nur noch notwendige Verträge abgeschlossen werden. SLAs sind trotz der Erweiterung weiterhin möglich.

[NFA-Portabilität]: Priorität 3

- Beschreibung: Das System kann unabhängig von der Hardware und dem Betriebssystem verwendet werden können.
- Bewertung: Das in dieser Arbeit allgemeine vorgestellte Konzept und die Implementierung basierend auf Shibboleth Centralized Discovery Service sind genauso portabel wie dieser Lokalisierungsdienst selbst.

[SEC-Metadaten]: Priorität 3

- Beschreibung: Die Verwaltung und Aktualisierung sicherheitsrelevanter Konfigurationsparameter, wie Metadaten, kann weitgehend automatisiert werden, so dass diese Metadaten nur noch an einer zentralen Stelle gepflegt werden müssen.
- Bewertung: Eine lokale Pflege der Metadaten bzw. die Pflege der Metadaten bei der TTP ist möglich, so dass diese Anforderung erfüllt wird.

[ORG-Metadaten]: Priorität 3

- Beschreibung: Die Metadaten können passend mit den Informationen über die Organisationen generiert werden.
- Bewertung: SAML-Implementierungen erlaubt die Generierung von Metadaten,

wobei Änderungen teilweise manuell getätigt werden müssen. Die in dieser Arbeit durchgeführte Erweiterung ändert an der Funktionalität nichts.

[ORG-SLA]: Priorität 3

- Beschreibung: Die Datenqualität und andere Gütemerkmale können passend spezifiziert werden. Die Einhaltung der SLAs kann automatisch überprüft werden.
- Bewertung: Durch die Erweiterung werden vorhandene bzw. nicht vorhandene Funktionen in der Implementierung nicht geändert.

[DSA-CoCo]: Priorität 3

- Beschreibung: Die Datenschutzrichtlinie der EU kann überprüft werden.
- Bewertung: In SAML kann grundsätzlich die Einhaltung des CoCo als EntityCategory in den Metadaten angegeben werden. Eine automatische Überprüfung der Einhaltung ist nicht möglich, wodurch dies im Rahmen dieser Arbeit nicht weiter betrachtet wurde.

Die Anforderungen der Priorität wurden zum größten Teil ebenfalls umgesetzt. Wie bereits bei den wichtigen Anforderungen erwähnt, werden Level of Assurance bzw. Level of Trust und Schema im folgenden Kapitel als zusätzliche Werkzeuge betrachtet. Self-Asserted Attribute wurde nicht weiter betrachtet, da die Forscher um User-Managed Access noch keine Umsetzung für SAML realisiert haben. Eine Einbindung kann zu einem späteren Zeitpunkt jedoch angedacht werden.

Die Tabelle 4.38 visualisiert die Umsetzung der Anforderungen. Sie zeigt auf, dass zumindest auf konzeptioneller Ebene alle essentiellen Anforderungen ganz und alle weiteren Anforderungen zumindest partiell erfüllt wurden. Die Anforderungen [FA-Kontext], [NFA-Usability], [SEC-Kontext] und [SEC-Multilateral] wurden zu Beginn des Kapitels in Vorgehensweise bewusst ausgeklammert. Durch die Verwendung des Frameworks von Helmut Reiser [Rei08] kann jedoch die Anforderung [SEC-Multilateral] erfüllt werden.

Die Verbesserungen hinsichtlich des Erfüllungsgrades der Anforderungen ergeben sich durch das Konzept der dynamischen virtuellen Föderationen; andererseits ermöglicht die Architektur der TTP einen dynamischen, durch den Nutzer initiierten Metadatenaustausch. Im Gegensatz zu den vorgestellten SAML-Implementierungen wurde die Automatisierung [FA-Automatisierung] und Dynamik [FA-Dynamik] verbessert. Nutzer haben nun die Möglichkeit den Metadatenaustausch zu initiieren [FA-Initiierung]. Die Einbindung von Homeless IdP ist, wie alle weiteren Identity Provider, möglich [FA-Homeless]. Zusätzlich wurde die Reichweite [FA-Reichweite] erhöht, wodurch dieser Dienst seltener benötigt wird. Zudem werden SLAs [FA-SLA] durch die Architektur unterstützt.

Die Automatisierung [SEC-Automatisierung] und Initiierung [SEC-Initiierung] wurden auch als Sicherheitsanforderung erfüllt. Bei den organisatorischen Anforderungen wurden die

Anforderung	Priorität	Bewertung	Anforderung	Priorität	Bewertung
Funktionale Anforderungen					
[FA-Aktualisierung]	2	+	[FA-Konfiguration]	1	+
[FA-Attributswahl]	2	o	[FA-Konnektor]	2	+
[FA-Automatisierung]	2	+	[FA-Kontext]	3	+
[FA-Datenkategorisierung]	1	+	[FA-Langlebigkeit]	1	+
[FA-Dynamik]	2	+	[FA-LoA]	2	o
[FA-Entscheidungshilfe]	3	-	[FA-Lokalisierung]	1	+
[FA-Fehlermanagement]	2	+	[FA-LoT]	2	o
[FA-Föderation]	1	+	[FA-Metadaten]	2	+
[FA-Grenzüberschreitend]	1	+	[FA-Monitoring]	2	-
[FA-Homeless]	3	+	[FA-Pull&Push]	1	+
[FA-Identitätswahl]	3	o	[FA-Reichweite]	2	+
[FA-Initiierung]	2	+	[FA-Rollen]	2	+
[FA-Integration]	1	+	[FA-SelfAsserted]	3	o
[FA-Interaktion]	1	+	[FA-SLA]	3	+
Nichtfunktionale technische Anforderungen					
[NFA-Dokumentation]	1	+	[NFA-Portabilität]	3	+
[NFA-Implementierungsunabhängigkeit]	2	+	[NFA-Protokollunabhängigkeit]	2	+
[NFA-Koexistenz]	1	+	[NFA-Skalierbarkeit]	1	+
[NFA-Performanz]	2	+	[NFA-Usability]	2	o
Sicherheitsanforderungen					
[SEC-ARPs]	1	+	[SEC-Kontext]	3	+
[SEC-Auditing]	2	+	[SEC-LoA]	2	+
[SEC-Authentifizierung]	1	+	[SEC-LoT]	2	+
[SEC-Automatisierung]	2	+	[SEC-Metadaten]	3	+
[SEC-Datenübertragung]	1	+	[SEC-Multilateral]	1	+
[SEC-Initiierung]	2	+	[SEC-Systemsicherheit]	1	+
[SEC-Integration]	2	+			
Organisatorische Anforderungen					
[ORG-Automatisierung]	2	+	[ORG-Metadaten]	3	+
[ORG-Föderation]	2	+	[ORG-Migration]	2	+
[ORG-Konfiguration]	2	+	[ORG-SLA]	3	+
[ORG-LoA]	2	o	[ORG-Supportprozesse]	2	+
[ORG-LoT]	2	o	[ORG-Validierung]	1	+
Datenschutzanforderungen					
[DSA-CoCo]	3	+	[DSA-LoT]	2	o
[DSA-Datenschutz]	1	+	[DSA-Selbstbestimmung]	1	+
[DSA-Initiierung]	3	+	[DSA-Zustimmung]	2	+
[DSA-Interaktion]	2	+			

Tabelle 4.38.: Bewertung des Konzepts

Registrierung [ORG-Registrierung], Validierung [ORG-Validierung], Konfiguration [ORG-Konfiguration] sowie die Anforderungen [ORG-Automatisierung], [ORG-Metadaten], [ORG-SLA] und [ORG-Supportprozesse] erfüllt. Aus datenschutzrechtlicher Sicht sind [DSA-ARPs], [DSA-CoCo], [DSA-Initiierung] und [DSA-Interaktion] verbessert worden.

Die Implementierungsunabhängigkeit [NFA-Implementierungsunabhängigkeit] ist durch die Erweiterung des Lokalisierungsdienstes gegeben. Die prototypische Erweiterung der IdP- und SP-Software erfolgt jedoch auf Basis von SAML, da dieses Protokoll am häufigsten eingesetzt wird und die Anforderungen am besten erfüllt. Zudem wurde durch die in diesem Kapitel konzipierte Architektur die Skalierbarkeit [NFA-Skalierbarkeit] bezüglich des Systems und der Metadaten verbessern.

Die Aspekte TTP, LoA bzw. LoT und ein universales Schema wurden in diesem Kapitel bewusst ausgeklammert und als Black Box betrachtet. Um den Erfüllungsgrad der Anforderungen weiter zu erhöhen, werden im nächsten Kapitel die Lösungen dieser wichtigen Anforderungen betrachtet. Ebenso werden die Anforderungen [FA-Realisierbarkeit], [FA-Schema], [NFA-OpenSource], [ORG-Realisierbarkeit], [ORG-Registrierung], [ORG-Schema] und [DSA-ARPs] detaillierter spezifiziert.

Kapitel 5

Werkzeuge

Inhalt dieses Kapitels

Im Kapitel 4 wurde eine skalierbare Architektur für Federated Identity Management konzipiert, die anhand von verschiedenen Modellen genauer definiert wurde. Dabei wurde auf die Integration von Security und Change Management geachtet. Das Konzept von dynamischen virtuellen Föderationen und Inter-Föderationen sowie einer Föderationsverwaltung wurde eingeführt. Um diese Modelle umzusetzen, wurde bereits eine Trusted Third Party als

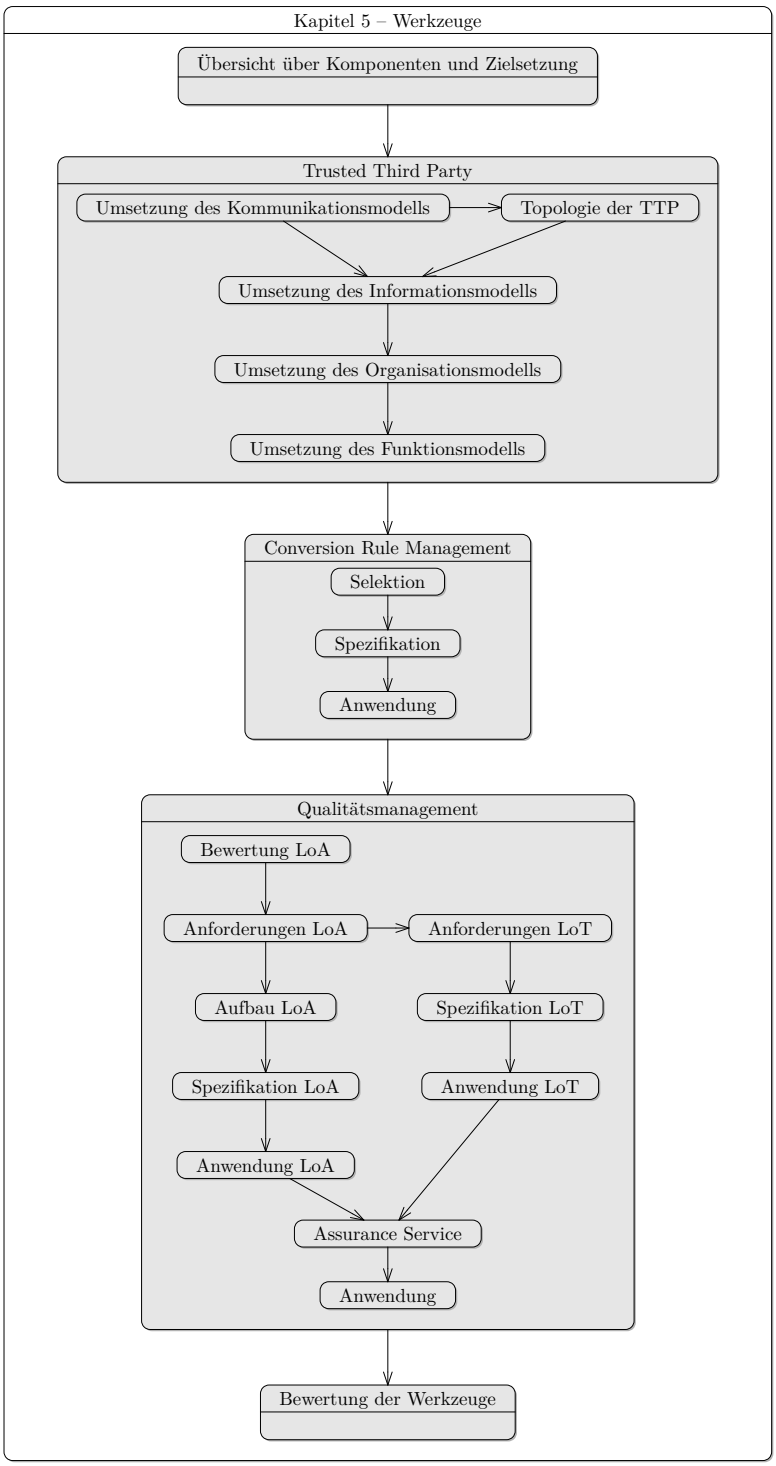

Abbildung 5.1.: Vorgehensmodell in diesem Kapitel

zentrale Komponente beschrieben. Diese Trusted Third Party dient bei der Managementarchitektur als Managementplattform für dynamisches FIM und den damit verbundenen Funktionalitäten.

In diesem Kapitel liegt die Zielsetzung auf ausgewählten zusätzlichen Werkzeugen und Klassifikationen, die zur besseren Erfüllung der Anforderungen nötig sind. Wie bereits im vorherigen Kapitel beschrieben, kann die Architektur durch verschiedene Werkzeuge optimiert werden. Hierfür werden verschiedene Ansätze aus Forschung und Praxis zu Rate gezogen, mit den Anforderungen abgeglichen und eigene Ideen eingebracht. Zunächst werden die Selektion der in diesem Kapitel beschriebenen Werkzeuge und die Zielsetzung erläutert:

- Managementplattform MdFIM der Managementarchitektur als Trusted Third Party,
- Conversion Rule Management und
- Trust Management mit Level of Trust und Level of Assurance.

Um die Managementplattform zu konzipieren, wird in Abschnitt 5.2 die *Trusted Third Party* eingeführt, die das Management von dynamischen FIM unterstützt. Hierbei werden Kommunikationsmodell, Informationsmodell, Organisationsmodell und Funktionsmodell aus dem vorherigen Kapitel umgesetzt. Die Funktionen dienen dazu, die Anforderungen zu erfüllen. Der Name TTP wird als technischer Begriff verwendet, während *MdFIM* die Managementplattform beschreibt. Die Managementplattform bildet die Grundlage für die folgenden beiden Werkzeuge.

Auf Grund der heterogenen, gewachsenen Strukturen werden lokal andere Attribute definiert als in den Föderationen verwendet. Damit trotzdem die Benutzerinformationen ausgetauscht werden können, müssen Attribute konvertiert werden. Bisher geschieht dies manuell pro Administrator. Da durch die manuelle Erstellung der *Konvertierungsregel*, in der Regel nachdem ein Benutzer seinen zuständigen Administrator informiert hat, die Automatisierung nicht mehr synchron, sondern asynchron ablaufen würde, werden im Abschnitt 5.3 unterschiedliche Ansätze betrachtet und ein *Conversion Rule Management* konzipiert.

Trotz oder insbesondere auf Grund des dynamischen Metadatenaustausches muss es für Entitäten trotzdem möglich sein, IdPs bzw. SPs schnell auf ihre Verlässlichkeit einschätzen zu können. Im Kapitel 3 wurden bereits verschiedene Möglichkeiten vorgestellt, wie der *Level of Assurance* bzw. die Verlässlichkeitsklassen ermittelt werden können. Trotz der bisher vorhandenen Normen und Vorgehensweisen in den einzelnen Föderationen wird auf der Ebene von Inter-Föderationen bisher auf eine Angabe verzichtet. Um die Gründe zu evaluieren, wird ein Vergleich der Normen mit der Vorgehensweise in den Föderationen im Abschnitt 5.4.1 stattfinden. Basierend darauf wird ein neuer Level of Assurance spezifiziert, anhand dem IdPs eingeteilt werden können. Dieser erfolgt insbesondere intern zur Umrechnung. Um den Identity Provider ebenfalls eine Sicherheit zu bieten, sollen auch Service Provider durch den so genannten *Level of Trust* spezifiziert werden können. LoT ist das Äquivalent zum LoA und wird in Abschnitt 5.4.2 entwickelt. Beide Klassifikationen sind ein fester Bestandteil des

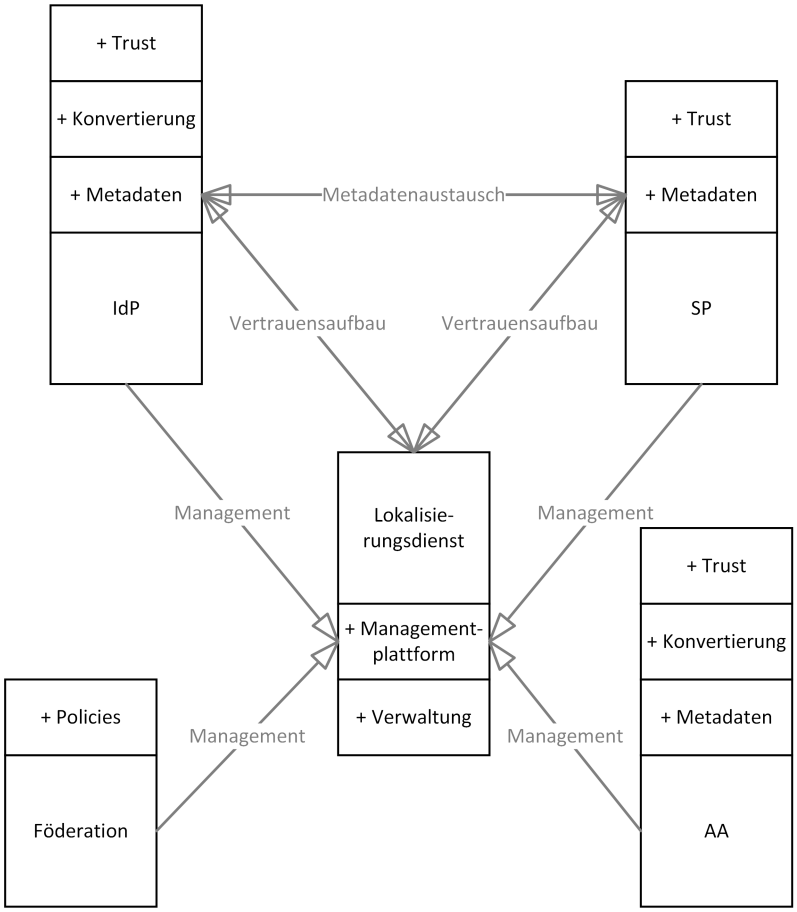

Abbildung 5.2.: Überblick über die notwendigen Erweiterungen der beteiligten Komponenten

Trust Managements, welches in Abschnitt 5.4 betrachtet wird.

Die genaue Vorgehensweise in diesem Kapitel wird in der Abbildung 5.1 dargestellt.

5.1. Übersicht über Komponenten

In den folgenden Abschnitten werden die hierfür nötigen Anpassungen und Erweiterungen aufgezeigt. Diese Erweiterungen sind in Abbildung 5.2 dargestellt und werden im Folgenden kurz erläutert, bevor sie anschließend spezifiziert werden.

5.1.1. Trusted Third Party mit der Managementplattform MdFIM

MdFIM ist die zentrale Komponente in der in dieser Arbeit beschriebenen Managementarchitektur mitsamt der im vorherigen Kapitel beschriebenen Funktionalitäten, Rollen, Informationen und Kommunikationen. In diesem Abschnitt wird MdFIM nur allgemein erklärt, bevor die Managementplattform im nächsten Abschnitt näher gehend anhand der beschriebenen Modelle definiert wird.

Funktionalität

Die Managementplattform soll die im Funktionsmodell beschriebenen Funktionalitäten, wie Configuration Management und Member Management, bereitstellen. Die Grundlage für das Datenmodell ist das Informationsmodell, welches im vorherigen Kapitel beschrieben wurde. Einen Überblick über die Datenhaltung wird im nächsten Abschnitt gegeben. Die Kommunikation beruht auf dem Kommunikationsmodell und wird ebenfalls genauer erläutert. Die API stellt hier das Entwurfsmuster einer Fassade dar, die die dahinter liegenden Module für die Benutzung vereinfacht. Bei MdFIM bestehen die üblichen Schnittstellen, d. h. die Aktionen werden protokolliert. Die Administratoren haben die Möglichkeit zur Konfiguration, beispielsweise wie die Information von aktualisierten Metadaten versendet wird und ob sie zur eigenen Föderation E-Mails erhalten. Weiter wird eine Fehleranalyse und eine Leistungsüberwachung benötigt, um die Performanz zu gewährleisten und um gegebenenfalls eingreifen zu können.

Sicherheit

Um die Sicherheit zu gewährleisten, müssen sich Entitäten und Föderationen autorisieren, bevor sie MdFIM nutzen können. Zudem wird eine Authentifizierung notwendig, wodurch MdFIM eine Benutzerverwaltung besitzt. Bevor Nutzer den Metadatenaustausch initiieren können, müssen sie sich bei ihrem IdP authentifizieren. Dies hilft unnötige Transaktionen zu vermeiden bzw. diese zumindest zurückverfolgen zu können. Die Organisationen sollen ihre Metadaten bzw. Policies möglichst vor Ort haben, um das Angriffsziel von MdFIM zu verringern und um die Performanz zu erhöhen. MdFIM selbst ist durch übliche Sicherheitsmechanismen gesichert und hochverfügbar. Zudem werden die Aktionen mitprotokolliert, wie aus der Sicherheitsbetrachtung in Abschnitt 4.8 ersichtlich. Dies ermöglicht eine bessere Fehleranalyse und die Auditierbarkeit von MdFIM.

Begründung

Bei der Entwicklung von MdFIM ist ein ganzheitlicher Ansatz wichtig, der auf bereits existierende Software aufbaut und diese für dynamischen Austausch der Metadaten erweitert. Die

einheitliche Lösung für Föderationsverwaltungen reduziert den Verwaltungsaufwand, sowohl für Entitäten als auch für Föderationen. Die Komponente ist so konzipiert, dass sie modular erweitert und angepasst werden kann. Sie basiert auf einem OpenSource-Produkt und ist selbst OpenSource [NFA-OpenSource]. Dabei ist [NFA-Implementierungsunabhängigkeit] und [NFA-Protokollunabhängigkeit] wichtig. Die Erweiterung ist so ausgelegt, dass sie möglichst performant ist ([NFA-Performanz]), beispielsweise durch lokale Speicherung der Dateien. Zugleich wird darauf geachtet, dass ein möglichst geringer Aufwand für die Migration besteht ([ORG-Migration]).

5.1.2. Conversion Rule Management

Das Conversion Rule Repository ist Teil des Conversion Rule Managements, welches eine wichtige Funktionalität des Konzeptes darstellt. Das Conversion Rule Management wird hierbei durch MdFIM bereitgestellt. Dieses Werkzeug wird in Abschnitt 5.3 vorgestellt.

Funktionalität

Damit Benutzerinformationen in das Format des SPs konvertiert werden können, werden Konvertierungsregeln eingesetzt. Um diese zu nutzen, müssen folgende Funktionalitäten vorhanden sein:

- Erstellen von Konvertierungsregeln,
- Ändern von Konvertierungsregeln,
- Löschen von Konvertierungsregeln,
- Wiederverwenden von Konvertierungsregeln,
- Validieren von Konvertierungsregeln,
- Konfiguration der Automatisierung bezüglich Konvertierungsregeln,
- Informationen über Änderungen bzw. nicht vorhandene Konvertierungsregeln,
- Akzeptieren von Konvertierungsregeln durch Föderationen oder Inter-Föderationen und
- Übernehmen der Eigentümerschaft von Konvertierungsregeln.

Entsprechend der verwendeten Implementierung bei IdP und AA müssen Anpassungen an die Regeln gemacht werden. Die Konvertierungsregeln sollen anschließend in die lokale Konfiguration eingefügt werden. Nachdem die Implementierungen unterschiedliche Formate und

Regeln zur Konvertierung verwenden, wie in Kapitel 3 zu sehen, müssen die Konvertierungsregeln möglichst generisch gespeichert werden. Diese generischen Regeln sollen anschließend in Abstimmung mit der Implementierung in das jeweils benötigte Format transferiert werden. Zur Speicherung und Kommunikation werden, um eine Einheitlichkeit zu erreichen, dieselben Medien verwendet, wie bei MdFIM.

Begründung

Um die Automatisierung der vorhandenen manuellen Workflows abzuschließen, ist es wichtig auch die Konvertierung der Benutzerinformationen inklusive der Anpassung in der lokalen Konfiguration zu automatisieren. Nachdem kein weltweit gültiges Schema, äquivalent zu einer weltweit gültigen Föderation, machbar ist, müssen weiterhin auch unterschiedliche Schemata behandelt werden. Um Benutzerinformationen zwischen verschiedenen Schemata zu konvertieren, wird in diesem Kapitel ein generisches Schema entwickelt sowie die weitere Kommunikation beschrieben. Das Sicherheitskonzept der MdFIM sowie deren Managementschnittstellen werden übernommen, um eine einheitliche Lösung zu erhalten.

5.1.3. Trust Management

Ein weiteres wichtiges Werkzeug dieser Arbeit ist das noch im Abschnitt 5.4 zu entwickelnde Trust Management, welches als Black Box im vorherigen Kapitel beschrieben wurde.

Funktionalität

Neben dem technischen Vertrauen werden hier die weiteren Aspekte des Vertrauens angesprochen. Dazu zählen die beiden Trust-Klassifikationen Level of Trust und Level of Assurance, wodurch IdP und SP besser eingeordnet werden sollen. Um das Werkzeug zu nutzen, muss MdFIM die Verwaltung von Vertrauensgraden, der Konfiguration und dem Vergleich verschiedener Klassifikationen ermöglichen.

Begründung

Damit Verlässlichkeitsklassen föderationsübergreifend genutzt werden können, soll das Werkzeug Trust Management aufgebaut werden. Dieses Werkzeug soll es erlauben unterschiedliche LoAs zu vergleichen. Zudem soll eine Klassifikation von SPs hierdurch ermöglicht werden. Das Trust Management ist ein wichtiger Bestandteil von MdFIM. Folglich sollen Datenmodell und Kommunikation von MdFIM hierfür verwendet werden.

5.1.4. Unterstützende Komponenten

Darüber hinaus werden, wie Wolfgang Hommel [Hom07] bereits in Abschnitt 4.4.13 beschrieb, unterstützende Komponenten benötigt. Neben den bereits erwähnten Conversion Rule Management und dem Trust Management spielt PKI eine Rolle. Das Management und die Überprüfung von Serverzertifikaten wird durch föderationsweite und interföderationsweite Public-Key-Infrastruktur vereinfacht. Auf Grund dieser bereits bestehenden Komponente, die nicht nur von FIM benötigt wird, werden ihr Aufbau und ihre Rollen nicht näher erläutert.

Die Implementierung der Provider Software soll, soweit möglich, als Art Adapter geschehen, um die darunter liegende Software möglichst einfach austauschen zu können. Wenn dies nicht möglich ist, wird eine Fassade zu MdFIM benötigt, um einerseits die Komplexität zur Benutzung zu vereinfachen und zum anderen um dieselben Befehle für alle möglichen Implementierungen verwenden zu können. Die SP-Software stellt das Pendant zur IdP-Software dar, um den Metadatenaustausch auf Seiten des SPs zu automatisieren.

Auf Grund der Erweiterung um den dynamischen Metadatenaustausch besitzt die IdP-Software folgende Funktionalitäten:

- Die Funktion der *Identifizierung und Authentifizierung von Nutzern*, die bereits in nativen Implementierungen verfügbar ist, wird auch durch die erweitere IdP-Software erfüllt. Die Anfragen zur Authentifizierung werden durch die SP-Software generiert. Im Gegensatz zur bisherigen Lösung, interagieren Identity Provider und Service Provider anfangs über MdFIM. Die IdP-Software überprüft, ob eine gültige Session für den Nutzer besteht. Die Gültigkeit dieser Sessions liegt meist zwischen wenigen Minuten und mehreren Stunden. Wenn für den Nutzer keine gültige Session existiert, wird er an einen IdP-lokalen Dienst zur Authentifizierung weitergeleitet (vgl. [SEC-Authentifizierung]). Anhand der ARPs wird überprüft, ob und welche Informationen an den SP bzw. MdFIM gesendet werden. Die ARPs müssen soweit angepasst sein, dass MdFIM die erste Authentifizierungsbestätigung erhält (vgl. [DSA-ARPs]).

- Im nächsten Schritt werden dem SP nach der Authentifizierungsbestätigung *Autorisierungsbestätigung* und *Attributsauskünfte* ausgestellt (vgl. [FA-Pull&Push] und [FA-Datenkategorisierung]). Die Attributsauskünfte werden über Schnittstellen zum lokalen Datenbestand erteilt. Durch ARPs, bei Shibboleth Attribut Filter genannt, werden die für den Service Provider bestimmten und erlaubten Attribute gefiltert ([DSA-ARPs]). Zusätzlich muss zur Unterstützung der Anforderung [DSA-Selbstbestimmung] protokolliert werden, welche Attribute wann an welchen SP gesendet wurden. Dies kann über die Erweiterung uApprove bzw. uApprove.jp geschehen. Hierbei kann auch die Zustimmung zu *Nutzungsrichtlinien* eingeholt werden. Die Benutzerinformationen sollen aktuell sein ([FA-Aktualisierung]), jedoch ist es Stand 2016 nicht vorgesehen die Service Provider über geänderte Attribute zu informieren. Falls die benötigten Konvertierungsregeln nicht vorhanden sind, sollen diese über das Conversion Rule Ma-

nagement verfügbar gemacht werden.

- Zusätzlich zu den nativ vorhandenen Funktionalitäten, soll der *Metadatenaustausch* automatisiert werden. Die dynamisch ausgetauschten Metadaten ([FA-Dynamik]) der vertrauenswürdigen SPs müssen automatisch in die lokale Konfiguration eingebunden werden ([FA-Automatisierung]), damit die zukünftige Kommunikation ohne Hilfe von MdFIM geschehen kann. Die Liste der lokal integrierten Metadaten entspricht somit der TAL, die die Basis für IdMRep und Dynamic Identity Federation (vgl. Abschnitt 3.4) bildet und entsprechend um dynamische Methoden erweitert werden kann. Um die Metadaten automatisch zu integrieren, müssen die Metadaten in einen Ordner gelegt werden, wo alle vertrauenswürdigen Metadaten liegen und der in der Konfiguration eingebunden ist. Der genaue Metadatenaustausch wird in der Realisierung der Kommunikation nachfolgend besprochen.

- Zu den zusätzlichen Funktionen zählt die Beantragung von Mitgliedschaften in offiziellen *Föderation*, die beispielsweise heutige nationale Föderationen darstellen können. Die Policies der Föderationen müssen transparent und für die interessierten Identity Provider lesbar sein. Nach den durch die Föderation vorgegebenen Gesichtspunkten wird die Mitgliedschaft erlaubt oder verweigert. Wenn die Föderation z. B. den IdP auditiert und dies durch ein entsprechendes Zertifikat oder einem Vermerk in den Metadaten sichtbar ist, kann dies förderlich für den Vertrauensaufbau zwischen Identity Provider und Service Provider sein.

Die SP-Software verfügt zudem über die folgenden beiden Eigenschaften:

- Die bisherigen Funktionalitäten, wie das Einholen der Authentifizierungsbestätigung, der Autorisieriungsbestätigung und der Attributsinformationen, bleiben bestehen.

- Zudem kann die SP-Software Metadaten von IdPs automatisch abfragen und integrieren. Dies geschieht ebenfalls über einen Ort, der in der Konfiguration genannt ist und wo alle Metadaten gespeichert werden.

Wie bereits durch Wolfgang Hommel [Hom07] aufgezeigt (u. a. Abschnitte 2.2 und 7.3.2), sind Konfiguration und Accounting essentielle Managementschnittstellen für die SP-Software. Zusätzlich zu den üblichen Schnittstellen zur Konfiguration, Protokollierung, Fehleranalyse und Leistungsüberwachung ist das Accounting wichtig, da gegebenenfalls die Anzahl sowie die Art und der Umfang der FIM-basierten Zugriffe auf einen Dienst mit dem jeweiligen SP abgerechnet werden muss. Hierfür ist die Granularität der zu erfassenden Daten festzulegen.

Ferner kommen Test-Entitäten, d. h. ein IdP und ein SP, in Frage, damit neu registrierte Entitäten überprüfen können, ob ihre Installation und Konfiguration funktioniert. Außerdem werden ein Service Desk und Monitoring-Tools benötigt. Da für die durchgängige Unterstützung organisationsübergreifender Prozesse die bisherige FIM Software nicht ausreicht, werden organisationsübergreifende Abläufe über MdFIM verwaltet. Diese zentrale Komponente bietet ferner weitere Funktionalitäten, wie Trust Management und Conversion Rule

Management, die aktuell nicht vorhanden sind. Im folgenden Abschnitt wird MdFIM näher beschrieben.

5.2. Managementplattform MdFIM

MdFIM realisiert die Verwaltung für die Managementarchitektur, welche im vorherigen Kapitel anhand von Organisationsmodell, Informationsmodell, Kommunikationsmodell und Funktionsmodell modelliert wurde. Die Trusted Third Party ist hingegen für den Metadatenaustausch zuständig. In diesem Abschnitt wird dieses Werkzeug und zugleich wichtigste Komponente des Konzepts näher beschrieben. Zunächst wird eine Übersicht über den Dienst anhand des MNM-Dienstmodells erstellt. Anschließend wird, basierend auf dem Kommunikationsmodell, die Kommunikation zwischen den beteiligten Entitäten modelliert. Hierfür wird ein passendes Architekturmuster ausgewählt, um die Metadaten dynamisch auszutauschen. Dieser Austausch und weitere benötigte Kommunikationen werden anhand von Workflows erklärt, bevor sie als Protokoll spezifiziert werden. Das Informationsmodell mündet in die Spezifikation der Datenhaltung und einer allgemein gültigen API, die anschließend beschrieben wird. Das Organisationsmodell bildet die Grundlage für ein Rollenkonzept, welches ebenfalls erörtert wird. Das Funktionsmodell wird ausgebaut, indem einzelne Funktionalitäten der MdFIM ausführlich beschrieben werden.

Die möglichen Architekturmuster, die Auswahl, das Grundkonzept der dynamischen virtuellen Föderationen und das Konzept für FIM über eine TTP wurden in der von der Autorin mitbetreuten Masterarbeit von Michael Grabatin [Gra14] veröffentlicht. Die vorliegende Arbeit geht darüber hinaus und begründet die Auswahl zudem mit den MNM-Modellen. Darauf aufbauend wird eine Managementplattform MdFIM gebildet, die weitere Funktionalitäten für das Management von dynamischen Föderation bietet. Die dynamischen virtuellen Föderationen werden anhand der Klassifikation von Föderationen genau definiert. Im GÉANT3plus OpenCall Projekt wurde durch Stefan Metzger und der Autorin FIM über eine TTP konzipiert. Das Konzept wurde u. a. durch [PMH14b], [PMH14a], [PMH14g] und [PMH14c] veröffentlicht. Die Autorin war dabei maßgeblich bei der Entwicklung beteiligt. Im Gegensatz zum Projekt geht diese Arbeit einer allgemein gültigen TTPs aus, die nicht nur für das Projekt GÉANT geeignet ist, sondern zusätzliche Anforderungen aus Föderationen und Communities außerhalb des Projektes betrachten. Die TTP ist die Managementplattformen mit Namen MdFIM einer Managementarchitektur, basierend auf Organisationsmodell, Informationsmodell, Kommunikationsmodell und Funktionsmodell, und stellt daher zusätzliche Funktionalitäten, wie ein Trust Management, zur Verfügung.

5.2.1. Übersicht über den Dienst MdFIM

Um eine allgemeine Übersicht über MdFIM zu erlangen, wird die Dienstsicht des MNM-Teams auf MdFIM angewandt. Um die Betreiber-Seite von MdFIM darzustellen, wird eine

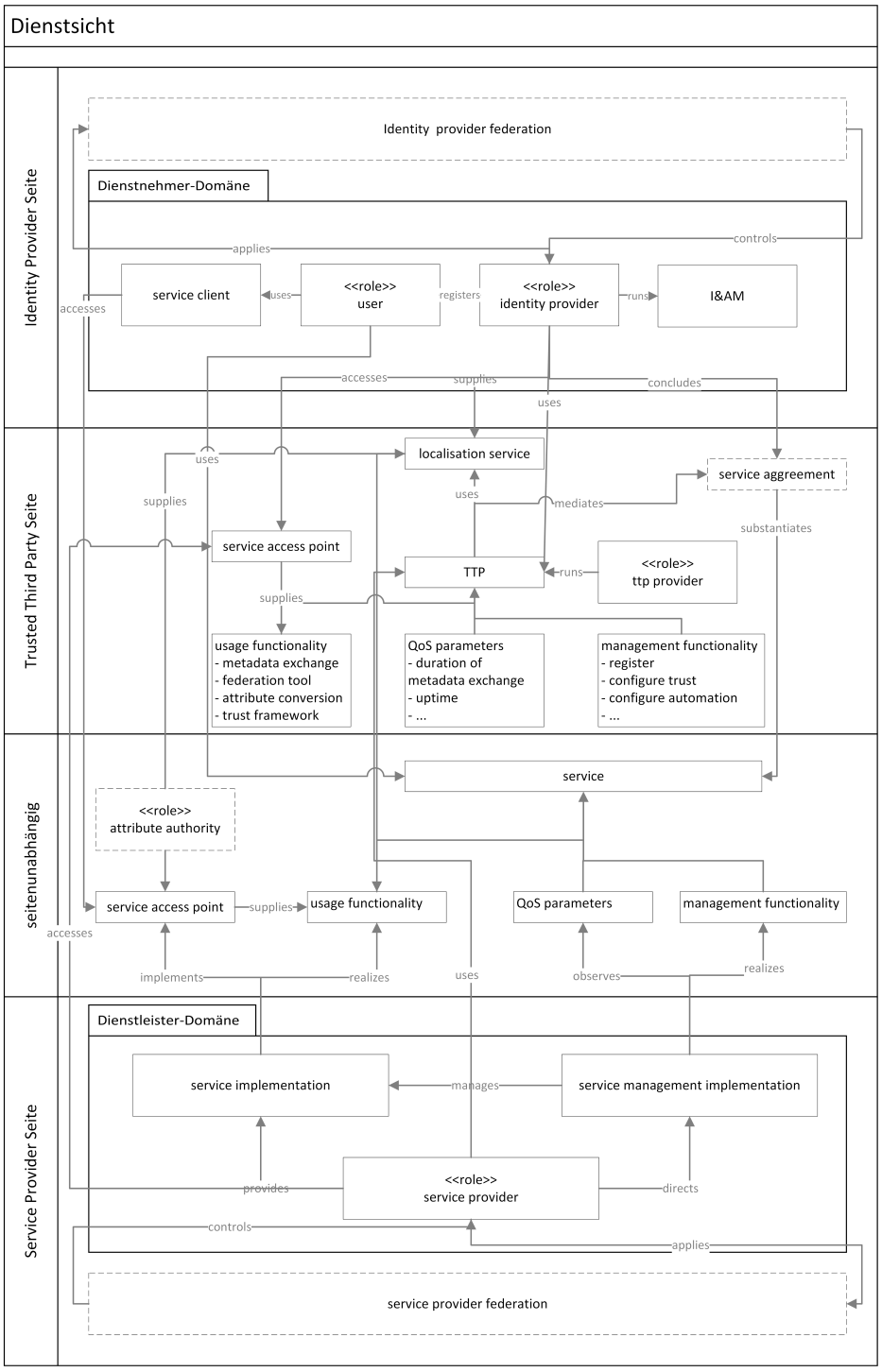

Abbildung 5.3.: FIM-Dienstsicht für die Architektur

weitere Seite eingefügt: *Trusted Third Party Seite*. Entsprechend der Provider wird Dienstnehmerseite in *Identity Provider Seite* und die Dienstleisterseite in *Service Provider Seite* umbenannt. Daraus ergibt sich folgendes Bild (vgl. 5.3):

Identity Provider Seite: Auf der Kundenseite befindet sich neben dem eigentlichen Nutzer auch der Kunde, der entweder Service Provider oder Identity Provider ist. Die jeweilige Entität kann einer Kunden-Föderation angehören. Der Nutzer möchte einen Dienst verwenden. Er greift über den service client, in der Regel ein Webbrowser, auf den Dienst zu. Der SP dieses Dienstes und der IdP, in dessen lokalen I&AM die Benutzerinformationen gespeichert sind, haben noch keine Metadaten miteinander ausgetauscht, weswegen der Dienst TTP verwendet werden soll.

Seitenunabhängig: Weiterhin seitenunabhängig ist der Dienst des SPs.

Trusted Third Party Seite: Die größte Änderung ergibt sich in dieser Domäne, da nun die TTP als Dienst für dynamischen, automatischen Metadatenaustausch und die notwendige Verwaltung angesehen wird:

- Der Nutzer stößt durch den erweiterten Lokalisierungsdienst den Metadatenaustausch an. Dieser Lokalisierungsdienst unterstützt den externen service access point des SPs.

- Da SPs und IdPs den TTP aktiv nutzen, haben sie Zugriff zur Management-Funktionalität. Die TTP Management-Funktionalität enthält Informationen über vorhandene Verbindungen, welche für weitere Statistiken und Statusreports verwendet werden kann. Da die Entitäten ihre Daten verwalten sollen, ist diese Management-Funktionalität Teil der Service View.

- Die Nutzungsfunktionalität besteht vor allem auf der Initiierung des Metadatenaustausches und der aktuelle Status des Vertrauensaufbaus. Aus Kundensicht ist die Verwaltung der Metadaten eine Hauptfunktionalität.

- Die wichtigen Quality of Service (QoS) Parameter sind Verfügbarkeit, Erreichbarkeit und Zeit, die für den Metadatenaustausch benötigt wird.

- Während der Endnutzer den service access point verwendet, der den Nutzer zum Lokalisierungsdienst weiterleitet, nutzen die Entitäten ein Web-Frontend bzw. die Erweiterung ihrer lokaler Software, um mit der TTP zu kommunizieren. Die Erweiterung der lokalen Software unterstützt zudem die Automatisierung bisher manueller Workflows.

- Nachdem alle Details zum Dienst spezifiziert wurden, kann das service agreement dargestellt werden. Da die TTP voraussichtlich von Föderationen, Inter-Föderationen oder anderen vertrauenswürdigen Organisationen betrieben wird, die keine weiteren finanziellen Forderungen stellen, wird das service agreement

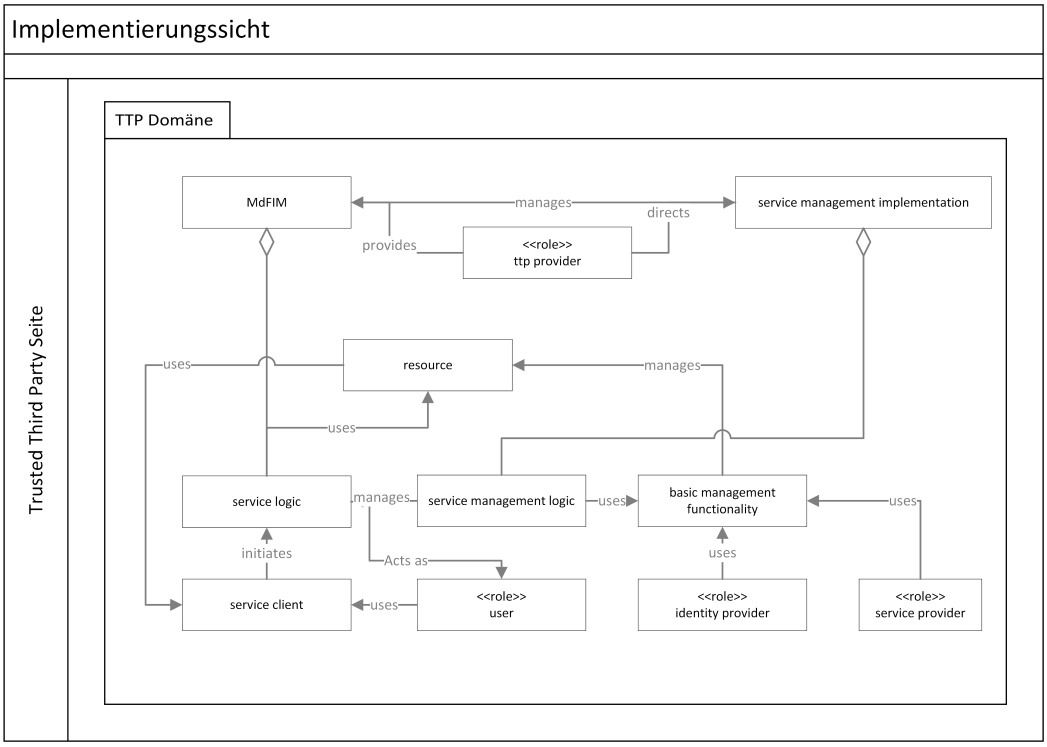

Abbildung 5.4.: FIM-Implementierungssicht für die Architektur

voraussichtlich nicht schriftlich erfolgen. Ist die Benutzung der TTP kostenpflichtig, sind jedoch SLAs notwendig.

Service Provider Seite: Der Service Provider der TTP wird, wie eben beschrieben, eine Föderation, eine Inter-Föderation oder eine weitere vertrauenswürdige Organisation sein. Folglich gibt es eine Verbindung zwischen Service Provider, TTP und einer SP Föderation. Der Betreiber der SP bietet sowohl Implementierung des Dienstes als auch die Management Implementierung für seine Kunden an.

Im Gegensatz zur allgemeinen *Implementierungssicht* für einen Dienst, der über FIM angeboten wird, ist in dieser Sicht die direkte Anwendung auf die TTP MdFIM in der TTP Domäne sinnvoll (vgl. Abbildung 5.4). Die sub-service logic entfällt, dafür initiiert der service client die service logic, also den Austausch der Metadaten sowie weitere Funktionalitäten, wie die Konvertierung von Benutzerinformationen. Die service logic agiert somit als Nutzer. Die service logic wird von der service management logic verwaltet. Die service management logic verwendet wiederum die basic management functionality, die von IdP und SP eingesetzt wird. Die Rolle *ttp provider* realisiert dabei MdFIM und lenkt die service management implementation.

Das Management lässt sich mit den folgenden Modellen auf die zu entwickelnde Lösung übertragen:

- Informationsmodell,
- Organisationsmodell,
- Kommunikationsmodell und
- Funktionsmodell.

Die Modelle werden nachfolgend für MdFIM realisiert und genauer spezifiziert.

5.2.2. Realisierung der Kommunikation

Wie im Kommunikationsmodell in Abschnitt 4.5 zu sehen, werden für das Management von Föderationen verschiedene Protokollinteraktionen verwendet. Während `discover` über einen Lokalisierungsdienst (vgl. Kapitel 3) realisiert werden kann, werden für die weiteren Protokollinteraktionen Workflows bzw. eine API benötigt. `get`, `set`, `query` sind hierbei typische Protokollinteraktionen, die in einer API zum Einsatz kommen. Ebenso können `post`, `create`, `delete` und `update` für die API verwendet werden. `notify` ermöglicht es über MdFIM andere Teilnehmer zu benachrichtigen. `register` ist eine Funktionalität von MdFIM, damit Entitäten und Föderationen sich registrieren können. Über diese Protokollinteraktionen sind u. a. diese Funktionalitäten möglich:

- Unterstützung von losen wie auch festen Strukturen auf dem Interaktionskanal,
- Protokollierung der durchgeführten Aktionen und der Aktivitäten auf dem Interaktionskanal für Auditierung,
- Publizieren und Finden von Schnittstellen der Interaktionskanäle,
- Durchsetzen von Zugangsbeschränkungen zu Interaktionskanäle und
- Austausch von Metadaten, Vertrauensinformationen und Konvertierungsregeln.

Die Unterstützung von losen wie auch festen Strukturen ist durch eine einheitliche API möglich. Diese API ist allgemein bekannt, jedoch nur authentifiziert durch einen API-Key nutzbar. Die Zugangsbeschränkungen werden über die Realisierung des Organisationsmodells, der damit vorhandenen Rollen und Berechtigungen sowie der Authentifizierung eingehalten. Die Protokollierung der durchgeführten Aktionen wird durch die Managementarchitektur gewährleistet. Hierfür können ebenfalls on-board Möglichkeiten der Server und der verwendeten Software, wie Datenbank, eingesetzt werden. Historientabellen, d. h. Tabellen über die historischen Stände, protokollieren ebenfalls Veränderungen. Nachfolgend

wird insbesondere auf den Austausch der Metadaten, Vertrauensinformationen und Konvertierungsregeln eingegangen, nachdem hierfür Nachrichten zwischen Entitäten und der Managementarchitektur notwendig sind.

Als Grundlage für den Nachrichtenaustausch muss ein Architekturmuster gewählt werden. Anschließend werden, um einen genau festgelegten Ablauf von Nachrichten und Aktionen für die zu entwickelnden Protokolle und der daran anschließenden prototypischen Implementierung zu erhalten, mögliche Workflows und ihre Varianten beschrieben werden. Die Workflows sind in Metadatenaustausch und unterstützende Workflows unterteilt und wurden in den Veröffentlichungen [PMH14b] und [PMH14c] partiell vorgestellt. Die Workflows zum Metadatenaustausch beschreiben den Austausch der Metadaten und somit den technischen Vertrauensaufbau. Die unterstützenden Workflows zeigen wiederum die notwendigen Schritte auf, um Metadaten, Trust, Konvertierungsregeln und Policies überhaupt verwalten zu können, bevor der Vertrauensaufbau stattfinden kann. Die Workflows werden anschließend jeweils um die Realisierung als Protokoll wiedergegeben. Abschließend wird die Kommunikation basierend auf der Interaktion des Organisationsmodells als Domänen dargestellt.

Architekturmuster

Um den Metadatenaustausch und somit die Föderationen automatisch und somit dynamisch zu gestalten (zweiter Aspekt der Anforderung [FA-Dynamik]), sollen Nutzer den Vertrauensaufbau initiieren können [FA-Initiierung]. Die Lösung soll alle potentiellen Nutzer erreichen [FA-Reichweite] und somit auch grenzüberschreitende Kooperationen ermöglichen [FA-Grenzüberschreitend]. Dazu müssen bedarfsgerechte Föderationen aufgebaut werden [FA-Föderation], während es möglich sein muss in mehreren Föderationen gleichzeitig teilzunehmen [NFA-Koexistenz]. Um die Wartezeit für Nutzer sowie den Aufwand für die Administrationen zu reduzieren, wird eine Automatisierung bestehender Prozesse und Workflows angestrebt [FA-Automatisierung]. Diese Automatisierung ist hierbei essentiell um die Anforderung [FA-Dynamik] zu erfüllen. Dynamisch bedeutet hierbei den automatischen Austausch von Metadaten, Konvertierungsregeln und folgende Anpassung der Konfiguration, um eine bilaterale Föderation zwischen IdP und SP zu ermöglichen, wenn ein Nutzer des IdPs einen SP verwenden will. Um größere, sich wandelnde Kooperationen zu beschreiben, lassen sich dynamisch Föderationen und Inter-Föderationen bilden (erster Aspekt der Anforderung [FA-Dynamik]). Wie bereits Bertino et al. [BFS04] beschrieben haben, bedeutet Trust negotiation, dass zwei Entitäten, die noch keine direkte Vertrauensbeziehung haben, Vertrauen aushandeln, indem sie Geheimnisse austauschen. Übersetzt in SAML werden Metadaten zwischen den Entitäten ausgetauscht. Dazu wird laut [BFS04] Vertrauen in eine gemeinsame dritte Entität vorausgesetzt.

Hierfür muss die bestehende Architektur der Föderationen im R&E-Umfeld angepasst und erweitert werden. Nachfolgend werden mögliche Architekturmuster, die zentrale und dezentrale Lösungen ermöglichen, aufgezeigt, wie sie Michael Grabatin in seiner Masterarbeit bereits beschrieben hat [Gra14]. Alle basieren auf dem Web Browser SSO Profile in

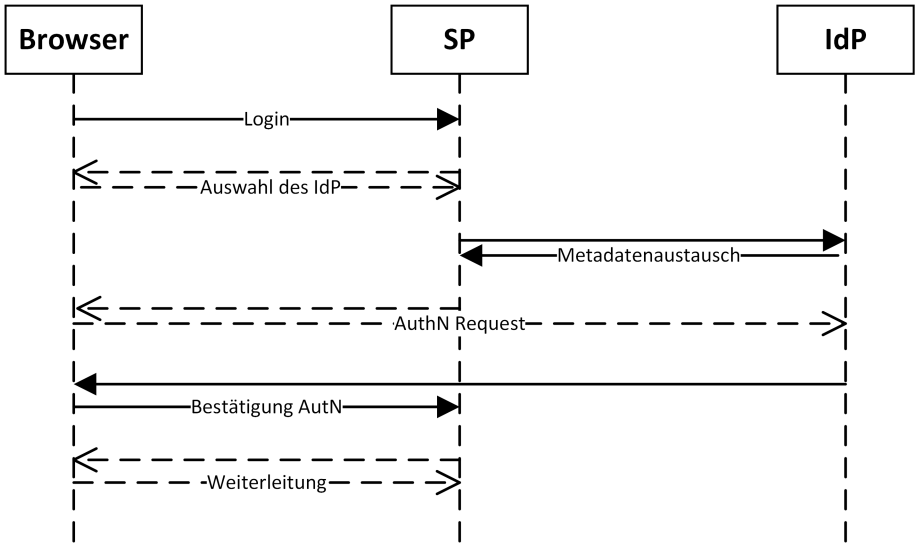

Abbildung 5.5.: Workflow bei einem direkten bidirektionalen Metadatenaustausch

Kombination mit dem Discovery Service. Damit wird u. a. die Kompatibilität mit bisherigen Systemen sichergestellt [FA-Integration]. Ferner können Tools wie uApprove für die Abfrage des User Consents (vgl. Abschnitt 3.2) eingebunden werden, um die Zustimmung zu realisieren [DSA-Zustimmung]. Anschließend wird daraus eine passende Architektur gewählt, die möglichst dynamische Zusammenschlüsse ermöglichen soll.

Erweiterung durch direkten bidirektionalen Metadatenaustausch

In den aktuellen Föderationen werden die Metadaten vorab statisch und aggregiert ausgetauscht. Ein alternativer Ansatz ist diesen Metadatenaustausch bidirektional bei Bedarf anzustoßen, ohne dass weitere Elemente in die Architektur hinzugefügt werden. Dazu wird eine Art globaler Lokalisierungsdienst benötigt, der alle Identity Provider kennt oder zumindest die Möglichkeit bietet weitere IdPs hinzuzufügen. Alternativ können verschiedene Lokalisierungsdienste ihre Informationen äquivalent zu Peer-to-Peer austauschen. Der grundlegende Workflow sieht wie folgt aus (vgl. Abbildung 5.5):

Schritt 1: Der Nutzer wählt durch den Lokalisierungsdienst seinen IdP aus.

Schritt 2: Der SP erhält durch das Discovery Protocol die Auswahl des Nutzers und überprüft, ob die Metadaten des IdPs bereits lokal vorhanden sind.

Schritt 3: Falls dies nicht der Fall ist, wird durch eine Erweiterung der bidirektionale Metadatenaustausch angestoßen. Dieser kann über das Metadata Query Protocol geschehen, welches in Abschnitt 3.3 vorgestellt wurde. Je nach verwendeter Implementierung und Konfiguration kann sich die URL, unter der die Metadaten liegen, unterscheiden. Folglich ist eine einheitliche Systematik oder eine Art globales Verzeichnis wichtig.

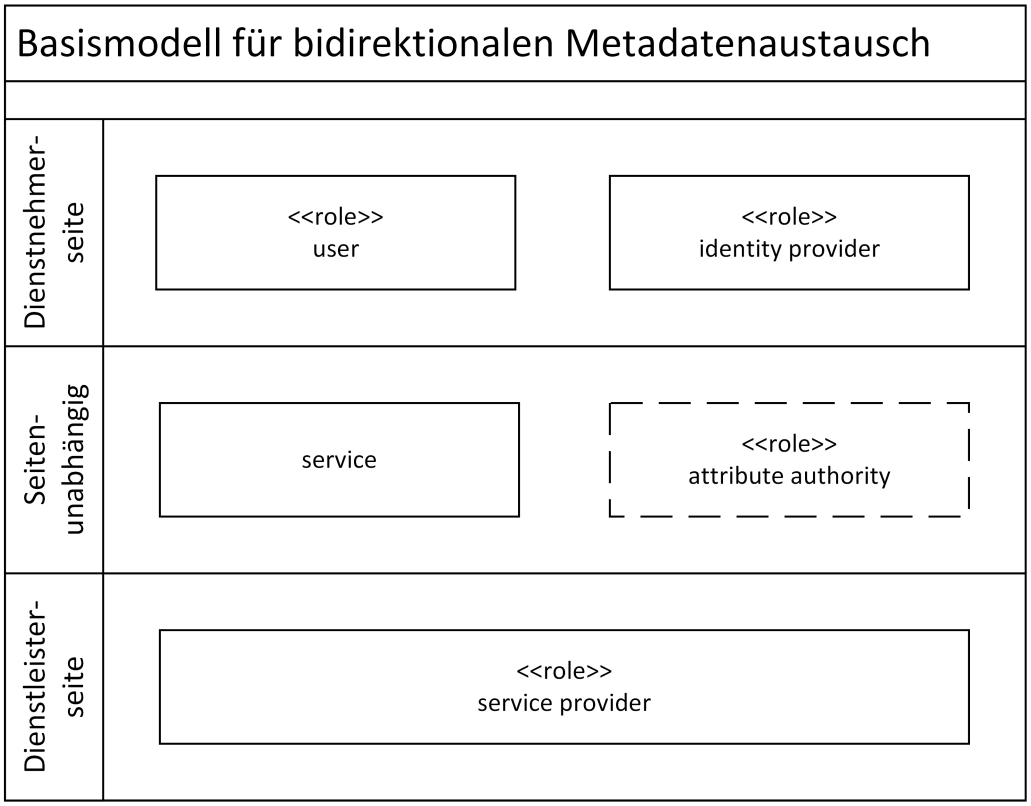

Abbildung 5.6.: FIM-Basismodell für einen dezentralen Metadatenaustausch

Schritt 4: Wenn sich der Nutzer nicht erfolgreich innerhalb einer vorher festgelegten Zeitspanne authentifiziert hat, muss der Metadatenaustausch zurück gerollt werden.

Bei diesem Ansatz werden Metadatenverwaltungen überflüssig, da die Informationen on demand und ohne einer TTP ausgetauscht werden. Zur Berechnung des dynamischen Vertrauens können Ansätze wie IdMRep, Abschnitt 3.4, oder Trust-Based Access Control von Latifa Boursas [Bou09] verwendet werden.

Im Abschnitt 2.2.4 wurde bereits das FIM-Dienstmodell, basierend auf Hegering et al. [HAN99] und Garschhammer [GHH+01] [GHK+01] beschrieben. Dieses Modell lässt sich, äquivalent zu den Szenarien, auf die Architekturmuster anwenden, was zu folgender Ausprägung (vgl. Abbildung 5.6) im *Basismodell* führt:

- Föderationen sind beim dezentralen Metadatenaustausch unbedeutend und daher nicht im Basismodell vertreten.
- Es gibt weder eine dezentrale noch eine zentrale Komponente, um die Metadaten auszutauschen und sich zu registrieren.

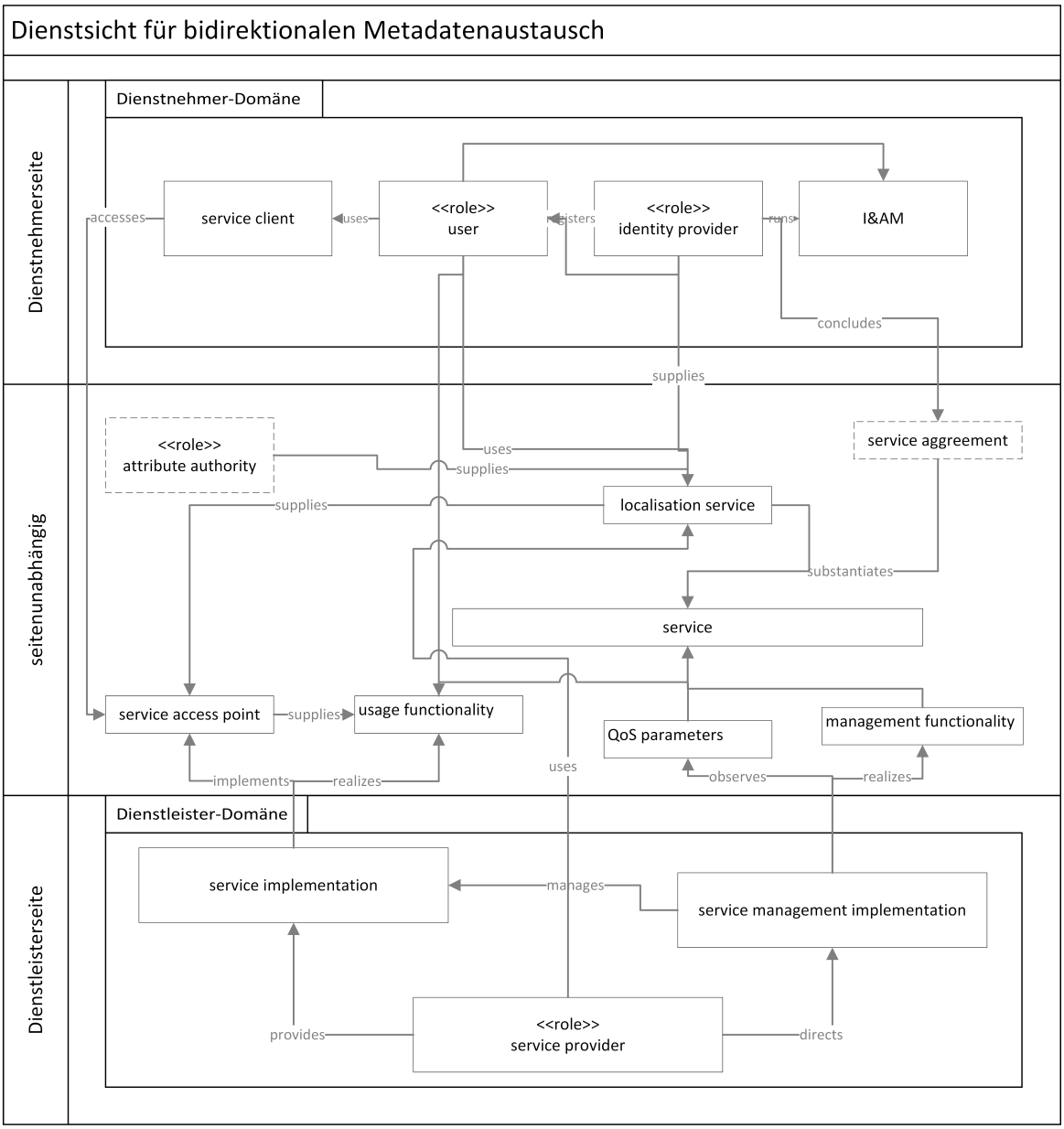

Abbildung 5.7.: FIM-Dienstsicht für einen dezentralen Metadatenaustausch

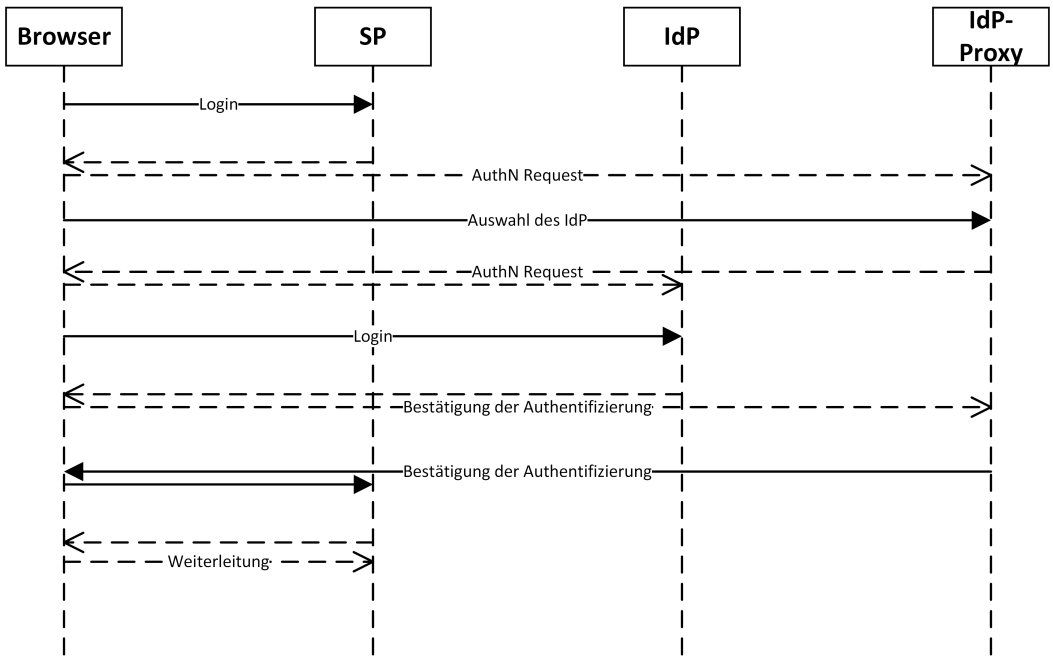

Abbildung 5.8.: Workflow bei Verwendung eines IdP-Proxys

Diese Änderungen spiegeln sich in der Dienstsicht (vgl. Abbildung 5.7) wieder. Sowohl auf Dienstnehmerseite als auch auf Dienstleisterseite befinden sich keine Föderationen mehr. Seitenunabhängig kommen alle Komponenten vor, die direkt zum Service gehören. Zudem kann eine AA sowie ein Service Agreement hier lokalisiert sein. Folglich existiert eine schlanke Dienstsicht ohne übergeordneten Organisationen. Damit dies funktioniert, muss der Lokalisierungsdienst dementsprechend angepasst werden.

IdP-Proxy

Ein weiterer möglicher Ansatz ist die Verwendung eines IdP-Proxies, wie in Abschnitt 3.3 beschrieben. Ein SAML IdP Proxy agiert als Bridge oder Gateway zwischen den IdPs und SPs einer Föderation [CS12] [Lin09]. Der genaue Ablauf ist wie folgt, vergleiche Abbildung 5.8:

Schritt 1: Ein Nutzer möchte den Dienst eines SPs nutzen, der sich hinter einem IdP-Proxy befindet.

Schritt 2: Der Client wird zur IdP-Komponente des Proxys weiter geleitet.

Schritt 3: Der Client schickt einen AuthnRequest an den IdP des Proxys.

Schritt 4: Der AuthnRequest wird zwischengespeichert und der Nutzer wird zu seinem IdP

weiter geleitet.

Schritt 5: Der Client schickt einen AuthnRequest an den IdP des Nutzers.

Schritt 6: Der IdP des Nutzers aktualisiert seinen Security Context und sendet eine Response an den Client.

Schritt 7: Der Client übermittelt den Response an den Assertion Consumer Service der SP-Komponente beim Proxy. Dieser validiert die Assertions, die in der Response enthalten sind.

Schritt 8: Die SP-Komponente des Proxys aktualisiert daraufhin seinen Security Context und leitet den Client zur IdP-Komponente des Proxys weiter.

Schritt 9: Der Client erstellt einen AuthnRequest an den IdP des Proxys. Der AuthnRequest entspricht dem in Schritt 3.

Schritt 10: Der IdP des Proxys aktualisiert den Security Context, gibt eine einzelne Assertion heraus und schickt eine Antwort an den Client. Die Response kann dabei die Assertions aus Schritt 6 enthalten.

Schritt 11: Der Client schickt eine Response an den Assertion Consumer Service des ursprünglich angefragten SPs.

Schritt 12: Der SP aktualisiert daraufhin ebenfalls seinen Security Context und leitet den Client zum Dienst weiter.

Schritt 13: Der Client sendet erneut den Request aus Schritt 1.

Schritt 14: Der Dienst entscheidet über die Nutzung anhand des Security Contexts und sendet die Ressource bzw. die Antwort zurück an den Client.

Die erfolgreiche Authentifizierung beim IdP des Nutzers wird durch den IdP-Proxy verifiziert, bevor der Nutzer einen Dienst im geschützten Bereich verwenden kann. Der IdP-Proxy enthält somit, wie in Abschnitt 3.3 dargestellt, eine Art Metadatenverwaltung und kann, äquivalent zum Ansatz Dynamic Identity Federation (vgl. Abschnitt 3.4), zum Vertrauensaufbau genutzt werden.

Wenn mehrere Föderationen, die jeweils ihren eigenen IdP-Proxy betreiben, kooperieren, erfolgt die Kommunikation über die IdP-Proxys. Bei Inter-Föderationen kann es somit sein, dass sowohl Föderationen als auch Inter-Föderation einen IdP-Proxy besitzen. Alternativ kann ein IdP-Proxy für die gesamte Inter-Föderation vorhanden sein, was jedoch zu Lasten der Verfügbarkeit gehen kann. Folglich besitzt das Basismodell für die Verwendung von IdP-Proxys pro Föderation einen IdP-Proxy, wie in Abbildung 5.9 zu sehen. Die Änderungen sind wie folgt:

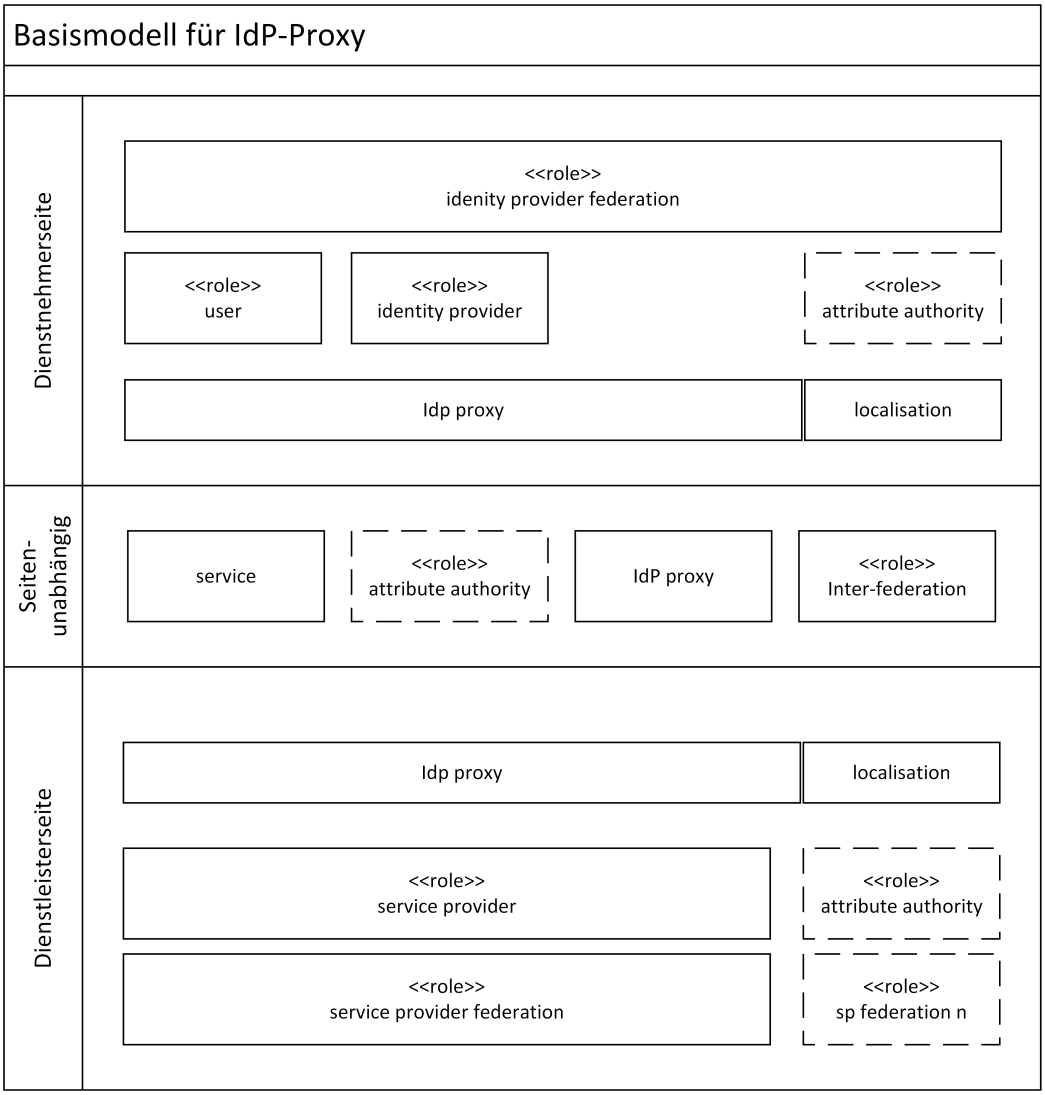

Abbildung 5.9.: FIM-Basismodell für die Verwendung von IdP-Proxys

- Sowohl IdP Föderation als auch SP Föderation betreiben einen IdP-Proxy.
- Die Inter-Föderation betreibt ebenfalls einen IdP-Proxy.
- Jeder IdP-Proxy hat, verkürzt in der Abbildung 5.9 dargestellt, eine Art Lokalisierungsdienst. Zudem ist eine Art Management Tool enthalten.

Um die Dienstsicht (vgl. Abbildung 5.10) bei Verwendung eines IdP-Proxys übersichtlicher zu gestalten, wurden die Rolle Inter-Föderation sowie der dazugehörige IdP-Proxy absichtlich weggelassen. Trotzdem zeigt sich, dass sowohl Dienstnehmerseite als auch Dienst-

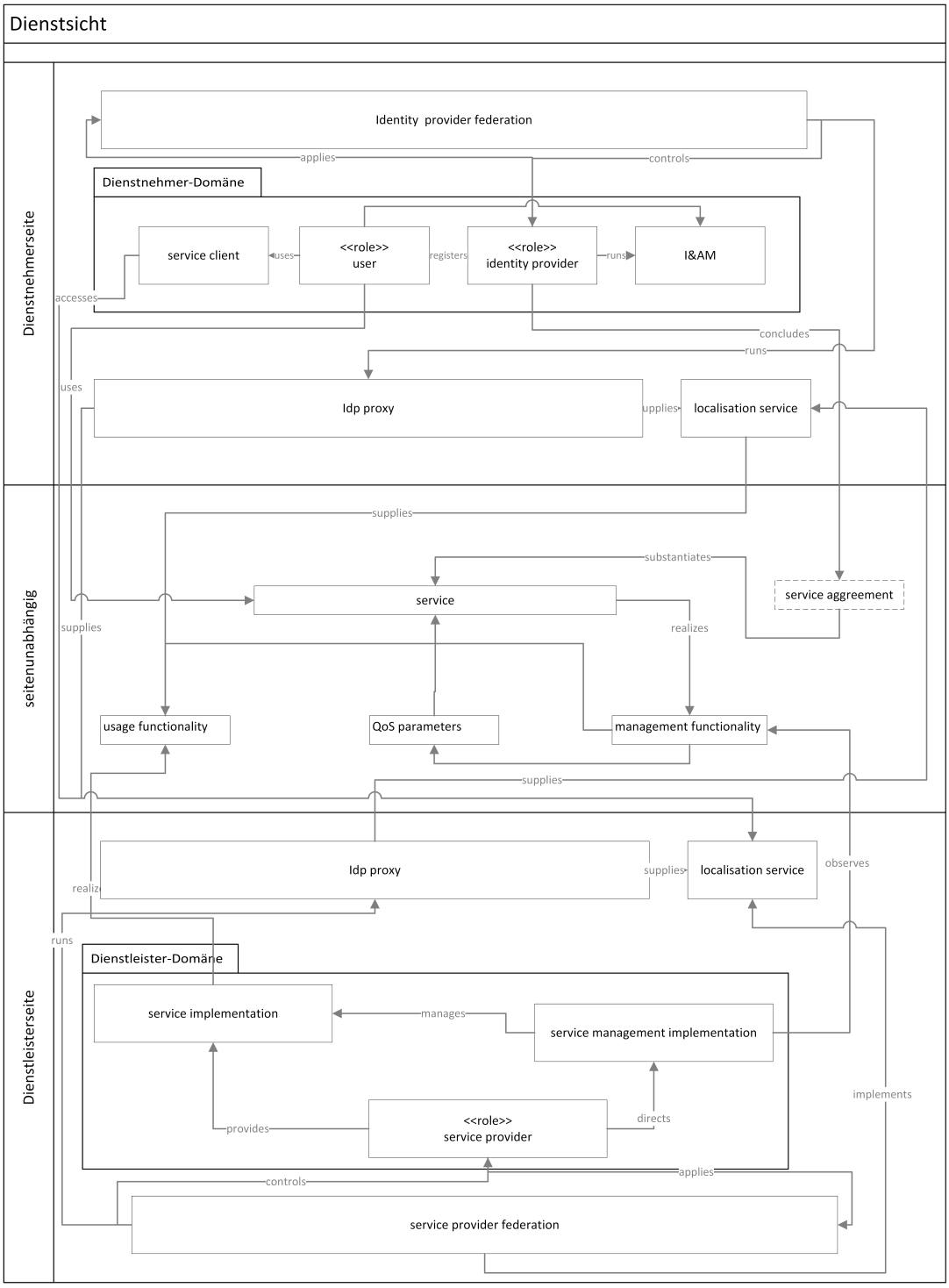

Abbildung 5.10.: FIM-Dienstsicht für die Verwendung von IdP-Proxys

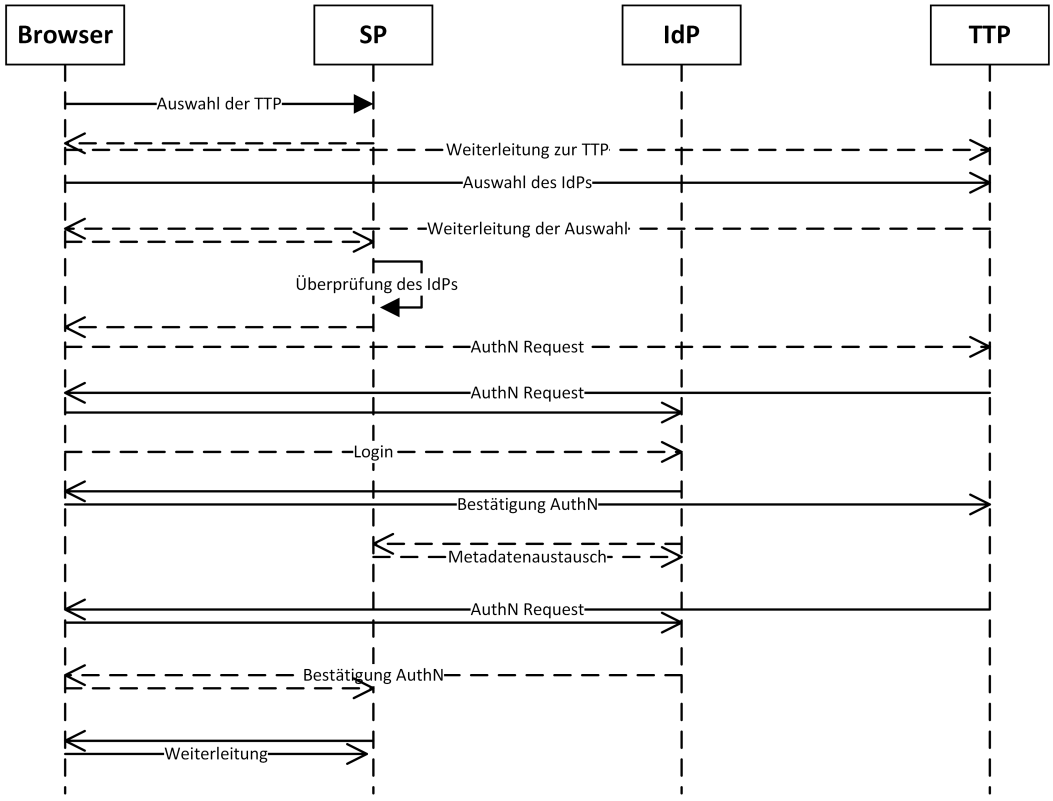

Abbildung 5.11.: Workflow bei Erweiterung des Discovery Service

leisterseite über mehr Komponenten verfügen. Die IdP-Proxys interagieren mit den Service-Komponenten, die seitenunabhängig sind, damit der Nutzer einen Dienst verwenden kann.

Erweiterung des Lokalisierungsdienstes

Ein weiterer Ansatz stellt die Erweiterung eines Lokalisierungsdienstes, z. B. des Centralized Discovery Services von Shibboleth, dar. Dies ist möglich, da der Lokalisierungsdienst Informationen zu beteiligten IdPs und SPs besitzt, wie in Abbildung 5.11 zu sehen:

Schritt 1: Der Nutzer möchte einen Dienst des SPs verwenden.

Schritt 2: Beim Lokalisierungsdienst wählt der Nutzer seinen IdP aus. Voraussetzung hierfür ist, dass der Lokalisierungsdienst über alle relevante Identity Provider Bescheid weiß. Alternativ muss der Nutzer die Möglichkeit haben seinen IdP hinzuzufügen. Der Austausch von Informationen zwischen miteinander vernetzten Lokalisierungsdiensten ist ebenfalls denkbar.

Schritt 3: Der Lokalisierungsdienst übermittelt die Auswahl über das Discovery Service Protocol an den SP.

Schritt 4: Nachdem der SP die Metadaten des gewählten IdPs nicht in seiner lokalen Konfiguration findet, sendet er einen Authentication Request an den IdP, der bei der Trusted Third Party Lokalisierungsdienst zwischengespeichert wird.

Schritt 5: Die TTP überprüft die Signatur des SPs, um die Authentizität sicher zu stellen.

Schritt 6: Zum Schutz vor Angriffen, muss sichergestellt sein, dass der Nutzer ein gültiges Benutzerkonto beim ausgewählten IdP besitzt. Dafür wird ein neuer Authentication Request an den IdP gesendet.

Schritt 7: Nun authentifiziert sich der Nutzer.

Schritt 8: Nach erfolgreicher Authentifizierung des Nutzers wird der Metadatenaustausch durch die TTP angestoßen. Dieser kann beispielsweise über das Metadata Query Protocol geschehen.

Schritt 9: Im Anschluss an den erfolgreichen Metadatenaustausch übermittelt die TTP den zuvor gespeicherten Authentication Request des SPs aus Schritt 4 an den IdP.

Schritt 10: Da der Nutzer erst erfolgreich authentifiziert wurde, existiert in der Regel eine gültige Sitzung.

Schritt 11: Darum leitet der IdP seinen Nutzer mit einer Authentication Response an den SP weiter.

Schritt 12: Der SP validiert die Response und der Nutzer kann anschließend den Dienst erfolgreich nutzen.

Für diesen Ansatz wird neben einer Erweiterung auf Seiten der Entitäten auch eine Erweiterung des Lokalisierungsdienstes benötigt. Im Gegensatz zum Ansatz DIMDS, beschrieben in Abschnitt 3.4, ist kein zentrales Identity Management System notwendig, indem jeder Nutzer ein Benutzerkonto anlegen muss. Es existieren weiterhin die lokalen Benutzerverwaltungen bei den Heimatorganisationen. Die TTP vermittelt das technische Vertrauen durch den Metadatenaustausch, ist verantwortlich für den anfänglichen Informationsaustausch und agiert somit als Broker. Das Basismodell lässt sich, äquivalent zu den Szenarien, auf die zu gestaltende Architektur anwenden (vgl. Abbildung 5.12). Somit befinden sich auf Dienstnehmerseite Nutzer, IdP und gegebenenfalls eine IdP Föderation, während es auf Dienstleisterseite einen Service Provider und eventuell eine SP Föderation gibt. Seitenunabhängig sind der Dienst selbst, mögliche Trusted Third Parties und die Trusted Third Party. Bei der *Dienstsicht* ist ebenfalls eine TTP eingefügt, vgl. Abbildung 5.3. Durch diese Komponente vereinfacht sich die Dienstbenutzung auch in dieser Sicht.

Auswahl des Architekturmusters

Im folgenden Abschnitt wird das Architekturmusters auf Grund der Anpassungen an den Standard-Workflow bei Webanwendungen von SAML sowie der Architektur ausgewählt.

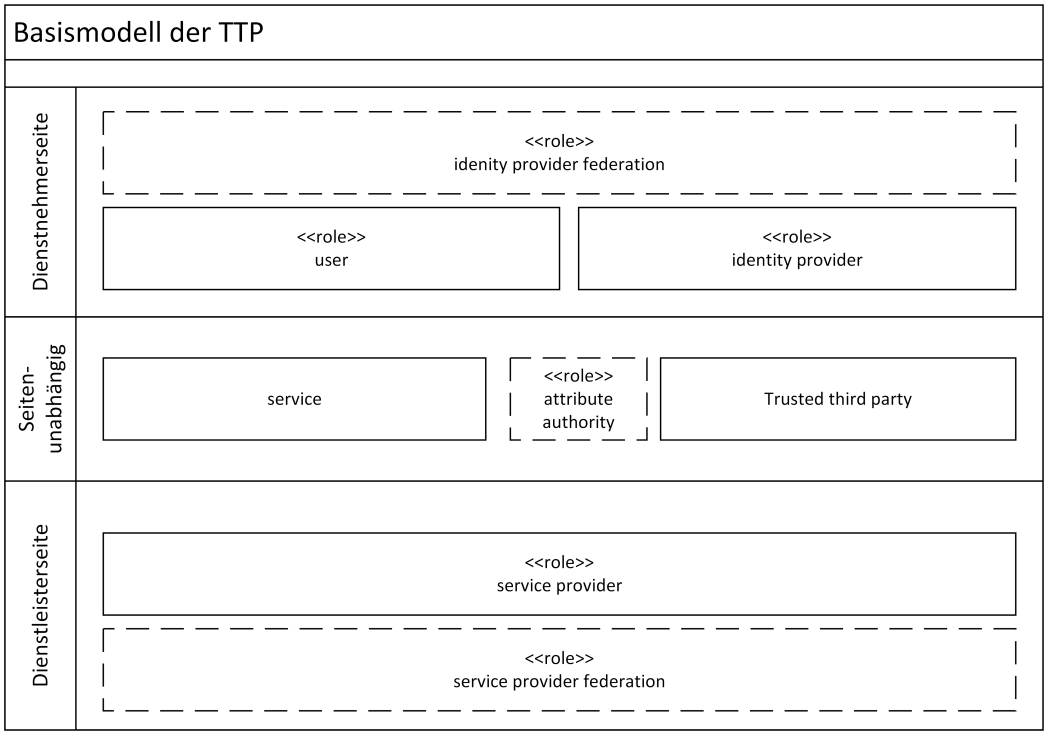

Abbildung 5.12.: Basismodell bei Erweiterung des Discovery Service

In der Tabelle 5.1 sind die Anpassungen dargestellt, die für jeden der drei Ansätze im Vergleich zum Standard-Workflow mit SAML Web Browser SSO [SAML2Prof] [HCH+05] und Discovery Service [SAMLIdPDisc] [LCWC08] notwendig sind.

Bidirektionaler Metadatenaustausch: Der dynamische, bidirektionale Metadatenaustausch, der in der Tabelle aufgezeigt wurde, ist mit SAML nicht so angedacht, jedoch mit einer entsprechenden Erweiterung möglich. Dies führt zu Erweiterungen der eingesetzten Software bei IdPs und SPs.

IdP-Proxy: Durch die zentrale Position des IdP-Proxys muss der Proxy alle Entitäten der angeschlossenen Systeme kennen. Da die Funktionalität über SAML bereits verfügbar ist, sind für Service Provider und Identity Provider folglich keine Veränderungen notwendig.

Erweiterung des Lokalisierungsdienstes: Eine zentrale Verwaltung der Entitäten bietet die Erweiterung des Lokalisierungsdienstes. Hierfür werden, ebenso wie beim dynamischen, bidirektionalen Metadatenaustausch, Erweiterungen benötigt. Diese Erweiterungen zielen auf Web Browser SSO Profile ab. Zudem muss das Identity Provider Profile angepasst werden.

SAML Web Browser SSO + Lokalisierungsdienst	Bidirektionaler Austausch	Erweiterung des Lokalisierungsdienstes	IdP-Proxy
Ausgangssituation			
IdP und SP sind Mitglied einer Föderation, Metadaten wurden vorab ausgetauscht	IdP und SP sind nicht Mitglied einer Föderation, Metadaten sind nicht bekannt	IdP und SP sind nicht Mitglied einer Föderation, Metadaten sind nicht ausgetauscht	Proxy bildet Bridge zwischen IdP und SP
Workflow			
Nutzer will Dienst beim SP nutzen	Nutzer will Dienst beim SP nutzen	Nutzer will Dienst beim SP nutzen	Nutzer will Dienst beim SP nutzen
Nutzer wird zum Lokalisierungsdienst weiter geleitet, um seinen IdP auszuwählen	Nutzer wird zum Lokalisierungsdienst weiter geleitet, um seinen IdP auszuwählen	Nutzer wird zum Lokalisierungsdienst weiter geleitet, um seinen IdP auszuwählen	Nutzer wird zum IdP-Proxy weitergeleitet
Nutzer wählt IdP und wird zum SP weiter geleitet	Nutzer wählt IdP und wird zum SP weiter geleitet	Nutzer wählt IdP und wird zum SP weiter geleitet	-
SP sendet SAML AuthRequest zum IdP	SP triggert Metadatenaustausch	SP sendet SAML AuthRequest zur TTP, welche diesen speichert	SAML AuthRequest zum IdP-Proxy, welcher diesen speichert
	IdP und SP tauschen Metadaten aus	TTP sendet 2. SAML AuthRequest zum IdP	SAML AuthRequest zum IdP des Nutzers
Authentifizierung		Authentifizierung	Authentifizierung
		IdP sendet SAML AuthResponse zur TTP	
		TTP triggert Metadatenaustausch mit IdP	
		TTP triggert Metadatenaustausch mit SP	
		TTP sendet gespeicherten AuthRequest des SPs zum IdP	
	SP sendet AuthRequest zum IdP		IdP sendet SAML AuthResponse zum IdP-Proxy, aktualisiert Security Context und leitet Nutzer zum IdP des Proxys
	Authentifizierung		Client sendet Antwort des SAML AuthRequest an den IdP des Proxys
IdP sendet SAML AuthResponse mit Assertion an SP und aktualisiert Security Context	IdP sendet SAML AuthResponse mit Assertion an SP und aktualisiert Security Context	IdP sendet SAML AuthResponse mit Assertion an SP und aktualisiert Security Context	IdP des Proxys sendet SAML AuthResponse mit Assertion an den SP
Nutzer kann Dienst verwenden	Nutzer kann Dienst verwenden	Nutzer kann Dienst verwenden	Nutzer kann Dienst verwenden
Ergebnis			
IdP und SP kommunizieren direkt miteinander	IdP und SP kommunizieren direkt miteinander, es gibt keine Föderationen	TTP initiiert Metadatenaustausch, ist nur für Metadatenaustausch zuständig	IdP-Proxy speichert alle Requests und leitet sie weiter

Tabelle 5.1.: Gegenüberstellung von Standard-SAML-Workflow, bidirektionalem Austausch, Erweiterung des Lokalisierungsdienstes und IdP-Proxy

Als weiterer Aspekt wird die Architektur auf Grund des Basismodells und der Dienstsicht betrachtet.

Bidirektionaler Metadatenaustausch: Der bi-direktionale Metadatenaustausch enthält wenige Komponenten und verfügt somit über eine schlanke Dienstsicht. Diese Architektur bietet eine gute Skalierbarkeit bezüglich System und Metadaten, und lässt sich dynamisch erweitern. Zudem ist der Metadatenaustausch dynamisch möglich. Nachdem weder Föderationen noch Inter-Föderationen mit ihren Management Tools existieren, werden keine Daten unnötig redundant vorgehalten. Ein Defizit dieses Ansatzes ist die fehlende organisatorische Kontrollmöglichkeit, vergleiche Anforderungen [ORG-Validierung] und [ORG-Registrierung], die aktuell durch Föderationsverwaltungen durchgeführt wird und für eine gewisse Qualität sorgt.

IdP-Proxy: Bei dem Einsatz von IdP-Proxys zeigt sich, dass sowohl Basismodell als auch Dienstsicht viele Komponenten besitzen. Die eingesetzten IdP-Proxys müssen alle Teilnehmer ihrer Föderation bzw. Inter-Föderation wissen. Dadurch kann sich der Proxy als Single Point of Failure erweisen, insbesondere da er für jede einzelne Authentifizierung benötigt wird. Dies hat Auswirkungen auf die Performanz [NFA-Performanz]. Gleichzeitig sind Skalierbarkeit bezüglich des Systems und Fehlertoleranz begrenzt. Da die Skalierbarkeit, [NFA-Skalierbarkeit], ein Defizit der aktuellen Föderationen und Inter-Föderationen darstellt, ist dieser Ansatz als suboptimal zu betrachten.

Erweiterung des Lokalisierungsdienstes: Im Gegensatz zum bi-direktionalem Metadatenaustausch sind bei der Erweiterung des Lokalisierungsdienstes mehr Komponenten involviert. Zudem sind sowohl auf Dienstnehmerseite als auch auf Dienstleisterseite Föderationen, die ihre SPs bzw. IdPs kontrollieren, was für ein gewisses Qualitätsmanagement sorgt. Die zentrale TTP kann zur Validierung und Registrierung verwendet werden, aber auch eine einheitliche Schnittstelle für weiteres Qualitätsmanagement darstellen. Beim Vergleich mit den Szenarien im Kapitel 2 zeigt sich, dass anstelle von mehreren Management Tools und somit Schnittstellen nur ein zentrales Tool, die TTP, existiert.

Um die geeignete Architektur auszuwählen, werden zudem die Anforderungen aus Kapitel 2 betrachtet. Die Bewertung erfolgt analog zu Kapitel 3 mit:

- +: Anforderung vollständig erfüllt.
- o: Anforderung teilweise erfüllt bzw. hängt von der Implementierung ab.
- -: Anforderung nicht erfüllt.

Zur Verdeutlichung der Differenzen werden nachfolgend in Tabelle 5.2 nur die Anforderungen betrachtet, die unterschiedlich bewertet werden. Diese gliedern sich in die folgenden Aspekte:

Anforderung	Bidirektionaler Austausch	Erweiterung des Lokalisierungsdienstes	IdP-Proxy
Funktionale Anforderungen			
[FA-Dynamik]	+	+	-
[FA-Entscheidungshilfe]	-	o	o
[FA-Fehlermanagement]	o	+	+
[FA-Föderation]	-	+	+
[FA-Grenzüberschreitend]	+	+	o
[FA-Reichweite]	+	+	o
[FA-Initiierung]	+	+	o
[FA-Integration]	o	o	+
[FA-Lokalisierung]	o	+	+
[FA-Monitoring]	-	-	o
[FA-SLA]	-	+	+
Nichtfunktionale technische Anforderungen			
[NFA-Performanz]	+	+	o
[NFA-Skalierbarkeit]	+	+	-
Sicherheitsanforderungen			
[SEC-Multilateral]	-	o	+
Organisatorische Anforderungen			
[ORG-Registrierung]	-	+	+
[ORG-Validierung]	o	+	+
Datenschutzanforderungen			
[DSA-CoCo]	-	o	+
[DSA-Datenschutz]	-	o	+

Tabelle 5.2.: Bewertung der Architekturmuster

[FA-Dynamik]: Durch bi-lateralen Metadatenaustausch und die Erweiterung des Lokalisierungsdienstes sind dynamische Austauschverfahren für Metadaten möglich. Dies vereinfacht Kooperationen und dynamische virtuelle Föderationen, was den ersten Aspekt der Anforderung betrifft. Diese Form der Dynamik ist bei IdP-Proxys nur dann möglich, wenn ein zentraler, globaler IdP-Proxy existiert. Dies wiederum ist auf Grund der Performanz und nachdem keine globale Föderation existiert (vgl. Kapitel 2) unwahrscheinlich.

[FA-Entscheidungshilfe]: Eine Entscheidungshilfe kann vor allem dann eingebaut werden, wenn es eine oder mehrere zentrale Komponenten gibt, die Informationen sammeln und diese auswerten. In einer ineffizienten und unsicheren Variante ist dies möglich, wenn es jeder SP für sich wahrheitsgemäß angibt. Folglich wird dieser Aspekt für den bidirektionalen Austausch mit Anforderung nicht erfüllt bewertet.

[FA-Fehlermanagement]: Für Fehlermanagement gilt dieselbe Begründung wie für die Anforderung [FA-Entscheidungshilfe].

[FA-Föderation]: Die Etablierung und Verwaltung von Föderationen ist durch die zentralen Komponenten von IdP-Proxy und dem erweiterten Lokalisierungsdienst möglich. Währenddessen existieren Föderationen im eigentlichen Sinn beim bidirektionalen Metadatenaustausch nicht mehr. Sie können durch zusätzliche Werkzeuge verwaltet werden, die außerhalb des Workflows liegen. Folglich wird diese Anforderung für den bidirek-

tionalen Austausch mit - bewertet.

[FA-Grenzüberschreitend]: Beim bidirektionalen Metadatenaustausch werden Grenzen bereits überschritten, wenn Identity Provider und Service Provider in unterschiedlichen Ländern lokalisiert sind. Die Erweiterung des Lokalisierungsdienstes sieht ebenfalls keine Beschränkung bezüglich der Örtlichkeit vor. Ein IdP-Proxy kann prinzipiell auch grenzüberschreitend eingesetzt werden, jedoch ist ein globaler IdP-Proxy nicht möglich. Ursprünglich wurden IdP-Proxys von Föderationen eingesetzt, um untereinander FIM zu betreiben. Damit die Teilnehmer auch außerhalb der Föderation FIM verwendet werden können, ist der Import und Export des Metadatenfeeds notwendig. Daher wird die Anforderung nicht vollständig erfüllt.

[FA-Reichweite]: Die Anforderung [FA-Grenzüberschreitend] zeigt die Auswirkungen in der Reichweite. Während die Reichweite vom bidirektionalen Metadatenaustausch prinzipiell unendlich ist, wird sie bei der Erweiterung des Lokalisierungsdienstes bereits durch die Registrierung beim Lokalisierungsdienst eingeschränkt. Bei der Verwendung von IdP-Proxys ist es essentiell, dass die IdP-Proxys alle Teilnehmer kennen. Durch die beschriebenen Nachteile des IdP-Proxys ist die Reichweite eingeschränkt.

[FA-Initiierung]: Die Initiierung des Metadatenaustausches ist beim bidirektionalen Austausch sowie der Erweiterung des Lokalisierungsdienstes möglich, siehe Anforderung [FA-Dynamik] der Metadaten. Währenddessen ist die Initiierung bei der Verwendung von IdP-Proxys nur eingeschränkt möglich.

[FA-Integration]: Diese Anforderung spiegelt die Erweiterung des Standard-Workflows von SAML wieder. Sowohl beim bidirektionalen Austausch als auch bei der Erweiterung des Lokalisierungsdienstes sind Anpassungen notwendig. IdP-Proxys lassen sich hingegen mit Standard-Workflows von SAML und den bekannten SAML-Implementierungen betreiben.

[FA-Lokalisierung]: Für die Lokalisierung beim bidirektionalen Austausch sind zu konzipierende Verfahren notwendig. Dagegen können ein erweiterter Lokalisierungsdienst sowie ein IdP-Proxy auf vorhandene Lösungen zurückgreifen.

[FA-Monitoring]: Je zentraler eine Lösung, desto leichter kann FIM überwacht werden. Während dies bei IdP-Proxys üblich ist, sieht ein dezentraler Metadatenaustausch keinerlei Überwachung und Monitoring vor. Bei der Erweiterung des Lokalisierungsdienstes ist Monitoring möglich, wenn es konzipiert und bedacht wird. Trotzdem wird es, um den Unterschied zu IdP-Proxys darzustellen, mit nicht erfüllt bewertet.

[FA-SLA]: Für diese Anforderung gilt erneut das Prinzip, dass eine zentrale Lösung, wie IdP-Proxy und TTP, die Etablierung von SLAs vereinfachen.

[NFA-Performanz]: Die vermutete Performanz beim bidirektionalen Metadatenaustausch ist am höchsten, nachdem keine zusätzlichen Komponenten involviert sind. Während-

dessen wird der IdP-Proxy auch für die Authentifizierung benötigt, was die Performanz vermindern kann. Die Erweiterung des Lokalisierungsdienstes liegt zwischen beiden Architekturmustern.

[NFA-Skalierbarkeit]: Die Begründung der Anforderung [NFA-Performanz] kann ebenfalls für die Skalierbarkeit eingesetzt werden. Der bidirektionale Metadatenaustausch skaliert am besten, während IdP-Proxys mehr Komponenten benötigen und daher am schlechtesten bezüglich der Skalierbarkeit des Systems abschneiden.

[SEC-Multilateral]: Die multilaterale Sicherheit lässt sich durch IdP-Proxys am einfachsten gewährleisten. Beim bidirektionalen Austausch ist eine multilaterale Sicherheit nur mit zusätzlichen Werkzeugen möglich und nicht steuerbar.

[ORG-Registrierung]: Nachdem es beim bidirektionalen Metadatenaustausch keine zentralen Komponenten gibt, können sich IdP und SP nicht registrieren. Das Gegenteil ist der Fall beim erweiterten Lokalisierungsdienst und beim IdP-Proxy.

[ORG-Validierung]: Die Validierung der Metadaten ist lokal möglich, jedoch hat ein Validierungstool einen größeren Nutzen, wenn es bei einer zentralen Lösung integriert ist. Dies hat zudem den Vorteil, dass sich die Entitäten sicher sein können, dass die potentiell ausgetauschten Metadaten den Konventionen entsprechen.

[DSA-CoCo]: Bei einer dezentralen Lösung kann der CoCo nur von den Teilnehmern kontrolliert werden. Bei zentralen Lösungen ist dies technisch möglich, während bei IdP-Proxys dies eventuell die Administratoren übernehmen. Jedoch kann nicht mehr als die Entity Category in den Metadaten sowie die Selbsterklärung überprüft werden.

[DSA-Datenschutz]: Die Begründung und Erläuterung von Anforderung [DSA-CoCo] gilt ebenso für den Datenschutz.

Dabei zeigt sich, dass eine Erweiterung des Lokalisierungsdienstes die meisten Anforderungen erfüllt. Es sind zwar stärkere Anpassungen an den Workflow notwendig, jedoch hat dies, in Hinblick auf das Basismodell und die Dienstsicht, den Vorteil, dass IdPs und SPs eine einheitliche Schnittstelle zu Föderationen und Inter-Föderationen erhalten. Zudem wird der Benutzer geführt und der Metadatenaustausch findet bedarfsgerecht statt.

Metadatenaustausch über eine Trusted Third Party

Der Austausch von Metadaten, falls Identity Provider und Service Provider sich noch nicht kennen, wurde im vorherigen Abschnitt bereits anhand des Standard-Workflows beschrieben. Der Workflow soll möglichst weitgehend mit den üblichen Workflows von SAML übereinstimmen und den Lokalisierungsdienst als Einstieg nutzen. Folglich werden HTTP Nachrichten verwendet, wie POST, GET und Redirect. Nach dem Austausch der Metadaten kann überprüft werden, ob weitere Konvertierungsregeln benötigt werden oder andere Mechanismen

greifen sollen. Wenn der SP auch die Metadaten des IdPs integriert und die benötigten Benutzerinformationen vom IdP erhalten hat, kann der Nutzer auf den Dienst zugreifen. Zusätzlich existieren folgende Varianten, die alle durch den standardisierten, modularen Ablauf abgedeckt werden können:

- Einbeziehung einer Attribute Authority, um alle Entitäten zu berücksichtigen.
- Konfigurationsmöglichkeiten, die es erlauben bestimmte Entitäten und bestimmte Vertrauensklassen auszuschließen.
- Verwendung von Entity Categories, die beim Szenario der Inter-Föderation eduGAIN eingesetzt wurden.
- Überprüfung des Datenschutzes, welcher wie beispielsweise der Code of Conduct ebenso als Entity Category modelliert werden kann.
- Einbeziehung einer Föderationsverwaltung, die weitere Konfigurationsmöglichkeiten hat.

Im Gegensatz zu OpenID Connect gibt es in SAML bisher keine Möglichkeit Metadaten dynamisch auszutauschen. Damit der Austausch von Metadaten für SAML dynamisch ablaufen kann, muss ein Protokoll spezifiziert werden. Keines der in Kapitel 3 genannten Protokolle unterstützt vollständig einen dynamischen Metadatenaustausch. IdP Discovery dient nur den passenden IdP zu wählen, während das Metadata Query Protocol einen Abruf von einzelnen Metadaten erlaubt, der jedoch manuell ausgeführt werden muss. Um einen möglichst modularen Aufbau zu ermöglichen, vorhandene Protokolle wiederzuverwenden und somit ein gleich bleibendes Nutzererlebnis zu ermöglichen, soll das IdP Discovery Protocol erweitert werden. Der Abruf von Metadaten kann hierbei über das Metadata Query Protocol geschehen, nachdem dies eine einfache und zugleich zielführende Art des Abrufes ist. Um die erfolgreiche Integration der Metadaten zu gewährleisten, soll MdFIM den Austausch orchestrieren. Zusätzlich zum eben beschriebenen Ablauf gelten folgende Bedingungen bzw. Besonderheiten:

Schritt 1: Als Vorbedingung müssen die Pfade zu den lokalen Implementierungen von MdFIM bekannt sein. Das Protokoll ist unterteilt in die Authentifizierung des Nutzers und den Metadatenaustausch. Nach der Auswahl des Dienstes durch den Nutzer, wird der HTTP Request des User Agents, normalerweise des Browsers des Benutzers, an die TTP gesendet.

Schritt 3: Anschließend erfolgt die Auswahl des IdPs, wodurch ein weiterer HTTP Request an den SP gesendet wird. Dieser Schritt entspricht weitgehend einem standardisierten SAML-Workflow und beeinflusst das Nutzererlebnis erheblich.

Schritt 4: Der wesentliche Unterschied zum Standard IdP Discovery liegt im zweiten Teil des Protokolls. Der Authentication Request wird über den User Agent vom SP nicht

an den IdP weiter geleitet, sondern an die TTP. Dieser Schritt ist notwendig, da IdPs auf Anfragen von ihnen unbekannten SPs nicht antworten.

Schritt 5: Dieser Authentication Request wird bei der TTP zwischen gespeichert.

Schritt 8: Damit die Metadaten ausgetauscht werden, muss dieser Schritt eingeleitet werden. Dies kann durch einen neuen HTTP Request geschehen, der eine definierte `action`, beispielsweise `action=fetchmetadata` in Zusammenhang mit der `entityID` als Parameter, verwendet. Nach der Abfrage der Metadaten muss die entsprechende Entität einen HTTP Response an die TTP mit dem Status der Integration senden. Da es möglich ist, dass der Server neu gestartet werden muss, soll dies in der Antwort ebenfalls ersichtlich sein. Nach dem Neustart wird hier ein weiterer Response notwendig, der den nun aktuellen Stand der Integration wiedergibt. Wenn IdP oder SP die Integration nicht erfolgreich abgeschlossen haben, soll die Integration zurück gerollt werden.

Der gesamte Workflow ist im Anhang A.1 dargestellt. Der modulare Aufbau erlaubt es ein anderes Protokoll, wie das Metadata Query Protocol, für den Abruf der Metadaten einzusetzen. Gleichzeitig wird hierdurch kein neuer Abruf spezifiziert, damit vorhandene Protokolle angewandt werden können. Das Nutzererlebnis ist ähnlich zum normalen Lokalisierungsdienst. Gleichzeitig kann das Verfahren zur Lokalisierung des Nutzers ausgetauscht oder der orchestrierte Metadatenaustausch erweitert werden.

Kommunikation zwischen Identity Provider und Trusted Third Party

In diesem Abschnitt wird die Kommunikation zwischen Identity Provider und Trusted Third Party im Detail beschrieben. Zunächst wird der zweite Authentication Request von der TTP zur IdP betrachtet. Nachdem die TTP den ursprünglichen Authentication Request des SPs zwischen gespeichert hat, muss die TTP einen neuen Request für den Identity Provider erstellen, der beispielsweise wie folgt aussieht (vgl. Listing 5.1).

```
1 GET /idp/profile/SAML2/Redirect/SSO?SAMLRequest=Lb8IwEIT%JvtdgeoJvdtc
2 Ytgkvndg2kMvleNswJjklgargroigfbrkjbs%2%2FdmC68No%2FeK22Qnz4S0jjDrUCF3
3 &target=tt:mem: HTTP/1.1
4 Host: idp.lrz.de
5 ...
```

Listing 5.1: Beispiel des zweiten Authentication Requests

Die URL des IdPs muss der TTP vorab bekannt sein. Dies wird dadurch ermöglicht, dass die URL in der Managementplattform MdFIM gespeichert ist. Alternativ können bekannte Pfade verwendet werden. Der darin enthaltene SAML Request initiiert die Authentifizierung, bevor die Metadaten zwischen Identity Provider und Service Provider ausgetauscht werden (vgl. Listing 5.2).

```
<samlp:AuthnRequest xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
        AssertionConsumerServiceURL="https://acs.ttp.edu/Shibboleth.sso/SAML2/
    POST"
        Destination="https://idp.lrz.de/idp/profile/SAML2/SSO"
        ID="_a6b3cad8ccea12as2f2751"
        IssueInstant="2014-12-01T15:44:16Z"
        Version="2.0">
        <saml:Issuer xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">
            https://acs.ttp.edu/shibboleth
        </saml:Issuer>
        <samlp:NameIDPolicy xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
            AllowCreate="1"/>
</samlp:AuthnRequest>
```

Listing 5.2: Beispiel des SAML Requests des zweiten Authentication Requests

Anschließend wird der Nutzer authentifiziert, um die Anzahl der ausgetauschten Metadaten zu reduzieren. So können nur Nutzer für ihren IdP Metadaten austauschen. Daraufhin antwortet der User Agent durch das HTTP POST-Binding, wie in Listing 5.3 zu sehen.

```
POST https://acs.ttp.edu/Shibboleth.sso/SAML2/POST HTTP/1.1
    POSTDATA
        RelayState=tt:mem:
        SAMLResponse=VbTG5CdJHHTfdJK67BveTbmNvZGl1
    Referer: https://idp.lrz.de/idp/profile/SAML2/Redirect/SSO
    Set-Cookie:
    _idp_session=Gtsdhkeudbt3jiMDEuc3J2LAvc2hpYmJvbGV0aA%3D%3D+;
    Version=1; Comment="Used to cache users successful authentication";
    Path=/
```

Listing 5.3: Beispiel des POSTS

Die darin enthaltene SAML Response enthält die Bestätigung der Authentifizierung. Zusätzlich wird ein Session Cookie gesetzt. Diese Nachricht wird an die TTP und nicht an den SP gesendet, da der zweite Authentication Request von der TTP an den IdP gestellt wurde. Anschließend wird der Metadatenaustausch, wie im nächsten Abschnitt beschrieben, initiiert. Nach dem erfolgreichen Metadatenaustausch wird der ursprüngliche Authentication Request an den IdP weiter geleitet. Dieser antwortet nun direkt an den SP. Da in der Regel ein aktiver Session Cookie existiert, muss sich der Nutzer nicht erneut authentifizieren.

Kommunikation zwischen Service Provider und Trusted Third Party

Nach der Auswahl des IdPs durch den Nutzer und der erfolgreichen Authentifizierung, müssen IdP und SP ihre Metadaten austauschen. Dies geschieht durch ein HTTP GET von der TTP zur SP, wie im folgenden Beispielcode (vgl. Listing 5.4), bei dem der SP SpringerLink die Metadaten des LRZs abrufen soll. Die URL muss vorab der TTP bekannt sein. Als `action` wurde `fetchmetadata` gewählt, während die `entityID` als Parameter angehängt wird.

```
1 GET /sp/profile/SAML2/DAME?action=fetchmetadata
2 &entityID=https://idp.lrz.de HTTP/1.1
3 Host: sp.springerlink.de
```

Listing 5.4: Beispiel eines HTTP GET zum Metadatenaustausch

Der Abruf der Metadaten kann durch das Metadata Query Protocol von Ian Young geschehen. Dieser verwendet ebenfalls HTTP GET. Das Protokoll legt zudem die URL des Abrufs fest (vgl. Listing 5.5).

```
1 GET /service/entities/https%3A%2F%2Fidp.lrz.de%2Fidp HTTP/1.1
2 Host: wayf.ttp.edu
3 Accept: application/samlmetadata+xml
```

Listing 5.5: Beispiel der Nutzung von Metadata Query Protocol zum Metadatenaustausch

Der Host ist der Discovery Service der TTP. Akzeptiert werden SAML Metadaten, die im XML-Format vorliegen. Als Antwort zum Metadata Request werden die entsprechenden Metadaten an den SP gesendet. Die Antwort auf den ersten HTTP Request zum Metadatenaustausch beinhaltet den Status der Integration, beispielsweise `HTTP/1.1 200 OK`. Durch die Abfrage des Status der Integration durch den IdP kann der SP erkennen, ob der Metadatenaustausch erfolgreich war oder nicht. Dies ist in Listing 5.6 dargestellt.

```
1 GET /service/DAME?action=add
2       &entityID=entityID
3       &status=success HTTP/1.1
4       Host: wayf.ttp.edu
```

Listing 5.6: Beispiel der Abfrage des Status der Integration der Metadaten

Methoden für den Metadatenaustausch

Um den dynamischen Metadatenaustausch zu realisieren, muss die TTP erweitert werden. Wichtige Handler für den Metadatenaustausch sind:

DiscoveryServiceHandler, der im Gegensatz zum bisherigen Discovery Service die bei MdFIM hinterlegten Metadaten der IdPs verwenden muss.

MetadataSyncHandler, der die Metadaten synchronisiert und die neueste Version lädt.

MetadataExchangeHandler, der sich um den Austausch der Metadaten kümmert. Dazu zählen die Requests, Trust-Abgleich, Orchestrierung und Fehlerbehandlung.

Diese Handler sowie der allgemeine Informationsfluss sind in der Abbildung 5.13 dargestellt. Der MetadataExchangeHandler wird im Folgenden anhand seiner Handler

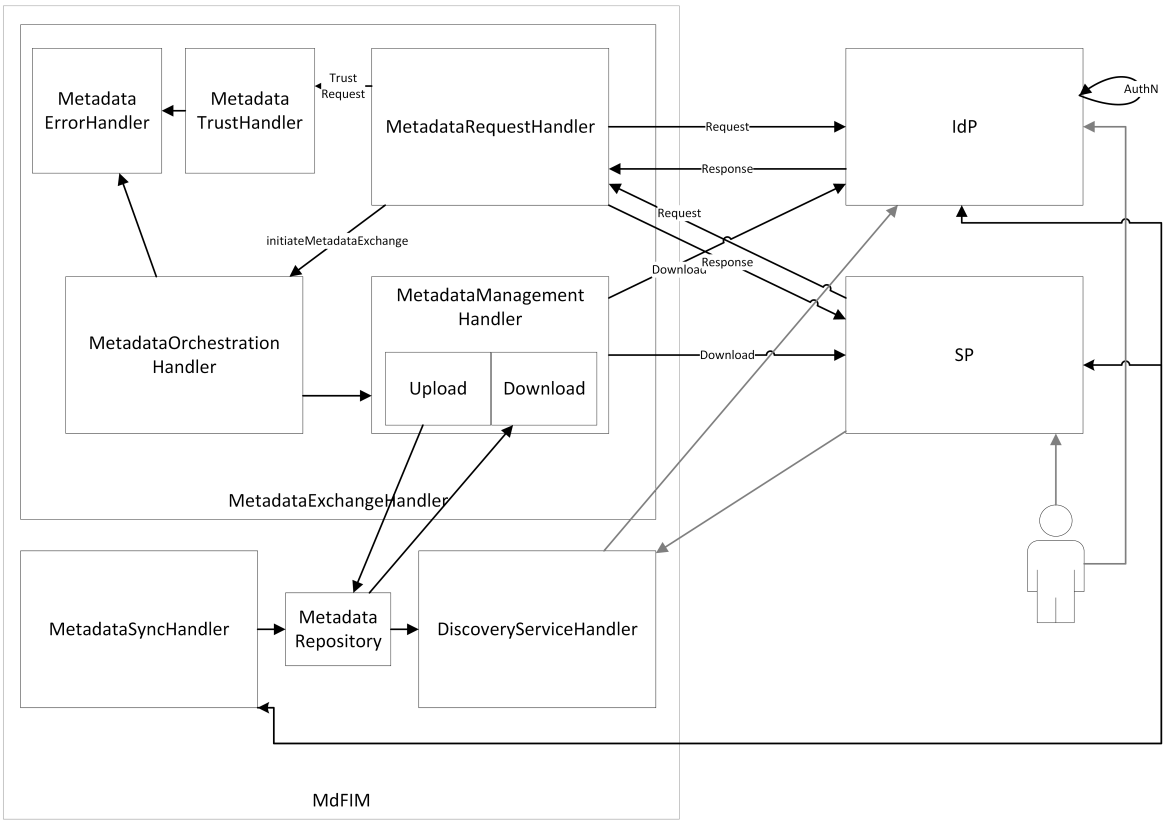

Abbildung 5.13.: Interner Aufbau des Metadata Managements

- MetadataRequestHandler,
- MetadataTrustHandler,
- MetadataOrchestrationHandler,
- MetadataErrorHandler und
- MetadataManagementHandler

näher erläutert.

Der `MetadataRequestHandler` ist für die Requests zuständig. Der Request des SPs enthält einen Authentication Request, der durch die TTP überprüft wird. Dazu müssen der Request bei Bedarf mit base64 dekodiert und die Parameter ausgelesen werden. Als Parameter sind die EntityIDs des SPs und IdPs wichtig, um den Request später an den Identity Provider leiten zu können. Diese beiden Parameter werden zusammen mit dem Relaystate und dem eigentlichen Request zwischengespeichert. Der neue Request an den IdP wird durch den `MetadataRequestHandler` erstellt und über das HTTP Redirect Binding abgeschickt. Nachdem sich der Nutzer erfolgreich bei seinem Identity Provider authentifiziert hat, wird der `MetadataOrchestrationHandler` für den Metadatenaustausch verwendet. Der zwischengespeicherte Request des SPs wird nach erfolgreichem Metadatenaustausch an den IdP weitergeleitet.

Der `MetadataTrustHandler` dient zum Abgleich des Vertrauens. Nach dem Auslesen des Requests kann anhand der bei MdFIM gespeicherten Informationen sowie der Metadaten überprüft werden, ob die Verbindung zwischen Service Provider und Identity Provider erlaubt ist. Hierfür müssen für beide Entitäten Voraussetzungen und eigene Vertrauenswerte ausgelesen und verglichen werden. Dies geschieht in der Funktion `compareTrust(idp, sp)`. Wenn die Verbindung in beide Richtungen erlaubt ist (`true`), wird der neue Request an den IdP geschickt und damit der Metadatenaustausch eingeleitet. Andernfalls liefert die Funktion `false` zurück und der Metadatenaustausch wird abgebrochen. Der Nutzer sowie die Administratoren müssen geeignet informiert werden.

Der `MetadataOrchestrationHandler` ist für den eigentlichen Metadatenaustausch zuständig. Nachdem sich der Nutzer erfolgreich authentifiziert hat, wird der Metadatenaustausch initiiert. Der Handler orchestriert hierbei den Austausch; der Austausch selbst wird durch den `MetadataManagementHandler` durchgeführt, der die Metadaten auch verwaltet. Identity Provider und Service Provider werden hierfür nacheinander angefragt, die Metadaten der anderen Entität herunterzuladen und zu integrieren. Nach dem Austausch der Metadaten wird der Integrationsstatus der Entitäten durch den Handler abgefragt. Wenn beide die Metadaten erfolgreich integriert haben, wird der zwischengespeicherte Request durch den `MetadataRequestHandler` an den IdP weitergeleitet. Ist dies nicht der Fall, muss der Metadatenaustausch zurückgerollt bzw. neu gestartet werden.

Der `MetadataErrorHandler` ist für die Behandlung der Fehler zuständig. Außer wenn ein Service nicht erreichbar ist, wird der Statuscode `200 OK` ausgeliefert. Der Statuscode von HTTP wird im Body der Nachricht verwendet, um den Fehler zu beschreiben. Hierfür sind vor allem die Statuscodes `4xx` relevant, wie die folgende Auswahl zeigt:

- `400 Bad Request`, wenn der Request fehlerhaft ist.
- `401 Unauthorized`, wenn keine gültige Authentifizierung vorhanden ist.
- `403 Forbidden`, wenn keine passenden Berechtigungen oder Authentifizierung vorliegt.
- `405 Method not Allowed`, wenn nur eine Methode (GET bzw. POST) erlaubt ist und diese falsch verwendet wurde.
- `406 Not acceptable`, wenn ein falsches Format (z. B. nicht XML in Metadaten) vorliegt.
- `408 Request timeout`, wenn in einer festgelegten Zeitspanne keine Antwort kam.
- `429 Too many requests`, wenn von einer Entität zu viele Anfragen gestellt wurden.
- `500 Internal Server Error`, wenn kein anderer Statuscode zutrifft. Zudem muss in diesem Fall, soweit möglich, ein weiterer Text die Ursache beschreiben.

Je nachdem welcher Fehler aufgetreten ist, fällt die Behandlung anschließend unterschiedlich aus. Wenn der Request des SPs nicht dekodiert und ausgelesen werden kann, kann der Code `400` verwendet werden. Die Aktion soll, nachdem der Request durch den SP erneut generiert wurde, wiederholt werden. Kann der Request nicht gespeichert werden, muss `500` mit dem Vermerk `Cache Error` versendet werden. Ist der Speicherplatz voll, muss der Administrator der MdFIM informiert werden. War die Speicherung durch einen anderen Fehler nicht möglich, soll die Aktion wiederholt werden. Bei einem nicht kompatiblen Vertrauen, ist `403` passend. Wird der Nutzer nicht erfolgreich authentifiziert, kann `401` verwendet werden. Bei einem Fehler bei der Integration der Metadaten wird der Statuscode `500` mit Vermerk `Metadata Exchange Error` eingesetzt. Der Fehler kann u. a. durch Schreibfehler, fehlerhafte Überprüfung der Prüfsumme, fehlende Berechtigungen oder fehlendem Speicher hervorgerufen werden. Dieser führt dazu, dass der Metadatenaustausch, solange Berechtigungen und Speicher nicht abgehen, erneut angestoßen wird. Der Status des Metadatenaustausches wird zugleich in der Datenbank gespeichert. Wenn der Metadatenaustausch erneut scheitert, wird der Metadatenaustausch zurückgerollt und abgebrochen. Neben dem Speichern des Status in der Datenbank und dem Logging des Status auf Seiten der Entitäten, muss dem Nutzer eine entsprechende Fehlermeldung durch die TTP angezeigt werden. Dies geschieht über eine allgemeine Fehlerseite, die die passende Fehlermeldung anhand des Status in der Datenbank sucht und anzeigt.

Die verschiedenen Workflows werden über eine Workflow Engine verwaltet und der aktuelle Status abgefragt. Eine Workflow Engine verwaltet und überprüft verschiedene Workflows. Die Basis einer Workflow Engine bilden ein Softwaremodul und eine Datenbank. Durch die Workflow Engine kann der nächste passende Workflow ausgesucht und ausgeführt werden. Geschieht ein Fehler, wird der Workflow zurückgerollt, wie im `MetadataErrorHandler` beschrieben. Die Workflow Engine ist somit die Kontrolleinheit der hier beschriebenen Handler.

5.2.3. Realisierung des Informationsmodells

Damit die benötigten Informationen zum Metadatenaustausch und zur Föderationsverwaltung für MdFIM zur Verfügung stehen, benötigt es eine Art Datenhaltung, die im Folgenden beschrieben wird. Im Informationsmodell in Abschnitt 4.4 wurden bereits Domänen definiert. Diese sollen die Grundlage für die Modellierung der Datenhaltung darstellen.
Zunächst wird auf aktuelle Lösungen aus Kapitel 3 sowie weitergehende Ansätze eingegangen, bevor das für die Lösung verwendete Schema basierend auf dem Informationsmodell aufgezeigt wird. Darauf aufbauend wird eine API entwickelt, die eine einheitliche Schnittstelle zu MdFIM bereitstellt und somit von verschiedenen Implementierungen und Protokollen genutzt werden kann.

Auswahl der Datenhaltung

Das im Kapitel 3 vorgestellte PEER erfüllt, wie bereits gezeigt, nur einen kleinen Teil der Funktionalität der Managementarchitektur. Nachdem sich ein Lokalisierungsdienst als Einstieg anbietet, da hierbei auch der jeweilige IdP ausgewählt wird, wird eine eigene Implementierung bevorzugt.

Um die Informationen in MdFIM zu speichern, gibt es verschiedene Möglichkeiten. Zunächst werden bereits vorhandene Ansätze betrachtet, bevor eine Auswahl getroffen wird. Die Resource Registry verwendet eine relationale Datenbank (MySQL), um die benötigten Informationen zu speichern. Dies hat den Vorteil, dass leicht Verknüpfungen zwischen den Tabellen hergestellt werden können. Nachteilig ist die Abbildung von Metadaten über ein relationales Datenbankschema. Da sich die Elemente in Metadaten unterscheiden können und beispielsweise OpenID Connect ein anderes Metadaten-Schema verwendet als SAML, ist eine Nachbildung eher schwierig. Die Metadaten-Verwaltung der DFN-AAI basiert auf einem Filesystem, auf dem die Metadaten der einzelnen Entitäten gespeichert werden. Dieses Vorgehen erleichtert die Koexistenz verschiedener Metadaten-Strukturen und kann zudem leicht für Policies und mögliche Konvertierungsregeln übertragen werden. Eine Föderationsverwaltung sowie Metadaten-Management ist jedoch bei diesem Ansatz schwierig, da zusätzliche Informationen benötigt werden. PEER, ebenfalls in Kapitel 3 beschrieben, unterstützt das Repository git, welches im Gegensatz zum Filesystem eine Versionskontrolle für den Anwender bietet. Für die Benutzerverwaltung wird zudem eine Datenbank mit einem einfachen Datenbankmodell angelegt.

Um alle Informationen, die für MdFIM benötigt werden (vgl. vorheriges Kapitel), zu speichern, wird eine Datenbank benötigt. Nachdem sich fast alle Informationen relational abbilden lassen, wird eine relationale Datenbank bevorzugt. Metadaten, mögliche Konvertierungsregeln und Policies sind hierbei ausgeschlossen. Die Nachbildung von Metadaten ist, wie bereits beschrieben, in relationalen Datenbanken schwierig. Alternativ besteht die Möglichkeit native XML-Datenbanken zu verwenden. Da jedoch auch relationale Daten verarbeitet, sind Dateisysteme und Repositories in Zusammenspiel mit einer relationalen Datenbank im Vorteil. Das bedeutet, dass möglichst alle Daten in einer relationalen Datenbank gespeichert werden. Dateien, die sich nur schwierig relational abbilden lassen, werden außerhalb in einem Dateisystem oder einem Repository gespeichert. In der Datenbank ist ein Link zu dem Speicherort hinterlegt. Diese Variante wird bereits bei der DFN-AAI und PEER eingesetzt. Repositories haben gegenüber Dateisystemen den Vorteil, dass die Dateien zudem versioniert werden.

Folglich soll die Kombination aus relationaler Datenbank und einem Repository eingesetzt werden. Nutzer von MdFIM sollen zudem die Möglichkeit haben sich zwischen dem zentralen Repository und einer lokalen Speicherung zu entscheiden. Die lokale Speicherung soll parallel nutzbar sein, damit der Speicherbedarf von MdFIM gering gehalten und mehrfache Datenhaltung vermieden wird. Jede Organisation soll über einen eigenen Zweig verfügen, bei dem nur sie schreibenden Zugriff hat. Der Zweig ist wiederum unterteilt in Ordner für jeden Typ, der im Repository gespeichert wird. Damit sind ein Ordner für Metadaten, ein Ordner für Policies und ein weiterer Ordner für Konvertierungsregeln gemeint. Dieser Aufbau ist ähnlich wie beispielsweise in GitLab. Die Datenbank bzw. ein passendes Skript benötigt lesenden Zugriff, um beispielsweise Informationen aus den Metadaten auszulesen und die Metadatendatei an eine andere Entität senden zu lassen. Für Datenbank und Repository werden zudem Prozeduren beschrieben, die nicht mehr benötigte Informationen und Dateien deaktivieren und nach einer gewissen Quarantänezeit löschen, damit kein unnötiger Speicher verbraucht wird.

Datenmodell

Basierend auf den im vorherigen Kapitel spezifizierten Domänen, ergeben sich für das Datenmodell folgende Tabellen, die in der Abbildung 5.14 dargestellt sind.

- `User`: Benutzerverwaltung von MdFIM.
- `FIM`: Falls der Administrator FIM wählt und kein lokales Benutzerkonto verwenden wird, werden hier alle wichtigen Informationen gespeichert.
- `Entity`: Basisinformationen zu den Entitäten, beispielsweise Art der Entität, EntityID und Verknüpfung zu den Metadaten.
- `AA`: Um die Workflows der Attribute Authorities besser zu unterstützen, werden benötigte Informationen in dieser Tabelle gespeichert.

- `Entity_User_Relationship`: Verknüpfung zwischen Entitäten und Benutzern mit der Zuordnung von Rollen und den damit einhergehenden Berechtigungen.
- `Organization`: Information zu der Organisation, z. B. Name und URL. Jede Organisation kann mehrere Provider betreiben.
- `Trust`: Vertrauensbeziehungen zwischen IdPs und SPs mit dem Status der Beziehung und der Information, wie mit geänderten Metadaten umzugehen ist.
- `Metadata`: Ort der Metadaten, Besitzer, Verknüpfung zu womöglich bereits vorhandenen Metadaten des Providers, Kommentar und Status der Metadaten.
- `Conversion Rule`: Informationen zu Konvertierungsregeln.
- `Federation`: Bevor Föderationen ihren Zusammenschluss verwalten können, müssen sie sich registrieren. Die Informationen hierfür, u. a. Name und Besitzer, werden in dieser Tabelle gespeichert.
- `Federation_User`: Beschreibung der Benutzer und deren Rollen in einer Föderation.
- `Entity_Federation_Relationship`: Informationen, welche Entitäten welcher Föderation angehören. Hierüber können auch Verknüpfungen zwischen Föderationen und Inter-Föderationen abgelegt werden.
- `Policy`: Um Policies zu verwalten, werden hier der Speicherort, entweder im Repository oder über eine URL, Besitzer und der Name der Policy gespeichert.
- `Federation_Policy_Relationship`: Verknüpfung zwischen Föderationen und Policies.
- `Metainformation`: Allgemeine Informationen für MdFIM, wie die Status und mögliche Rollen. Hier können zudem Informationen, wie die verwendete Verschlüsselung oder Anzahl der Tage, nachdem nicht mehr gültige Metadaten gelöscht werden.

Application Programming Interface

Um eine einheitliche Schnittstelle zu MdFIM zu gewährleisten, entstehen aus den Protokollinteraktionen festgelegte API Calls, wie in [PMH14g] beschrieben. Diese API kann von verschiedenen Teilnehmern und unterschiedlichen Implementierungen sowie Protokollen aufgerufen werden. Sie bietet somit eine einheitliche Schnittstelle für die Erweiterungen der IdP- und SP-Software, sowie AAs. Nachdem verschiedene Protokolle und Implementierungen vorhanden sind, ist eine einheitliche Schnittstelle wichtig, damit jeder Teilnehmer darüber kommunizieren und sie wiederverwenden kann. Sie sorgt folglich dafür, dass MdFIM generisch ist und universell eingesetzt werden kann.

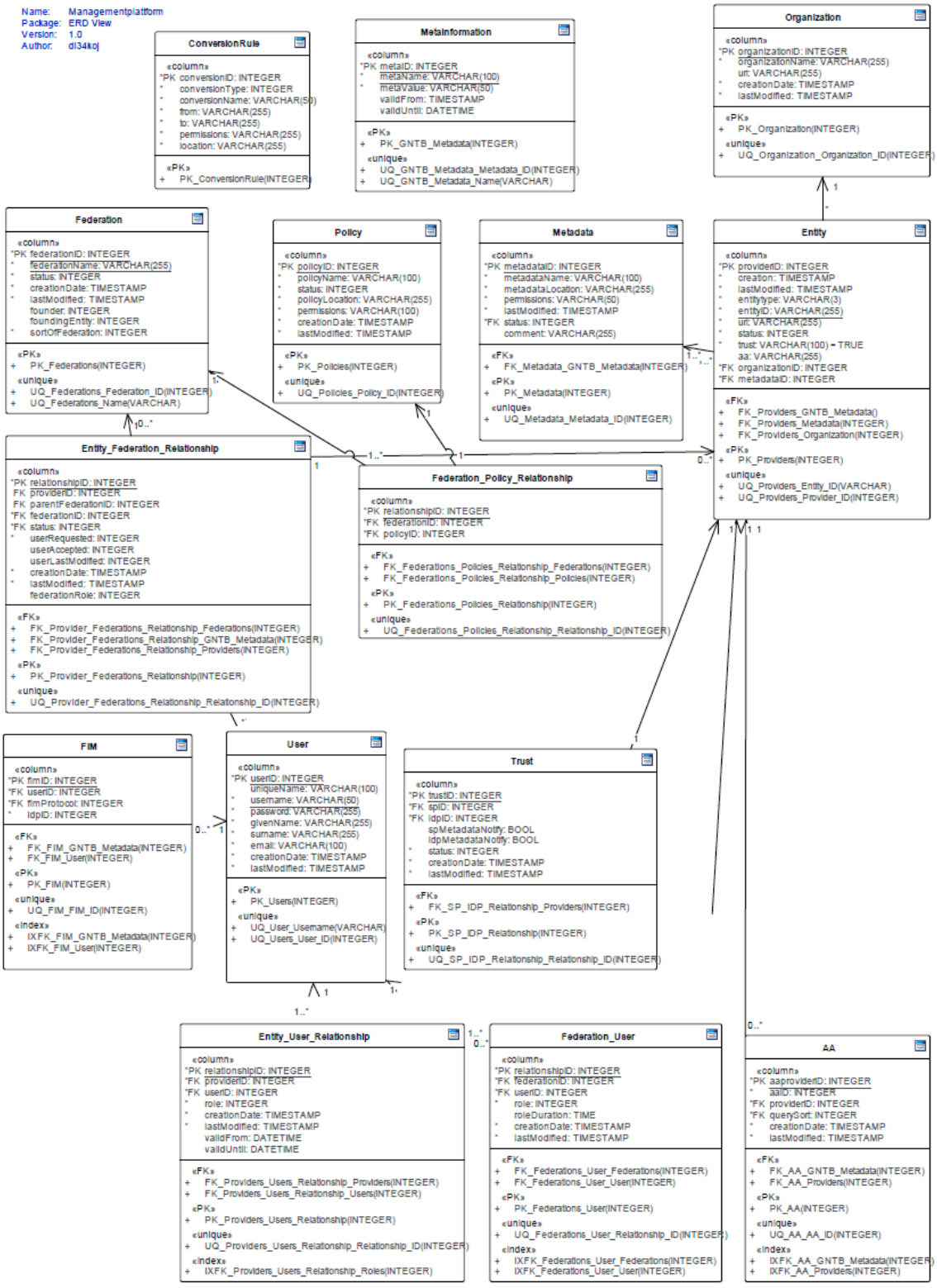

Abbildung 5.14.: Datenmodell der Trusted Third Party MdFIM

Basierend auf den bereits spezifizierten Protokollinteraktionen als Methoden der Domänen werden die API Calls festgelegt. Diese decken alle Methoden der Domänen ab und werden beispielhaft an den Klassen `Entity` und `Metadata` beschrieben. Die API Calls haben die Struktur:

`ttp Abk_Name(Parameter)`.

`ttp` steht für den Namen der TTP. `Abk` beschreibt die Kurzbezeichnung der Art des API Calls. Der `Name` spezifiziert die genaue Aktion, die durchgeführt werden soll. Die `Parameter` können beispielsweise EntityID, File oder Name sein.

Laut Informationsmodell besitzt die Klasse `Entity` die Methoden

- `getMetadata()`,
- `getTrust()` und
- `register()`.

Angewandt an das Schema ergeben sich die Methoden

- `ttp Entity_getMetadata(entityID)`,
- `ttp Entity_getTrust(entityID)` und
- `ttp Entity_register(entityID, name, url, entitytype, aa, timestamp)`.

Die Methode `ttp Entity_getMetadata(entityID)` ist dafür da, die Metadaten einer Entität herunterzuladen. Diese können durch die Erweiterung der Software automatisch in die lokale Konfiguration integriert werden. `ttp Entity_register(entityID, name, url, entitytype, aa, timestamp)` registriert eine Entität, während `ttp Entity_getTrust(entityID[IdP])` das Vertrauen einer Entität überprüft, was für den Vertrauensabgleich notwendig ist. Weitere Methoden sind dafür zuständig, Informationen abzufragen, Objekte zu erstellen, zu ändern und zu löschen sowie bestimmte Werte festzusetzen.

Für interne Zwecke können diese Methoden eingesetzt werden:

- `ttp Internal_handoverFed(federationID, userID[old], userID[new])`, um eine Föderation zu übergeben bzw. `ttp Internal_ handoverIFed(federationID, userID[old], userID[new])` um eine Inter-Föderation zu übergeben .
- `ttp Internal_acceptChange(changeID, boolean)`, um Änderungen zu akzeptieren oder abzulehnen.
- `ttp Internal_correlateMonitoringData(timeframe, federation/geolocation/...)` um Monitoring-Daten zu korrelieren.

- `ttp Internal_provideMonitoringData(monitoringDataID, email)`, um Monitoring-Daten für andere bereit zu stellen.

Die letzten beiden Methoden sind dafür da, Daten für das Monitoring zu korrelieren und bereitzustellen. Das Monitoring wird als besonders wichtige Funktion innerhalb des Service Managements beschrieben.

Bei Metadaten und Trust zeigt sich folgendes Bild:

- Der Nutzer stößt über `ttp Metadata_initiate(entityID[IdP], entityID[SP])` den Metadatenaustausch an.
- Die Methode `ttp Metadata_setAutomation(entityID, sort, rules)` konfiguriert die Automatisierung des Metadatenaustausches.
- Um das Vertrauen zu berechnen und zu vergleichen, wurde im Informationsmodell die Methode `ttp Trust_calculateTrust(entityID[IdP], entityID[SP])` angegeben.
- Metadaten können über die Methode `ttp Metadata_acceptMetadata(entityID, metadataID)` akzeptiert werden.
- Ebenso können Metadaten gesetzt (`ttp Metadata_setMetadata(entityID, metadataName, file)`) und
- aktualisiert (`ttp Metadata_updateMetadata(entityID, metadataName, file)`) werden.
- Entsprechend sollen Metadaten gelöscht (`ttp Metadata_deleteMetadata(metadataID)` und
- abgefragt (`ttp Metadata_getMetadata(metadataID/entityID)`) werden können.

Durch die API Calls, wie im Anhang A.2 aufgelistet, können ganze Workflows gebildet werden. Um die Aufrufe sowie die Fehlerbehandlung plastisch darzustellen, ist in Listing 5.7 ein Pseudocode für den Metadatenaustausch aufgeführt. Bevor die Metadaten ausgetauscht werden, wird das Vertrauen überprüft. Stimmt das überein, wird der orchestrierte Metadatenaustausch angestoßen.

```
typedef enum {NOT_FULFILLED, FULFILLED} comparison_result;

if metadata_initiate(entityID_idp, entityID_sp) {
  idp_trust := entity_getTrust(entityID_idp);
  sp_trust := entity_getTrust(entityID_sp);
  trust := metadata_calculateTrust(idp_trust, sp_trust);
  if trust=true{
     entity_getMetadata(entityID_idp);
          entity_getMetadata(entityID_sp);
          integration_idp := query_idp_metadata_integration;
          integration_sp := query_sp_metadata_integration;
          if integration_idP=true and integration_sp=true{
            return FULLFILLED;
          } else {
          message := "Integration failed";
          }
  }
  else if trust=false{
        message := "Not enough trust";
  }
  return NOT_FULFILLED(message);
}
```

Listing 5.7: Metadatenaustausch als Pseudocode

Die beschriebenen API Calls können jeweils für die unterschiedlichen Schnittstellen verwendet werden. Dadurch ist der einheitliche Aufruf der Funktionen gesichert, der auch für weitere Komponenten in Nachfolgearbeiten angewandt werden kann. Die API erlaubt es auch die Funktionen unterschiedlich zu nutzen, beispielsweise über ein Skript für IdPs und SPs oder über ein Web-Interface, was insbesondere für Validierung und Föderationen wichtig ist. Allgemein findet die Kommunikation über HTTP statt.

Abfragen, u. a. zur Verwaltung der Metadaten, geschehen äquivalent zum Metadatenaustausch über HTTP nach demselben Schema. Als Beispiel wird die Aktualisierung der Metadaten gezeigt. Der API Call für die Aktualisierung heißt
`ttp Metadata_updateMetadata(entityID, metadata)`.

Daraus entsteht der HTTP POST, der vom SP bzw. IdP zur TTP gesendet wird (vgl. Listing 5.8).

```
POST /ttp/metadata:updateMetadata
entityID=EntityID&metadata=File
```

Listing 5.8: Beispiel der Aktualisierung von Metadaten

Die TTP antwortet darauf mit einem HTTP Status. Solange der Service erreichbar ist, wird die Antwort `200 OK` gesendet. Der Status der Aktualisierung steht im Inhalt der Antwort (vgl. Listing 5.9). Dafür werden ein Status-Code sowie eine Beschreibung des Status verwendet. Dadurch kann auch eine Fehlermeldung der Entität mitgeteilt werden.

```
1 HTTP/1.1 200 OK
2 Content-Type: application/xml
3 Last-Modified: Wed, 01 Dec 2014 15:44:25
4 Content-Length: 1
5 <ttp:status status=status status_description="description"/>
```

Listing 5.9: Beispiel der Antwort auf die Aktualisierung von Metadaten

5.2.4. Realisierung des Organisationsmodells

Im Organisationsmodell (vgl. Abschnitt 4.3) wurden zunächst verschiedene Domänen beschrieben. Anschließend wurden, äquivalent zur Domäne *Role* des Informationsmodells, unterschiedliche Rollen in den Domänen spezifiziert. Diese Rollen können für das Rollenkonzept verwendet werden mit Ausnahme von User, der keinen direkten Zugriff auf MdFIM hat, sondern nur den Metadatenaustausch initiiert. Somit bleiben die folgenden Rollen zu berücksichtigen:

- Rolle *Administrator* (A): Diese Rolle kümmert sich um die reine Administration und hat all diejenigen Berichtigungen, die durch die unten genannten Rollen nicht abgedeckt sind.
- Rolle *Relationship Manager* (RM): Diese Rolle betreut die Mitgliedschaften in Föderationen bzw. Inter-Föderationen.
- Rolle *Configuration Manager* (ConM): Diese Rolle kann die Konfiguration erstellen, anpassen und löschen.
- Rolle *Change Manager* (CM): Die Rolle CM kann Änderungen, genauer gesagt Major Changes, autorisieren.
- Rolle *General Manager* (GM): Diese Rolle hat die Berechtigung Statistiken zu erzeugen und diese weiterzuleiten.
- Rolle *Initiator*: Diese Rolle kann eine Föderation bzw. Inter-Föderation erzeugen und den Aufnahmeprozess erstellen sowie die Föderation übergeben.

Zudem werden zwei weitere Rollen etabliert:

- Rolle *Master User*: Der *Master User* hat alle Berechtigungen, jedoch soll er nur im Notfall eingreifen können. Ein Beispiel hierfür ist, dass alle Rollen einer Föderation keinen Zugriff mehr auf MdFIM haben. Die Änderungen müssen durch die Rolle CM autorisiert werden. Zudem müssen die Aktionen geloggt werden, um sie nachvollziehbar zu machen.

- Rolle *Audit User*: Der *Audit User* soll alle Logfiles lesen können, damit MdFIM auch auditiert werden kann.

Ferner sollen die jeweiligen Rollen Stellvertreter-Rollen erstellen können, die nicht die vollen Berechtigungen der jeweiligen Rolle haben, aber bestimmte Aktionen selbstständig ausführen können. Dies kann beispielsweise der Fall bei einer Urlaubsvertretung sein, die diese Aufgaben noch nicht kennt.

Um das Rollenkonzept umzusetzen, benötigt es mehrere Komponenten. Diese Rollen und die damit zusammenhängenden Berechtigungen sollen in der Meta-Tabelle beschrieben werden. Während Datenbanken es erlauben den Zugriff auf bestimmte Tabellen zu beschränken, können durch native Mittel keine Zugriffe auf Seiten der Webanwendung eingeschränkt werden. Auch Webserver können nicht so feingranular über Berechtigungen entscheiden. Dies wird dadurch erreicht, dass die Berechtigung des Nutzers für einzelne Elemente abgefragt wird. Entsprechend der Rolle wird das Element freigeschalten oder nicht. Damit dies möglich ist, benötigt es die Information, welche Rolle welche Berechtigung hat. Die Logik, die es erlaubt Elemente freizuschalten basierend auf der Berechtigung und wird als ACL Service bezeichnet. Nach der Authentifizierung des Nutzers wird durch die Funktion `userACL` die Rolle des Nutzers abgefragt und in der Session des Nutzers gespeichert. Jeder API Call überprüft, ob der Nutzer die entsprechenden Berechtigungen hat. Bei der Webanwendung der Managementplattform MdFIM kann dies dadurch realisiert werden, dass jedes HTML-Element die Funktion `userACL` abfragt. Entsprechend der Element-ID und der Rolle wird das Element freigeschalten oder nicht. Dies kann neben aktivieren und deaktivieren auch lesenden Zugriff realisieren. Folglich reicht `true` und `false` nicht aus, sondern die einzelnen Stufen müssen als Resultat übermittelt werden. Die Funktion besteht aus einem Regelwerk, was Element-ID bzw. API Call mit verschiedenen Rollen und Berechtigungen verknüpft. Im Endeffekt ist die Kombination aus darunter liegenden Funktionen und der Abfrage der Berechtigung eine einfache Art von Workflow Engine, die auch in anderen Teilen der Managementplattform verwendet wird.

Beim bereits genannten API Call `ttp Metadata_updateMetadata(EntityID, File)` wird die Rolle des Nutzers abgefragt. Ist der Nutzer beispielsweise ein Relationship Manager, wird der Upload der neuen Metadaten nicht erlaubt. Hat der Nutzer die Rolle Administrator, ist eine Aktualisierung der Metadaten möglich. In der Webanwendung ist dies u. a. durch ein Aktivieren/Deaktivieren der Schaltflächen zum Aktualisieren der Metadaten möglich. Beim Funktionsaufruf der Aktualisierung soll die Berechtigung erneut abgefragt werden.

5.2.5. Realisierung des Funktionsmodells

Im Funktionsmodell, welches in Abschnitt 4.6 beschrieben wurde, wurden verschiedene Funktionen erläutert. Im Folgenden werden die folgenden Funktionen als Beispiele genauer beschrieben:

- Configuration Management,
- Member Management und
- Policy Management.

Configuration Management

Die Funktion Configuration Management behandelt die Konfiguration, also das *Erstellen*, *Ändern*, *Löschen* von Konfigurationen. Auf der Ebene der Föderation und Inter-Föderation können unterschiedlichen Ausprägungen der Föderation angegeben und der Aufnahmeprozess festgelegt werden. Dies beinhaltet die Policies und sonstigen Voraussetzungen, wie Audits und LoA, die Entitäten erfüllen müssen, falls dies laut den Ausprägungen benötigt wird. Bei Entitäten beinhaltet die Konfiguration die Automatisierung und das Vertrauen.

Im Folgenden sollen diese Fragen geklärt werden:

- Wie sieht Ausprägung einer Föderation bzw. Inter-Föderation aus und wie wird das modelliert?
- Was ist ein Aufnahmeprozess?
- Wie kann LoA/LoT oder allgemein Trust konfiguriert werden?
- Wie lässt sich die Automatisierung konfigurieren?

Die Ausprägung einer Föderation bzw. Inter-Föderation basiert auf der Klassifikation von Föderationen, wie sie im Kapitel 2 eingeführt wurden. Insbesondere die Aspekte Gruppenstruktur, organisatorische Dimension, Dauer der Föderation und Koordination können bei der Gründung einer Föderation angegeben werden, da sie essentiell sind. Dadurch werden ebenso die Aspekte Anzahl der Teilnehmer und Kooperationsstruktur festgelegt. Die Art der Zusammenarbeit soll zudem deklariert werden können, um gegebenenfalls Schnittstellen bereitstellen zu können. Die Aspekte des Vertrauens beschreiben das Vertrauen innerhalb der Föderation, während der Gründungsprozess durch dynamische virtuelle Föderationen sowie feste Föderationen gesteuert wird. Die räumliche Distanz wird durch die Teilnehmer bzw. durch die Föderationsverwaltung bestimmt und wird für die Ausprägung nicht weiter beachtet. Somit werden die einzelnen Aspekte der Klassifikation der Föderationen berücksichtigt, wie in Abbildung 5.15 dargestellt.

Der Aufnahmeprozess legt den Prozess fest, den eine Entität beschreiten muss, um bei einer Föderation aufgenommen werden. Folglich ist es wichtig festzulegen, ob

- Entitäten überprüft werden müssen oder nicht,

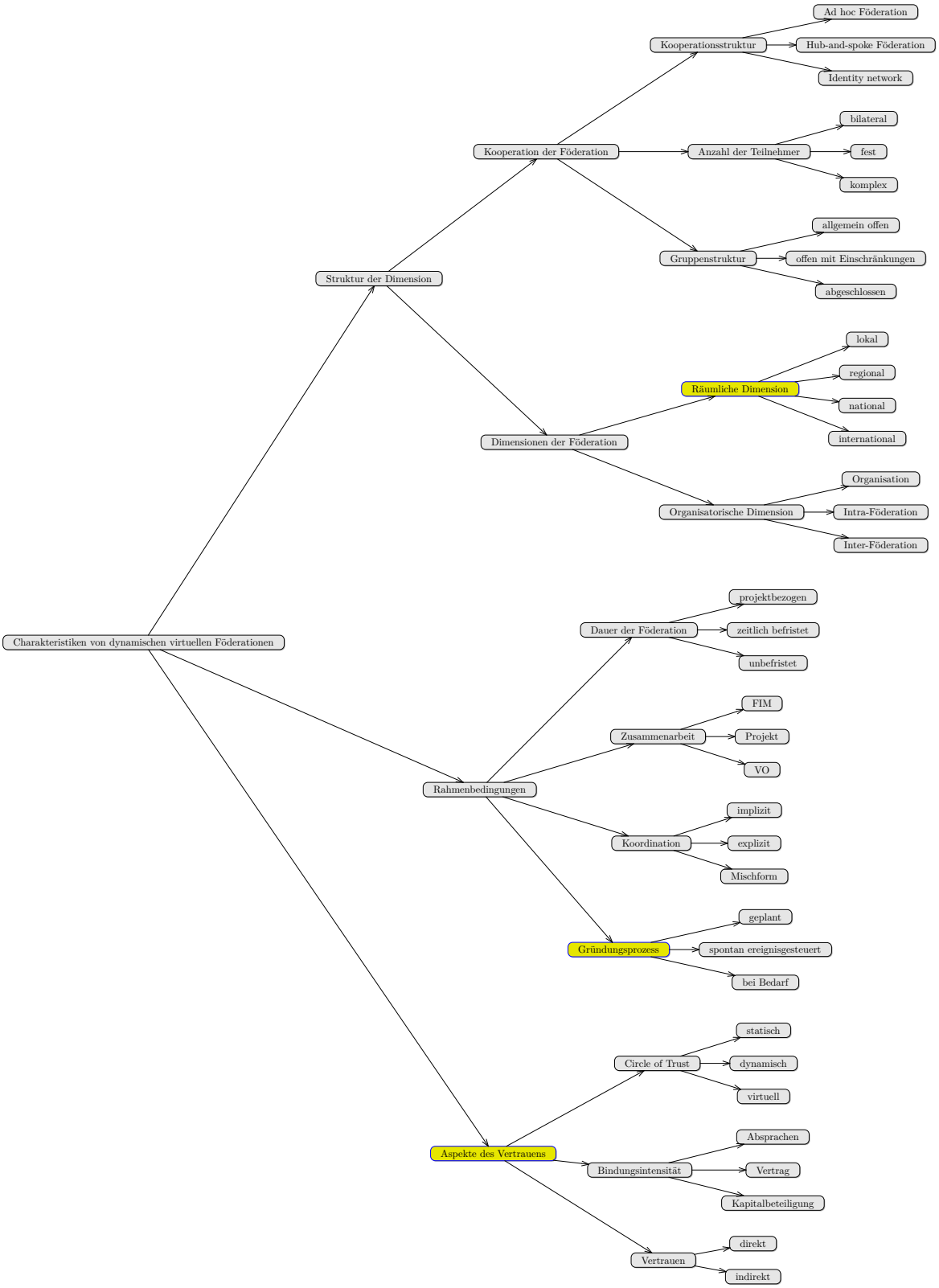

Abbildung 5.15.: Ausprägung von Föderationen anhand der Klassifikation

- Entitäten Policies akzeptieren müssen,
- die Einhaltung der Policies überprüft wird,
- Entitäten auditiert werden müssen und wie das Audit ausgeprägt ist (intern, extern, gegenseitig usw.) und
- Entitäten anschließend eine Art Teilnahmebestätigung, beispielsweise anhand eines Zertifikats, erhalten sollen.

Dies geschieht über eine Workflow Engine, die alle möglichen Varianten des Workflows abdecken kann. Die Anpassung des Workflows muss in der Datenbank mit der jeweiligen Föderation verknüpft sein, um den passenden Workflow auszuführen. Die Workflow Engine verwaltet Prozessabläufe, die in einer Datenbank gespeichert sind. Dazu wird der aktuelle Status des Workflows abgefragt, die Berechtigung des Nutzers überprüft und eine Funktion ausgeführt, die die einzelnen Schritte triggert. Werden alle Schritte korrekt ausgeführt, wird die Entität in die Föderation aufgenommen. Der Status der Entität wird in der Datenbank gespeichert. Anschließend wird der zuständige Nutzer über E-Mail von der Aufnahme bzw. Ablehnung informiert.

Die Workflow Engine wird ebenso verwendet, wenn Metadaten ausgetauscht werden sollen. Ein Teilschritt ist hierbei die Abfrage des Vertrauens. Das Vertrauen kann generisch konfiguriert werden durch die Methoden

- `ttp [Entity | Federation]_changeTrustRule()`,
- `ttp [Entity | Federation]_createTrustRule()` und
- `ttp [Entity | Federation]_deleteTrustRule()`.

Nachdem Trust im Abschnitt 5.4 näher erläutert wird, werden hier nur allgemeine Möglichkeiten der Konfiguration aufgezeigt:

- [Liste von Entitäten | Föderationen | Art von Entitäten | ab bestimmten Vertrauen | default] akzeptieren.
- [Liste von Entitäten | Föderationen | Art von Entitäten | unter bestimmten Vertrauen | default] ablehnen.
- [Liste von Entitäten | Föderationen | Art von Entitäten | zwischen bestimmten Vertrauenswerten | default] nachfragen.

Neben Akzeptieren und Ablehnen, soll es auf Grund der Konfiguration der Automatisierung auch eine Option Nachfragen geben, bei denen der zuständige Administrator beispielsweise über E-Mail aufgefordert wird, dem Metadatenaustausch zuzustimmen oder ab-

zulehnen, also eine explizite Entscheidung zu treffen. Liste von Entitäten und Föderationen kommt einer Whitelist bzw. Blacklist gleich. Die Art der Entitäten kann beispielsweise über Entity Categories angegeben werden. Als Beispiel werden alle Entitäten in der Forschung automatisch akzeptiert, während kommerzielle Entitäten abgelehnt werden. Forschung in der Wirtschaft wird nachgefragt, um finanzielle Konsequenzen zu überprüfen. Zudem soll es möglich sein auf Grund des Vertrauens Entitäten zu akzeptieren, abzulehnen oder nachzufragen. Praktisch kann die Abfrage des Vertrauens als eine geschachtelte if-then-else-Funktion `compareTrust(idp, sp)` im `MetadataTrustHandler` realisiert werden.

Ein weiteres Anwendungsgebiet stellen die Konvertierungsregeln dar, die im nächsten Abschnitt näher erläutert werden. Hier ist es beispielsweise möglich Konvertierungsregeln der eigenen Föderation automatisch integrieren zu lassen, während andere Regeln erst durch den Administrator freigegeben werden müssen. Dies ergibt folgende Konfigurationsmöglichkeiten:

- Konvertierungsregeln von [Liste von Entitäten | Föderationen | Art von Entitäten | ab bestimmten Vertrauen | default] akzeptieren.
- Konvertierungsregeln von [Liste von Entitäten | Föderationen | Art von Entitäten | unter bestimmten Vertrauen | default] ablehnen.
- Konvertierungsregeln von [Liste von Entitäten | Föderationen | Art von Entitäten | zwischen bestimmten Vertrauenswerten | default] nachfragen.

Die Konfiguration, die in der Datenbank gespeichert ist, wird durch den `ConversionTrustHandler` abgerufen und in der Funktion `convTrust(idp)` überprüft. Äquivalent zur Abfrage des Vertrauens bei den Metadaten wird dies ebenfalls durch eine geschachtelte if-then-else Funktion realisiert und durch die Workflow Engine verwaltet.

Member Management

In diesem Abschnitt soll das Member Management genauer beschrieben werden. Hierbei geht es um die Mitgliedschaft in sowohl festen als auch dynamischen virtuellen Föderationen und Inter-Föderationen. Um überhaupt eine feste Föderation zu erstellen, muss ein Workflow angestoßen werden. Dies ist mit Pseudocode in Listing 5.10 dargestellt.

Wenn die Etablierung erfolgreich war, können im nächsten Schritt Mitgliedschaften angenommen oder abgelehnt werden. Dies erfolgt möglichst automatisch durch einen Workflow und die Überprüfung der im nächsten Abschnitt beschriebenen Policy. Der beispielhafte Ablauf ist mit Pseudocode in Listing 5.11 beschrieben.

```
status := create_federation(federationName, timestamp);
if status = created{
        federation_id := get_federation_id(federationName);
        create_federation_configuration.workflow := workflow;
        create_federation_configuration.policy := policy;
    try {
           configuration := create_federation_configuration(federation_id);
        } catch { exception};

} else {
  if error = "404"{
    status := create_federation(federationName, timestamp);
  }
  if error = "409"{
  ...
}
```

Listing 5.10: Member Management - Erstellen einer Föderation in Pseudocode

```

if requestMembership(entityID) = true{
  request := notify_membership_request(entityID);
  foreach policy_item in policy{
     if auto_check_policy_item = true{
           auto_check[] := auto_check_policy_item;
         } else {
           manual_check[] := policy_item;
         }
   }
   foreach auto_check_item in auto_check[]{
       check_result := run_auto_check;
           if check_result = false{
              return to error_message("message");
           }
   }
   if manual_check[]!= null{
      foreach manuel_check_item in manual_check[]{
              display(manual_check);
              if manual_check = false{
                        return to error_message("message");
                  }
          }
   }
   acceptance := true;
   if error_message!=null{
      acceptance := false;
   }
   accept_membership(acceptance, message);
}
```

Listing 5.11: Member Management - Mitgliedsschaftsanfrage an eine Föderation in Pseudocode

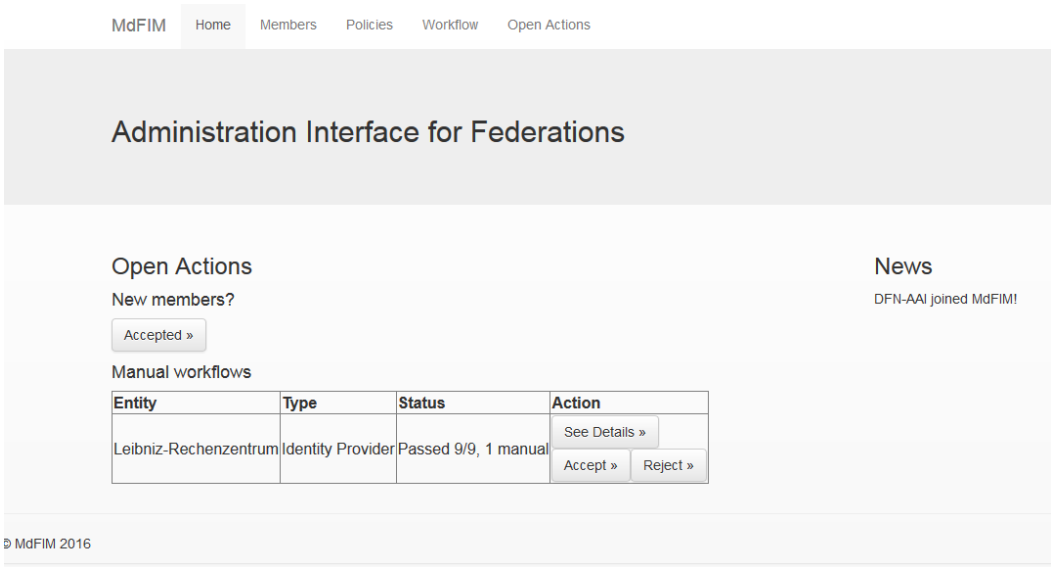

Abbildung 5.16.: Mockup für die Webanwendung für Föderationen zur Verwaltung der Mitglieder

Nachdem nicht jedes Policy-Item automatisch überprüft werden kann, weil beispielsweise manuelle Aktionen ausgeführt werden müssen oder nicht alle Informationen in den Metadaten stehen, wird in der Webanwendung für Föderationen eine Übersicht mit anstehenden manuellen Aktionen und Überprüfungen benötigt. Dies ist im Mockup der Abbildung 5.16 dargestellt.

Dynamische virtuelle Föderationen benötigen im Gegenzug keine Weboberfläche, um ihre Föderation zu verwalten. Basierend auf Vertrauensgraphen (Trust Graphs), die auf ausgetauschte Metadaten beruhen, können verstärkte Verbindungen festgestellt werden. Dies wird als Clusterkoeffizient bezeichnet. Während der lokale Clusterkoeffizient angibt, wie stark die Nachbarknoten miteinander vernetzt sind, wird der globale Clusterkoeffizient durch den Mittelwert der lokalen Clusterkoeffizienten gebildet. Diese Small-Worlds können durch unterschiedliche Graphen, wie Random Power Law Graphs, beschrieben werden. Eine weitere Möglichkeit der Optimierung ist Link Prediction, was zukünftige Vertrauensbeziehungen anhand des aktuellen Standes voraussagt. Die Algorithmen und Graphen sollen kontinuierlich beobachtet und verbessert werden. Unterschiede der Vertrauensbeziehungen zwischen festen Föderationen und diesen losen Verknüpfungen können zudem untersucht werden. Eine solche Untersuchung übersteigt den Rahmen dieser Arbeit, aber kann in zukünftigen Arbeiten genauer analysiert werden.

Policy Management

Ein essentieller Part des Aufnahmeprozesses ist, wie im vorherigen Abschnitt dargestellt, der Abgleich mit Policies. Die Policies sollen möglichst in XML bzw. XML Query Language (XQuery) in einem generischen Format gespeichert werden, um einen automatischen Abgleich über ein Skript gegenüber den Metadaten zu erlauben. Beim Vergleich der Policies der DFN-AAI, von SWITCHaai und der UK federation stellt sich heraus, dass die Anforderungen in drei Bereiche eingeteilt werden können: Allgemein, IdP-betreffend und SP-betreffend. Basierend aus diesen Policies wird durch eine gezielte Strukturierung, die erweiterbar ist, ein Schema für Policies entworfen. Dieses Schema kann auf XML angewandt werden und enthält die wichtigsten Aspekte der besagten Policies sowie weitere Informationen, die Kooperationen verbessern können.

Allgemeine Informationen und Anforderungen sollen Informationen über die Föderation, wie Name und Beschreibung, enthalten. Compliance betrifft neben der Aktualität und Richtigkeit von Daten das Ende der Mitgliedschaft, entweder durch die Föderation oder durch die Entität. Hier sollen bestimmte Fristen festgelegt werden, die beispielsweise eine Übergangszeit beschreiben. Technische Anforderungen beschreiben verwendetes Schema, Zertifikatsanforderungen, Bindings und Protokolle. Bei Bedarf kann dies erweitert werden. Organisatorische Anforderungen betreffen mögliche Dokumentationen, Audits (Art, Häufigkeit, um zwei Parameter zu nennen), Datenschutz und mögliche Einschränkung der Föderation, u. a. auf ein bestimmtes Projekt.

Audits wurden ursprünglich als Konsequenz des Sarbanes-Oxley Act of 2002 [tC02] in den Vereinigten Staaten von Amerika eingeführt. Gemäß dem Government Information Security Management Board VAHTI in Finnland [Boa06] sollen laut Mikael Linden [Lin09] Reporting Tools Reports über folgende Aspekte liefern:

- welche Rollen an einen Benutzer vergeben werden,
- welche Berechtigungen an eine Rolle vergeben sind,
- welche Berechtigungen an einen Benutzer vergeben sind,
- welche Benutzer eine bestimmte Rolle haben und
- welche Benutzer bestimmte Berechtigungen haben.

Durch die Tools soll bei Audits im Bereich von Identity Management herausgefunden werden, ob

- Benutzer im I&AM aktiv sind, die keine Mitarbeiter der Organisation mehr sind.
- es Rollen gibt, die nicht mehr benutzt werden.

- Objekte oder Berechtigungen vorhanden sind, die nicht mehr benötigt werden.
- es Autorisierungen gibt, für Objekte, die nicht mehr existieren.
- die Separation of Duties korrekt implementiert wurde.
- Rollen, Objekte und Prozesse Eigentümer mit definierten Verantwortungen haben.
- Identity Management Prozesse definiert und gelebt werden.

Die Klassifikation des Audits hängt hier wiederum von der Föderation bzw. Inter-Föderation ab. Beispielsweise verwendet SWAMID gegenseitige Audits, Haka interne Audits, während beim IAP Silver in InCommon externe Audits gefordert werden. Ferner soll es möglich diese Anforderungen für Mitglieder zu setzen, die erst überprüft werden, wenn die Mitgliedschaft offiziell ist. IdP-spezifische Anforderungen betreffen großteils die digitale Identität und Authentifizierungsart. Dies kann Teil eines LoA sein, der ebenfalls angegeben werden kann. Entsprechend sind die Anforderungen speziell für den SP LoT und Datenschutz, der auch Teil eines LoT sein kann. Daraus ergibt sich folgendes grundlegendes Schema (vgl. Abbildung 5.12):

```
Policy
        - General
                -- Information
                        --- Name
                        --- Description
                        --- Legal
                        --- Liability
                        --- Documentation
                -- Compliance
                        --- Registration
                        --- Data accuracy
                -- Technical Requirements
                        --- Schema
                                ---- XML-Name
                                ---- XML-URI
                                ---- Attribute
                                ---- Value
                        --- Certificate
                        --- Bindings
                                ---- DiscoveryResponse
                                ---- AssertionConsumerService
                                ---- AttributeService
                                ---- ...
                        --- Profiles
                -- Organizational Requirements
                        --- Documentation
                        --- Audits
                                ---- internal/external
                                ---- Form of Audits
                                ---- Regularity
                                ---- Product/Process/Combined
                                ---- Audit Documentation
                                ---- Metadata Element
                        --- Data Protection
                        --- Entity Category
                        --- Project/Country/...
                -- Members
                        --- Certificate
                        --- Entity Category
                        --- Data Protection

        - IdP
                -- General
                        --- Assurance
                                ---- Level
                                ---- Schema
                -- Digital Identity
                        --- Unique
                        --- Attributes
                        --- Accuracy
                        --- Deprovisioning
                -- Authentication
        - SP
                -- Trust
                        --- Level
                        --- Schema
                -- Data protection
```

Listing 5.12: Schema einer Policy

Die Überprüfung der Policy erfolgt über den `PolicyHandler`, der über eine geschachtelte if-then-else-Abfrage den Ist-Wert mit dem Soll-Wert abgleicht (`comparePolicy(entityID, federationID)`). Der Ist-Wert steht zum großen Teil entweder in den Metadaten oder in der Datenbank. Der Soll-Wert ist ebenfalls in der Datenbank gespeichert und kann über die federationID abgerufen werden. Wenn der Ist-Wert nicht in den Metadaten spezifiziert werden kann, wie z. B. Audit, dann muss eine manuelle Eingabe (`manualPolicyInput(entityID, federationID, item`) erfolgen. Dies ist durch die Föderation zu konfigurieren und der jeweilig zuständige zu informieren. Nach einer manuellen Auswertung und Eingabe der einzelnen Überprüfung, wird die Policy-Überprüfung fortgesetzt. Die bereits genannte Workflow Engine kann für die Überprüfung verwendet werden. Ist die Überprüfung positiv, wird die Entität in die Föderation aufgenommen (`addToFederation(result, entityID, federationID)`). Ist die Überprüfung negativ, wird diese abgelehnt. In beiden Fällen wird der zuständige Administrator über E-Mail (`informAdmin(result)`) informiert.

Nachdem die meisten Anforderungen Bestandteil der Metadaten oder der darin enthaltenen Verlässlichkeitsklasse sind, kann die Policy automatisch über ein Skript gegen die Metadaten überprüft werden. Dies ist bei automatischer Auswertbarkeit immer ein Vergleich von Ist-Wert, der aus den Metadaten ausgelesen wird, und Soll-Wert, der in der Datenbank bzw. im Policy-XML gespeichert ist. Nachdem XML XQuery als Abfragesprache verwendet, soll die Übereinstimmung mit den Soll-Werten durch XQuery aus dem Metadaten-XML der entsprechenden Entität abgefragt werden. Durch die einzelnen ausgefüllten Bereiche und deren Übersetzung in XQuery kann modular ein Skript zur Überprüfung aufgebaut werden. Als ein Beispiel für einen modularen Baustein soll eine Teilnahmebestätigung abgefragt werden. Die Policy kann flexibel aufgebaut sein, wie Figure 5.13 zeigt. Ein Policy-Element kann durch *type*, *value* und gegebenenfalls weiteren Informationen in *additional* spezifiziert sein. XQuery als Wert ist ebenso möglich. Wenn nur das Element in den Metadaten vorhanden sein muss, aber der Wert unwichtig ist, wird dies entsprechend im XML beschrieben.

```
<policy federation="" version="" date="">
   <general version="" date="">
      <information>
         <name type="mdui:DisplayName" value=""/>
         <description type="mdui:Description" value=""/>
      </information>
      <compliance>
         <registration>
            <registration_item id="1" value="https://www.aai.dfn.de"
                               type="mdrpi:RegistrationInfo" additional="
    @RegistrationAuthority"/>
            <registration_item id="2">
               exists(//mdrpi:RegistrationInfo[@registrationAuthority='https:
    //www.aai.dfn.de'])
            </registration_item>
         </registration>
         <data_accuracy>
            <data_accuracy_item id="" value="" type=""/>
         </data_accuracy>
      </compliance>
   </general>
   <identity_provider version="" date="">
    ...
   </identity_provider>
   <service_provider version="" date="">
    ...
   </service_provider>
</policy>
```

Listing 5.13: Ausschnitt aus der Policy in XML

Dieser Teil der Policy wird gegenüber dem Metadatensatz, genauer gesagt einem Ausschnitt davon, einer Entität geprüft (vgl. Listing 5.14).

```
<mdrpi:RegistrationInfo
 registrationAuthority="https://www.aai.dfn.de"
 registrationInstant="2009-05-27T12:36:25Z">
        <mdrpi:RegistrationPolicy xml:lang="en">
                https://www.aai.dfn.de/en/join/
        </mdrpi:RegistrationPolicy>
        <mdrpi:RegistrationPolicy xml:lang="de">
                https://www.aai.dfn.de/teilnahme/
        </mdrpi:RegistrationPolicy>
</mdrpi:RegistrationInfo>
```

Listing 5.14: Information über Registrierung in den Metadaten

Über XQuery kann dies beispielsweise wie folgt abgefragt werden (vgl. Listing 5.15). Wenn `true` das Ergebnis ist, wurde der Wert gesetzt und die Entität ist laut Metadaten Teilnehmer der Föderation DFN-AAI.

```
1 exists(//mdrpi:RegistrationInfo[@registrationAuthority='https://www.aai.dfn.de
    '])
```

Listing 5.15: Abfrage der Information zur Registrierung in den Metadaten

Die Managementplattform mitsamt der TTP sorgt u. a. für den dynamischen Metadatenaustausch, der Verwaltung der Mitglieder und der Kommunikation zwischen den beteiligten Entitäten. Auf Basis der Managementplattform MdFIM werden die beiden weiteren Werkzeuge Conversion Rule Management und Trust Management definiert, um ein dynamisches, automatisches FIM zu erhalten.

5.3. Conversion Rule Management

Die Konvertierung von Attributen ist immer dann notwendig, wenn der Service Provider ein anderes Schema verwendet als der Identity Provider. Dies ist beispielsweise dann der Fall, wenn der SP in einer anderen Föderation teilnimmt, die ihr eigenes Schema einsetzt und kein globales Schema vorhanden ist. Folglich ist die Konvertierung von Attributen eine relevante Art der Konfiguration für SPs. Aktuell erstellt jeder IdP-Administrator manuell die Regeln, was zum einem einen erhöhten Aufwand bedeutet und zum anderen Zeit bis zur Nutzung eines gewünschten Dienst für den Nutzer kostet. Im Kapitel 3 wurden bereits drei unterschiedliche Ansätze vorgestellt. Diese Ansätze und Alternativen werden analysiert, um die essentielle Komponente für die Automatisierung zu konzipieren.

Ein FSCS ähnliches Werkzeug zur Konvertierung von Benutzerinformationen wurde in der von der Autorin mitbetreuten Masterarbeit von Michael Grabatin [Gra14] veröffentlicht. Im GÉANT3plus OpenCall Projekt wurde durch Stefan Metzger und die Autorin die Attributskonvertierung speziell für Shibboleth konzipiert. Das Konzept wurde durch [PMH14b], [PMH14a], [PMH14g] und [PMH14c] veröffentlicht. Die Autorin war dabei maßgeblich bei der Entwicklung beteiligt. Die vorliegende Arbeit geht über die eben genannten Arbeiten hinaus und erstellt ein allgemeines Schema zur Konvertierung, wodurch das Werkzeug nicht nur durch Shibboleth, sondern auch durch andere Implementierungen und Protokolle genutzt werden kann. Das Konzept ist in [PH16] erläutert.

5.3.1. Selektion des Werkzeugs

Bevor die vorhandenen Ansätze betrachtet werden, sollen die Anforderungen an das Werkzeug nochmals hervorgehoben werden. Die Anforderung [FA-Schema] lässt sich, wie in Kapitel 2 zu sehen, in die folgenden Teilanforderungen untergliedern, die in diesem Abschnitt näher betrachtet werden:

- [Konv-Wiederverwendbarkeit],

- [Konv-Automatisierung],
- [Konv-Abdeckung],
- [Konv-Implementierungsunabhängigkeit],
- [Konv-Konvertierungen],
- [Konv-Modularität]und
- [Konv-Qualität].

Anhand dieser Anforderungen werden vorhandene Ansätze bewertet und es wird eine Auswahl getroffen.

Vorhandene Ansätze

Der aktuelle Ansatz, dass jeder IdP Administrator seine eigenen Regeln schreibt, ist nicht effizient. Somit kann [Konv-Wiederverwendbarkeit] nicht eingehalten werden. Zudem ist die Automatisierung nicht möglich. Nachdem jeder Administrator seine eigenen Regeln erstellen kann, können alle möglichen Schemata und Konvertierungen unterstützt werden. Die manuelle Arbeit ist implementierungsunabhängig und kann modular aufgebaut sein. Die Qualität der Regeln hängt vom IdP Administrator ab. In manchen Föderationen werden die wichtigsten Konvertierungsregeln zur Verfügung gestellt und sind daher auch von hoher Qualität.

Im Kapitel 3 wurde bereits der Konvertierungsservice eduGAIN Credential Conversion Service vorgestellt. eCCS bietet einen Online-Service für SPs an, um die Benutzerinformationen zu konvertieren. Der eCCS-Service ist beim MDS angesiedelt und kann durch den jeweiligen SP angefragt werden, nachdem die Föderationen Regeln festgelegt haben, wie ihr internes Schema in das eCCS-Schema umgewandelt werden kann. Nachdem die Föderationen einmal Regeln festlegen, die durch die Service Provider verwendet werden kann, ist die Wiederverwendbarkeit erfüllt. Ebenso sind alle Konvertierungen möglich und die Regeln sind von hoher Qualität. Die Modularität hängt von den Regeln ab, die die Föderationen bestimmt haben. Die Automatisierung hängt von der SP-Implementierung ab und wurde nicht umgesetzt. Die Abdeckung des Konzepts beschränkt sich auf die Inter-Föderation eduGAIN. Nachdem eCCS nicht umgesetzt wurde und keine interne Informationen verfügbar sind, kann [Konv-Implementierungsunabhängigkeit] nicht bestimmt werden.

Der Ansatz FSCS beruht auf Shibboleth und ist somit nur bedingt implementierungsunabhängig. Die Wiederverwendbarkeit ist über das Repository gegeben, da ein IdP seine Regel zu den Konvertierungen hochladen und andere Identity Provider diese Regel verwenden können. Je nachdem wie die Regeln aufgebaut sind, können sie auch modular eingesetzt werden. Die Automatisierung wird nicht ausgeführt. Ebenso hängt die Qualität stark davon

ab, wer die Regeln verfasst.

Die Ansätze im Bereich der Ontologie verschieben das Problem auf eine andere Ebene, denn auch die Ontologien müssen beschrieben werden. Nachdem einmal die Ontologien und die Mappings dargestellt wurden, können diese Informationen wiederverwendet werden. Die Abdeckung ist theoretisch sehr gut, ebenso sind alle Konvertierungen möglich. Modularität ist prinzipiell möglich. Die Qualität ist abhängig von den Mappings, die erstellt wurden. Im Gegensatz zu den vorherigen Ansätzen basieren Ontologien nicht auf SAML. Theoretisch können somit alle Implementierungen abgedeckt werden, jedoch ist die Realisierbarkeit geringer. Ebenso ist die Automatisierung nicht geklärt.

Insgesamt zeigt sich, wie in Tabelle 5.3 visuell dargestellt, dass kein Ansatz alle Anforderungen erfüllt. Die Automatisierung ist der größte Schwachpunkt. Zugleich ist die Qualität der Konvertierungen mit Ausnahme von eCCS nicht gesichert.

Anforderung	aktuelle Praxis	eCCS	FSCS	Ontologien
[Konv-Wiederverwendbarkeit]	-	+	+	+
[Konv-Automatisierung]	-	-	-	-
[Konv-Abdeckung]	+	-	+	+
[Konv-Implementierungsunabhängigkeit]	+	-	o	o
[Konv-Konvertierungen]	+	+	+	+
[Konv-Modularität]	o	-	+	o
[Konv-Qualität]	-	+	-	-

Tabelle 5.3.: Bewertung von Ansätzen zur Konvertierung

Auswahl

Nachdem eine Konvertierung ohne Regelaustausch nicht machbar ist, wird ein Ansatz ausgewählt, der geeignet erweiterbar ist. Sowohl FSCS als auch die Ontologien erfüllen die meisten Anforderungen. Nachdem FSCS am einfachsten umzusetzen ist und es bereits eine Referenzimplementierung mit GNTB existiert, wird dieser Ansatz an die Anforderungen angepasst. Dazu zählen die folgenden Aspekte:

- Die Qualität der Konvertierungsregeln soll kontrollierbar sein. Dazu sollen vorhandene Regeln der Föderationsverwaltungen in das Konvertierungsregel-Repository importiert werden. Diese Regeln erhalten eine hohe Bewertung, da sie durch die Föderationsverwaltungen getestet wurden. Folglich ist eine Bewertung der Regeln anhand des Erstellers notwendig. Damit können Regeln, die später von IdPs hinzugefügt werden, eine niedrigere Bewertung erhalten. Werden diese Regeln durch die entsprechende Föderationsverwaltung akzeptiert, steigt die Bewertung auf das Niveau von Föderationsverwaltungen. Dadurch kann die Qualität besser sichergestellt werden, vgl. [Konv-Qualität].

- Um die Automatisierung [Konv-Automatisierung] zu optimieren, soll auf Seiten des

IdPs eine Erweiterung der IdP-Software stattfinden. Diese kommuniziert mit MdFIM, um Regeln auszutauschen, herunterzuladen und automatisch in die lokale Konfiguration zu integrieren. MdFIM muss dementsprechend um die entsprechenden Funktionen erweitert werden, damit dies möglich ist. Dies gilt auch für die Erweiterung der IdP-Software.

- Die Konvertierungsregeln sollen in einem einfach zu konvertierenden Format abgespeichert werden, um die Implementierungsunabhängigkeit [Konv-Implementierungsunabhängigkeit] zu gewährleisten.

5.3.2. Spezifikation

Damit das Werkzeug Conversion Rule Management die eben genannten Aspekte enthält, wird es im Folgenden näher spezifiziert.

Konvertierung von Attributen

Um die wichtigste Funktionalität von Konvertierungsregeln zu gewährleisten, sollen die essentiellen Konvertierungen möglich sein. Wie bereits beschrieben, sind die häufigsten Konvertierungen Umbenennen, Transformieren und Splitten/Mergen von Benutzerinformationen. Zusätzlich werden im R&E-Umfeld häufig `scoped`-Attribute eingesetzt. Nachdem jede Implementierung von SAML jedoch andere, teils zusätzliche Konvertierungen ermöglicht, sollen diese zunächst verglichen werden.

Zur Umwandlung stehen in Shibboleth, wie in Kapitel 3 aufgezeigt, mehrere vorgefertigte Definitionen zur Verfügung, von denen die Folgenden für diese Arbeit relevant sind:

- Umbenennen durch Mappen von Attributen.
- regex für spezielle Definitionen mit regulären Ausdrücken, u. a. um ein Attribut zu splitten.
- Mergen durch die Template Attribute Definition mit der Velocity Template Sprache.
- Scoped Attribute Definition wird verwendet, um Attribute mit einfachem Namen und einen bestimmten Geltungsbereich zu definieren.
- Principal Name Attribute Definition bildet den Principal Name.
- Basierend auf der verwendeten Methode zur Authentifizierung wird ein Principal Authentication Method Attribute Definition kreiert.
- Falls keine vorgefertigte Transformation passt, kann das gewünschte Attribut mit

ECMA Skripten gebildet werden.

Das zeigt, dass für Mapping, Splitten und Mergen spezielle Definitionen vorhanden sind, auch wenn für Splitten eine regex eingesetzt wird. Transformation hingegen ist nur durch das Schreiben von Skripten möglich. Für scoped-Attribute gibt es ebenfalls eine Definition. Bei der Implementierung SimpleSAMLphp werden insbesondere die folgenden Konvertierungen eingesetzt:

- `core:AttributeAdd`: Fügt Attribute der Antwort (`Response`) hinzu.
- `core:AttributeAlter`: Sucht und ersetzt Werte von Attributen.
- `core:AttributeMap`: Benennt Attribute um.
- `core:AttributeRealm`: Erstellt Attribute durch die Hilfe des Benutzers.
- `core:PHP`: Modifiziert Attribute durch PHP Code.
- `core:ScopeAttribute`: Fügt Gültigkeitsbereich zu Attributen hinzu.
- `core:ScopeFromAttribute`: Erstellt ein neues Attribut basierend auf den Gültigkeitsbereich eines anderen Attributes.

Umbenenne und Scopen ist hier ebenfalls leicht möglich, während Transformation, Splitten und Mergen durch PHP, `AttributeAlter` oder eine Kombination ermöglicht wird.

PySAML2 verwendet ein Python Dictionary, um Attribute zu Mappen. `identifier` beschreibt das Namensformat, welches unterstützt werden soll. `to` und `fro` enthält anschließend das Mapping zwischen den Namen. Dies ist ein Beispiel für einfaches Umbenennen. Die Umbenennung erfolgt durch `attribute_converter.py`. Für Scopen muss eine regex eingesetzt werden. Zusätzlich können Funktionen aus Python eingesetzt werden. Somit ist nur einfaches Umbenennen durch vorhandene Konvertierungen möglich.

Claim Rules bei ADFS beschreiben, welche Claims an die Relying Party über ein Security Token gesendet werden. Zunächst müssen die Claims aus dem AD geholt werden. Nachdem Claims durch ADFS nicht das in R&E-Föderationen üblichen Format gesendet werden, müssen sie erst auf das Zielformat über die ADFS 2.0 Custom Rule Language transformiert werden. Dies geschieht erneut über das Administrationstool, in welches nach der Auswahl der zu erstellenden Regeln die Transformation geschrieben wird. Somit sind für ADFS erneut unterschiedliche Regeln und Sprachen notwendig.

Eine Zusammenstellung der Umsetzung der wichtigsten Konvertierungen findet sich in Tabelle 5.4. Die Tabelle zeigt auf, dass keine allgemein gültige Regel in MdFIM gespeichert werden kann. Vielmehr sollen alle wichtigen Informationen gespeichert werden, damit eine Umwandlung in die entsprechende Sprache möglich ist. Falls durch den Einsatz von Skripten

Implementierung	Umbenennen	Transformieren	Splitten	Mergen	Scopen
Shibboleth	+	Skripte	regex	+	+
SimpleSAMLphp	+	Skripte	Pattern	+	+
PySAML2	+	Skripte	Replace	Plus	regex
ADFS	Skripte	Skripte	Skripte	Skripte	Skripte

Tabelle 5.4.: Vorhandene Definitionen von Konvertierungsregeln

dies nicht möglich ist, sollen dem Administrator wichtige Informationen mitgeteilt werden. Dies sieht bei den Implementierungen wie folgt aus:

- Für ADFS ist dies immer der Fall.
- Bei PySAML2 wird für alle Konvertierungen außer Umbenennungen bei der ersten Verwendung nachgefragt. Bis auf Transformieren sollten alle Konvertierungsregeln durch Python-Funktionen einfach möglich sein.
- Bei SimpleSAMLphp ist dies bei Transformieren und Splitten der Fall.
- Shibboleth-Administratoren werden bei Transformieren und Splitten gefragt.

Weitere Konvertierungen müssen ebenfalls zunächst manuell eingepflegt werden. Wenn ein generisches Format entwickelt wird für diese Konvertierungen, kann dies wieder zu einer Automatisierung führen. Konvertierungsregeln, die Skripte enthalten, sollen aus Gründen der Sicherheit immer manuell heruntergeladen werden, während einfache Regeln, wie Umbenennen, automatisch heruntergeladen, transformiert und integriert wird. Da das Transformieren in das IdP-Format lokal geschieht, ist die Architektur möglichst leichtgewichtig und MdFIM speichert nur die wichtigsten Informationen.

Umsetzung der Konvertierung bei MdFIM

Die Regeln sollen wiederverwendbar sein. Das bedeutet, dass nur eine Person eine Regel erstellen muss, während andere diese wiederverwenden können. Hierfür muss eine Regel gespeichert werden. Da je nach Implementierung eine unterschiedliche Umsetzung verwendet wird, muss die IdP-Erweiterung ein Template für die Umsetzung der einfachen Konvertierungsregeln besitzen, während die generische Regel bei MdFIM gespeichert ist. Enthält eine Regel Skripte, muss diese Regel bei MdFIM im spezifischen Format gespeichert sein, um anschließend manuell herunter geladen zu werden. Um die Konvertierungen möglichst automatisch durchführen zu können und gleichzeitig auch komplexere Konvertierungsregeln speichern zu können, werden die folgenden Informationen benötigt:

- Art der Konvertierung,

- ursprüngliche Attribute,
- Zielattribute und
- zusätzliche Informationen, wie die regex.

Daraus ergibt sich folgendes generisches Format (vgl. Abbildung 5.16):

```
source={source1, source2,...}
transformation = [renaming, merging, regex, conversion, scoping]
target={target, targeturn1, targeturn2}

source(transformation) ⇒ target
```

Listing 5.16: Generisches Format für Konvertierungsregeln

Im Falle einer einfachen Umbenennung wird daraus Folgendes (vgl. Abbildung 5.17):

```
source
transformation = renaming
target={target, targeturn1, targeturn2}
```

Listing 5.17: Generisches Format für die Umbenennung als Konvertierungsregel

Die folgenden Begriffe sind Schlüsselwörter, die im später vorgestellten generischen Format der Implementierungen verwendet werden:

- `source`,
- `target`,
- `targeturn1`,
- `targeturn2` sowie die Transformationen
- `regex` bzw. `pattern` und
- `conversion`.

Diese werden für die Umsetzung in eine konkrete Konvertierungsregel durch die Werte in der Datenbank bzw. die Eingabe des zuständigen Administrators ersetzt. Zudem ist die Art der Konvertierung (*transformation*) wichtig, um zu wissen, welches generisches Format der Implementierung benötigt wird. Die Datenbank besteht aus den folgenden Tabellen. Diese sind auch in der Abbildung 5.17 dargestellt.

- `ConversionRule`: Konvertierung von einem oder mehreren Attributen in ein anderes Attribut.

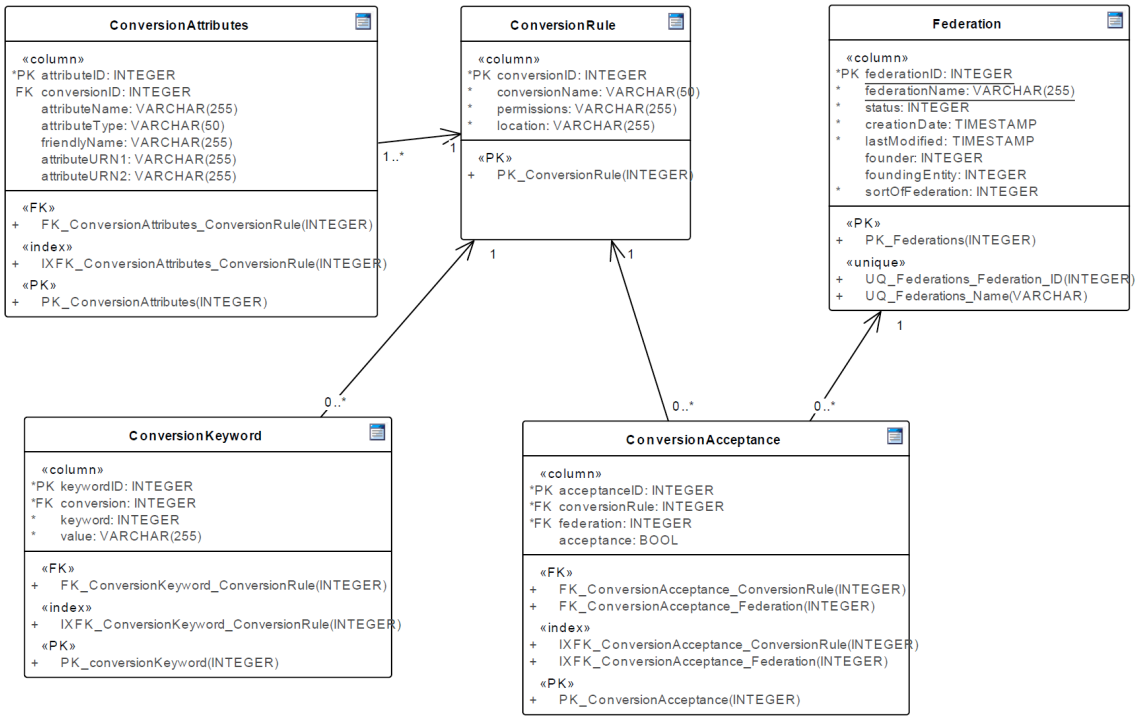

Abbildung 5.17.: Datenbank für die automatische Konvertierung

- `ConversionKeyword`: Setzt für eine konkrete Konvertierungsregel Werte in die Schlüsselwörter ein.

- `ConversionAttribute`: Beschreibt `source` und `target` für eine spezielle Konvertierungsregel.

Für die Konvertierungsregeln ist der `ConversionHandler` zuständig, der abhängig vom Stand des Workflows verschiedene andere Handler triggert. Damit eine generische Regel wiederverwendet werden kann, soll zunächst die Regel über den `ConversionUploadHandler` hochgeladen, validiert und diese über eine Funktion, `writeRuleToDb(file)`, in die Datenbank geschrieben werden. In der Datenbank befindet sich das generische Format, wodurch die Regel in die Formate der Implementierungen konvertiert werden kann. Die automatische Konvertierung in andere Formate geschieht IdP-seitig über den `ConversionImplementationHandler`. Eine Übersicht über die verschiedenen Handler und die wichtigsten Aufrufe bzw. Funktionen zeigt die Abbildung 5.18.

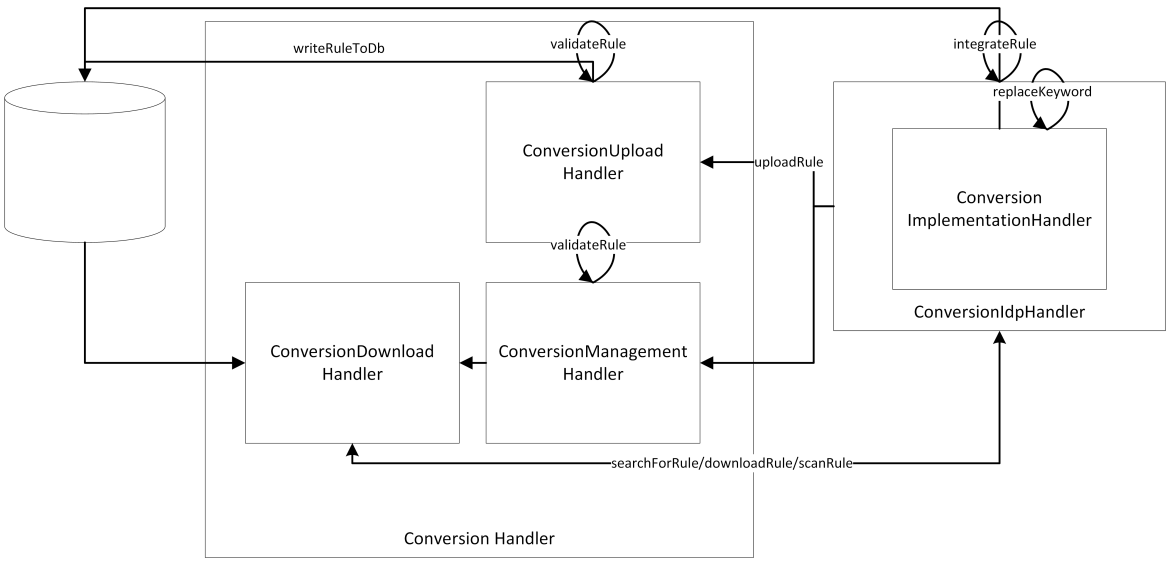

Abbildung 5.18.: Interner Aufbau des Conversion Rule Managements

Für eine möglichst automatische Konvertierung von Benutzerinformationen sind die folgenden Informationen in der Tabelle `ConversionRule` der Datenbank notwendig:

- Berechtigungen und Namen.

- Speicherung der speziellen Konvertierungsregel, wenn Skripte enthalten sind.

Da es mehrere Source-Attribute geben kann und auch das Target verschiedene Informationen enthält, werden diese in der Tabelle `ConversionAttribute` hinterlegt:

- ursprüngliches Attribut bzw. ursprüngliche Attribute zur Information: `source`.

- Ziel-Attribut: `target`.

Nachdem unterschiedliche Schlüsselwörter für eine Konvertierungsregel benötigt werden können, werden diese in der Tabelle `ConversionKeyword` gespeichert:

- Schlüsselwort,
- Wert des Schlüsselwortes und
- Link zur konkreten Konvertierungsregel.

Die Konvertierungsregeln können hochgeladen, geändert und gelöscht werden. Dies geschieht über die Ausnutzung der API, über die mit MdFIM durch HTTP Post und Request kommuniziert wird. Änderungen und Löschungen werden durch den `ConversionManagementHandler` behandelt. Die Regeln werden nach dem Hochladen durch MdFIM validiert. Dazu rufen der `ConversionManagementHandler` und der `ConversionUploadHandler` die Funktion `validateRule` auf, die die hochgeladene Regel überprüft. Hierbei wird überprüft, ob alle möglichen Schlüsselwörter ausgefüllt sind und ob die Regel bereits vorhanden ist. Ist die Überprüfung positiv, wird die Regel in die Datenbank gespeichert.

Ebenso wie beim Metadatenaustausch, sollen bei Konvertierungsregeln auch Fehler und andere Informationen durch HTTP Statuscodes im Body der Nachricht beschrieben werden. Die folgenden Statuscodes sind für erfolgreiche Operationen relevant:

- `201 Created`: Konvertierungsregel erstellt oder geändert.
- `202 Accepted`: Konvertierungsregel ist erfolgreich validiert.
- `202 Found`: Passende Konvertierungsregel gefunden.
- `300 Multiple Choices`: Mehrere Konvertierungsregeln gefunden.

Wenn mehrere Konvertierungsregeln gefunden wurden, wird zunächst auf Seiten des IdPs die Konfiguration überprüft, ob beispielsweise der Administrator in dem Fall informiert werden soll oder ob bestimmte Regeln bevorzugt werden sollen. Schlägt eine Aktion fehl, muss dies ebenso geloggt werden. Daneben ist eine an den Fehler angepasste Behandlung wichtig. Dies ist für die folgenden Statuscodes und Fehler dargestellt:

- `401 Unauthorized`: Fehlende Authentifizierung oder fehlende Berechtigungen. Die Aktion wird abgebrochen und der Nutzer darüber informiert.
- `404 Not Found`: Keine passende Konvertierungsregel gefunden. Dies wird dem Benutzer so weiter gegeben, so dass er eine neue Abfrage starten kann.
- `406 Not Acceptable`: Validierung der Konvertierungsregel ist fehlgeschlagen. Dies

wird ebenso dem Nutzer mitgeteilt, damit er die Konvertierungsregel anpasst und erneut hochlädt.

- `412 Precondition Failed`: Schlüsselwörter sind nicht oder nur teilweise ausgefüllt. In diesem Fall muss der Benutzer die fehlenden Schlüsselwörter ausfüllen.

- `423 Locked`: Konvertierungsregel ist durch einen anderen Benutzer gesperrt. Die Aktion soll so lange warten, bis die Konvertierungsregel durch den anderen Benutzer freigegeben wird. Hierbei soll der Benutzer die Aktion auch abbrechen können, falls die Datei längerfristig gesperrt ist.

IdP-seitige Umsetzung

Um überhaupt die Information zu erhalten, welche Attribute benötigt werden und ob bereits eine passende Konvertierung in der Konfiguration verfügbar ist oder ob eine geladen bzw. generiert werden muss, wird ein lokaler Handler bei IdP und AA benötigt. Der `ConversionIdpHandler` überprüft vorhandene Konvertierungsregeln, integriert heruntergeladene Regeln und kümmert sich um den Upload von neuen Regeln. Hierfür müssen die vorhandenen Attribute (`resolver:Dependency`) bei Shibboleth eindeutig über `name` und `friendlyName` beschrieben werden, wie Michael Grabatin in seiner Masterarbeit aufzeigte. Die Funktion `searchForRule` extrahiert die angeforderten Attribute aus den Metadaten des SPs. Wenn diese bereits in der `attribute-resolver.xml` vorhanden sind, werden keine zusätzlichen Konvertierungsregeln gebraucht. Ist dies nicht der Fall, fragt die Funktion MdFIM an. Der genaue Ablauf ist wie folgt:

Schritt 1: Die angefragten Attribute werden durch die Funktion `searchForRule` aus den Metadaten des SPs ausgelesen.

Schritt 2: Die angefragten Attribute werden durch die Funktion mit den in `attribute-resolver.xml` verfügbaren Attributen verglichen.

Schritt 3: Ist ein Attribut nicht lieferbar, wird MdFIM über die Funktion `queryRule` angefragt. Zunächst wird das benötigte, nicht verfügbare Attribut gefragt.

Schritt 4: MdFIM sendet daraufhin die möglichen Konvertierungsregeln in einer Übersicht.

Schritt 5: Die lokale Funktion `scanRule` überprüft, ob eine Konvertierungsregel verwendet werden kann basierend auf den ursprünglichen Attributen.

Schritt 6: Kann eine Konvertierungsregel automatisch generiert werden, wird diese durch die lokale Funktion `dowloadRule` geladen und in die Konfiguration über `integrateRule` eingefügt. Dazu wird die Datei, in der die Konvertierungsregeln stehen, geöffnet, der Abschnitt hinein kopiert und die Datei wieder geschlossen.

Schritt 7: Wird eine manuelle Anpassung der Regel benötigt, weil eine generische Regel zwar vorhanden, aber beispielsweise die `regex` nicht bekannt ist, muss der Administrator manuell sich die Regel anschauen, herunterladen und einfügen. Dies ist allgemein bei Regeln, die Skripte enthalten, der Fall. Das Informieren geschieht über E-Mail durch die Funktion `informAdmin`.

Schritt 8: Ist keine passende Konvertierungsregel vorhanden, wird die durch den Administrator manuell neu erstellte Regel über die Funktion `uploadRule` zu MdFIM hochgeladen, validiert und in die Datenbank eingetragen.

Die Automatisierung lässt sich, ähnlich wie bei den Metadaten, konfigurieren. Die Konfiguration der Konvertierungsregeln erfolgt in einer Konfigurationsdatei und wird über die Funktion `convConfig` vor der Integration abgefragt. Der Ablauf des Workflows wird durch eine einfache Workflow Engine überwacht.

Die Integration in die lokale Konfiguration erfolgt ebenfalls über eine Funktion, `integrateRule`. Soweit dies von der Implementierung erlaubt ist, sollen die durch MdFIM hinzugefügten Konvertierungsregeln in einer eigenen Datei stehen, auf die verwiesen wird. Dies stellt eine Art Komposition dar, die zudem den Parallelbetrieb erleichtert. So können selbst erstellte Konvertierungsregeln und die für MdFIM verwendeten unterschieden werden. Die Unterscheidbarkeit sowie der Aufbau ist erweiterbar für die Verwendung mehrerer MdFIM und zusätzlicher Werkzeuge. Jede hinzugefügte Konvertierungsregel wird ebenso in einer Logdatei vermerkt, um die Änderungen nachvollziehen zu können.

Die benötigten Templates sind passend zum IdP lokal gespeichert, wo die Schlüsselworte für einfache Konvertierungen anhand der Funktion

```
replaceKeywordConv(implementationID, conversionRuleID, conversionRuleSort)
```

ersetzt werden. Hierfür muss die generische Regel heruntergeladen und die darin enthaltenen Schlüsselworte in die Templates des IdPs eingefügt werden. Der `ConversionImplementationHandler` ist dafür zuständig, eine bereits vorhandene Konvertierungsregel in die entsprechende Implementierung zu übersetzen. Dies ist möglich, wenn bereits alle benötigten Schlüsselwörter bekannt sind. Im Nachfolgenden werden die generischen Regeln für die Implementierungen Shibboleth, SimpleSAMLphp und PySAML2 mit den entsprechenden Schlüsselwörtern aufgezeigt.

Die Umformung wird bei Shibboleth über XML durchgeführt. Als Schlüsselwörter werden `target`, `source` und zwei `targeturn` verwendet, um sowohl Umbenennung als auch die URN mitzuliefern, wie in Listing 5.18 zu sehen. Zur Verdeutlichung der Schlüsselwörter wird hier die Syntax der Templatesprache Jinja2 von Python verwendet.

```
<resolver:AttributeDefinition xsi:type="Simple"
  xmlns="urn:mace:shibboleth:2.0:resolver:ad" id="{{ target }}"
  sourceAttributeID="{{ source }}">
  <resolver:Dependency ref="{{ source | resource }}" />
  <resolver:AttributeEncoder xsi:type="SAML1String"
   xmlns="urn:mace:shibboleth:2.0:attribute:encoder"
   name="{{ targeturn1 }}"/>
  <resolver:AttributeEncoder xsi:type="SAML2String"
   xmlns="urn:mace:shibboleth:2.0:attribute:encoder"
   name="{{ targeturn2 }}" friendlyName="{{ target }}" />
</resolver:AttributeDefinition>
```

Listing 5.18: Umbenennung eines Attributs als generische Shibboleth-Konvertierung

Merging von Attributen geschieht über ein `Script`, die ein `target` und mehrere `source` enthält (vgl. Listing 5.19).

```
<resolver:AttributeDefinition id="target" xsi:type="Script"
 xmlns="urn:mace:shibboleth:2.0:resolver:ad">
    <resolver:Dependency ref="{{ source1 }}" />
    <resolver:Dependency ref="{{ source2 }}" />
    <resolver:AttributeEncoder xsi:type="SAML1String"
          xmlns="urn:mace:shibboleth:2.0:attribute:encoder"
          name="{{ targeturn1 }}"/>
    <resolver:AttributeEncoder xsi:type="SAML2String"
          xmlns="urn:mace:shibboleth:2.0:attribute:encoder"
          name="{{ targeturn2 }}" friendlyName="{{ target }}" />
    <Script><![CDATA[
       importPackage(Packages.edu.internet2.middleware.shibboleth.common.
            attribute.provider);
       {{ target }} = new BasicAttribute("{{ target }}");
       merge = {{ source1 }}.getValues().get(0) + " "
           + {{ source2 }}.getValues().get(0);
       {{ target }}.getValues().add(merge);
    ]]></Script>
</resolver:AttributeDefinition>
```

Listing 5.19: Merging von Attributen als generische Shibboleth-Konvertierung

Zum Splitten wird eine regex verwendet. Damit diese passend eingefügt werden kann, ist `regex` ein zusätzliches Schlüsselwort, wie in Listing 5.20 dargestellt.

```
<resolver:AttributeDefinition xsi:type="RegexSplit"
 xmlns="urn:mace:shibboleth:2.0:resolver:ad" id="{{ target }}"
 sourceAttributeID="{{ source }}"
 regex="{{ regex }}">
  <resolver:Dependency ref="{{ source | resource }}" />
   <resolver:AttributeEncoder xsi:type="enc:SAML1ScopedString"
   name="{{ targeturn1 }}" />
  <resolver:AttributeEncoder xsi:type="enc:SAML2ScopedString"
   name="{{ targeturn2 }}" friendlyName="{{ target }}" />
</resolver:AttributeDefinition>
```

Listing 5.20: Splitten eines Attributs als generische Shibboleth-Konvertierung

Ebenso kann ein scoped-Attribut erstellt werden. Im Gegensatz zu einer einfachen Umbenennung ist zudem das Schlüsselwort `scope` vorhanden, um den Scope näher zu beschreiben (vgl. Listing 5.21).

```
<resolver:AttributeDefinition xsi:type="ad:Scoped" id="{{ source }}_scope"
scope="{{ scope }}" sourceAttributeID="{{ source }}">
<resolver:Dependency ref="{{ source | resource }}" />
  <resolver:AttributeEncoder xsi:type="enc:SAML1ScopedString"
   name="{{ targeturn1 }}" />
  <resolver:AttributeEncoder xsi:type="enc:SAML2ScopedString"
   name="{{ targeturn2 }}" friendlyName="{{ target }}" />
</resolver:AttributeDefinition>
```

Listing 5.21: Scopen eines Attributes als generische Shibboleth-Konvertierung

Eine generische Transformation ist mit Listing 5.22 möglich. Durch das zusätzliche Schlüsselwort `conversion` ist zumindest bei der ersten Verwendung eine manuelle Eingabe durch den Administrator notwendig. Da diese Umformung Skript benötigt, soll diese Regel manuell vom Administrator heruntergeladen und eingefügt werden.

```
<resolver:AttributeDefinition id="{{ source }}" xsi:type="Script"
 xmlns="urn:mace:shibboleth:2.0:resolver:ad" sourceAttributeID="{{ source }}">
 <resolver:Dependency ref="{{ source | resource }}" />
  <resolver:AttributeEncoder xsi:type="SAML1String"
   xmlns="urn:mace:shibboleth:2.0:attribute:encoder" name="{{ targeturn1 }}" /
   >
  <resolver:AttributeEncoder xsi:type="SAML2String"
   xmlns="urn:mace:shibboleth:2.0:attribute:encoder" name="{{ targeturn2 }}"
   friendlyName="{{ target }}" />
  <Script><![CDATA[
    importPackage(Packages.edu.internet2.middleware.shibboleth.common.
        attribute.provider);
    // Create attribute to be returned from definition
    target = new BasicAttribute("{{ target }}");
        source = new BasicAttribute("{{ source }}");
    // Convert first value of source attribute, if available
    if (typeof source != "undefined" && source != null &&
        source.getValues().size() > 0) {
      value = new String(source.getValues().get(0));
      target.getValues().add(value.replace({{ conversion }}));
    }
  ]]></Script>
</resolver:AttributeDefinition>
```

Listing 5.22: Transformieren eines Attributs als generische Shibboleth-Konvertierung

Die Umbenennung in SimpleSAMLphp geschieht über eine Konfigurationsdatei, die PHP verwendet. Die Art der Umformung steht in Zeile 3, `core:AttributeMap`. Als Schlüsselworte werden, wie in Listing 5.23 dargestellt, `target` und `source` verwendet.

```
'authproc' => array(
        50 => array(
                'class' => 'core:AttributeMap',
                '{␣target␣}}' => '{{␣source␣}}',
        ),
),
```

Listing 5.23: Umbenennung eines Attributs als generische SimpleSAMLphp-Konvertierung

Das Mergen sieht ähnlich aus, dargestellt durch Listing 5.24. Im Gegensatz zur einfachen Umbenennung werden hier mehrere Source-Attribute eingesetzt. Alternativ kann auch die Methode `core:AttributeAdd` verwendet werden.

```
'authproc' => array(
    50 => array(
        'class' => 'core:AttributeMap',
        '{␣target␣}}' => array('{{␣source1␣}}', '{␣source2␣}}'),
    ),
),
```

Listing 5.24: Mergen von Attributen als generische SimpleSAMLphp-Konvertierung

Der Scope wird über die folgende generische Konvertierung für SimpleSAMLphp hinzugefügt (vgl. Listing 5.25).

```
'authproc' => array(
    50 => array(
        'class' => 'core:ScopeFromAttribute',
        'sourceAttribute' => '{{ source }}'
        'targetAttribute' => '{{ target }}'
    ),
),
```

Listing 5.25: Scopen eines Attributes als generische SimpleSAMLphp-Konvertierung

Bei Splitten kann innerhalb von SimpleSAMLphp `core:AttributeAlter` in zwei Variationen eingesetzt werden (vgl. Listing 5.26). Folglich muss in diesem Fall neben `source`, `pattern` und `target`, falls nicht derselbe Attributsname verwendet werden soll, eine Unterscheidung zwischen `replace` und `remove` unternommen werden.

```
'authproc' => array(
        10 => array(
                'class' => 'core:AttributeAlter',
                'subject' => '{{ source }}',
                'pattern' => '{{ pattern }}',
                'target' => '{{ target }}',
                '%replace',
        ),
        10 => array(
                'class' => 'core:AttributeAlter',
                'subject' => '{{ source }}',
                'pattern' => '{{ pattern }}',
                '%remove',
        ),
),
```

Listing 5.26: Splitten eines Attributes als generische SimpleSAMLphp-Konvertierung

Bei PySAML2 ist nur eine Umbenennung vorgesehen (vgl. Listing 5.27).

```
MAP = {
    "identifier": "urn:oasis:names:tc:SAML:2.0:attrname-format:basic",
    "fro": {
        '{{ target }}': '{{ source }}',
        },
    "to": {
        '{{ source }}': '{{ target }}',
    }
}
```

Listing 5.27: Mapping von Attributen als generische PySAML2-Konvertierung

Das Konkatenieren von Strings wird in Python durch den Operator + gemacht. Dies wird für Konvertierungsregeln genutzt (vgl. Listing 5.28).

```
MAP = {
    "identifier": "urn:oasis:names:tc:SAML:2.0:attrname-format:basic",
    "fro": {
        '{{ target }}': '{{ source1 }}'+'{{ source2 }}',
        },
    "to": {
        '{{ source1 }}'+'{{ source2 }}': '{{ target }}',
    }
}
```

Listing 5.28: Mergen von Attributen als PySAML2-Konvertierung

Das Splitten von Strings ist in Python durch die Operation `result.split(pattern, string)` möglich. Übertragen auf die Schlüsselwörter für Konvertierungsregeln sieht das wie folgt aus (vgl. Listing 5.29).

```
{{ target }}.split({{ pattern }}, {{ source }})
```

Listing 5.29: Splitten von Attributen als PySAML2-Konvertierung

Durch den `ConversionImplementationHandler`, der eine Funktion

`replaceKeywordConv(implementation, conversionID)`

aufruft, werden die Schlüsselwörter durch die Werte in der generischen Regel ersetzt. Unterschiedliche Regeln einer Implementierung können miteinander kombiniert und somit modular eingesetzt werden. Der `ConversionDownloadHandler` ist für das Herunterladen der Regel zuständig, während der `ConversionSyncHandler` dafür sorgt, dass die Konvertierungsregeln immer aktuell sind.

Scheitert die Integration, muss entsprechend des Fehlers bzw. der Ursache gehandelt werden. Je nachdem ist ein erneuter Versuch oder ein Abbruch inklusive des entsprechenden Logeintrages optimaler. Nachfolgend werden häufige Probleme beschrieben:

- Fehlen die entsprechenden Berechtigungen, soll dies entsprechend geloggt und der Vorgang abgebrochen werden.
- Ist die benötigte Datei aktuell gesperrt, soll es mit einer Zeitverzögerung erneut probiert werden.
- Sind die Dateien von MdFIM fehlerhaft, hängt die Vorgehensweise von der Prüfsumme ab. Stimmen Prüfsumme und Datei nicht überein, soll sie erneut heruntergeladen und überprüft werden. Sind andere Fehler zu finden, wird der Vorgang abgebrochen und das Ereignis geloggt.

- Fehlt der Speicherplatz, muss die Aktion zurückgerollt werden.
- Ist die Datei bereits, trotz vorheriger Überprüfung, vorhanden, wird der Vorgang ebenfalls abgebrochen.

Gelingt die Aktion nicht, muss der Administrator ebenso informiert werden. In den Logdateien wird dies über den HTTP Statuscode `500 Internal Server Error` sowie der Datei bzw. die Aktion, die fehlgeschlagen ist, angezeigt.

Bezogen auf das Organisationsmodell und deren Umsetzung in Software, zeigt sich, dass alle Aktionen bei MdFIM durch die Rolle TTP-Konvertierungsregel (Conv) getätigt werden. Die technische Interaktion auf Seiten von IdP und AA erfolgt ebenfalls durch die Rolle Conv (IdP-Conv und AA-Conv), während Konfiguration, Verwaltung von eigenen Regeln und die Zustimmung zu einer Regel durch einen entsprechenden Administrator (IdP-A bzw. AA-A) geschehen. Die Validierung der Konvertierungsregel durch die Föderation wird von der Rolle Fed-ConM ausgeführt.

SP-seitige Umsetzung

Um eine möglichst automatische Konvertierung zu gewährleisten, benötigt es auf Seiten der SPs eine Erweiterung der Metadaten. Bisher können SPs angeben, welche Attribute sie unbedingt und optional benötigen. Diese angefragten Attribute sind UND-verknüpft. Eine ODER-Verknüpfung wird zwar in der Praxis verwendet, ist aber bisher technisch nicht vorgesehen. Laut SAML [`SAML2Meta`] [CMPM05] können Attribute durch das folgende Schema (vgl. Listing 5.30) in den Metadaten angefragt werden:

```
1 <element name="RequestedAttribute" type="md:RequestedAttributeType"/>
2 <complexType name="RequestedAttributeType">
3    <complexContent>
4      <extension base="saml:AttributeType">
5        <attribute name="isRequired" type="boolean" use="optional"/>
6      </extension>
7    </complexContent>
8 </complexType>
```

Listing 5.30: Definition von RequestedAttribute

Dies sieht beispielsweise wie in Listing 5.31 in den Metadaten aus. Mehrere Attribute können hierbei UND-verknüpft werden, indem sie hintereinander als `md:RequestedAttribute` stehen.

```
<md:RequestedAttribute isRequired="false"
    NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
    Name="urn:oid:2.5.4.42"
    FriendlyName="givenName"/>
<md:RequestedAttribute isRequired="true"
    NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
    Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.7"
    FriendlyName="eduPersonEntitlement">
    <saml:AttributeValue
        xsi:type="xs:anyURI">
                https://www.lrz.de/entitlement
            </saml:AttributeValue>
</md:RequestedAttribute>
```

Listing 5.31: Beispiel von RequestedAttribute

Eine ODER-Verknüpfung ist durch die Spezifikation in [SAML2Meta] [CMPM05] nicht vorgesehen. Um diese gebräuchliche Verknüpfung auch zu automatisieren, muss die Definition der Metadaten erweitert werden. Eine ODER-Verknüpfung kann zum Beispiel wie in Listing 5.32 aussehen.

```
<md:RequestedAttribute isRequired="true" isOr="true" name="id">
  <md:RequestedAttributeOr
      NameFormat="urn:mace:dir:attribute-def:uid"
      Name="urn:oid:0.9.2342.19200300.100.1.1"
      FriendlyName="uid"/>
  <md:RequestedAttributeOr
      NameFormat="urn:oasis:names:tc:SAML:2.0:nameid-format:persistent"
      Name="urn:oid:1.3.6.1.4.1.5923.1.1.1.10"
      FriendlyName="eduPersonTargetedID"\>
</md:RequestedAttribute>
```

Listing 5.32: Beispiel von ODER-verknüpften RequestedAttribute

Damit diese ODER-Verknüpfung auch definiert ist, wird die oben beschriebene Spezifikation um ein Kindelement `RequestedAttributeOr` erweitert. Zudem wird das Attribut `isOr` eingeführt, um eine ODER-Verknüpfung anzuzeigen. Wird `isOr` nicht verwendet, wird von einer UND-Verknüpfung ausgegangen. Das neu hinzugekommene Attribut `name` ist optional und dient der Unterscheidung verschiedener ODER-Verknüpfungen. Daraus ergibt sich das Listing 5.33.

```
<element name="RequestedAttribute" type="md:RequestedAttributeType">
<complexType name="RequestedAttributeType">
   <complexContent>
     <extension base="saml:AttributeType">
       <attribute name="isRequired" type="boolean" use="optional"/>
           <attribute name="isOr" type="boolean" use="optional"/>
           <attribute name="name" type="string" use="optional"/>
           <sequence>
             <element ref="RequestedAttributeOr" minOccurs="0">
           </sequence>
     </extension>
   </complexContent>
</complexType>
```

Listing 5.33: Definition von ODER-verknüpftem RequestedAttribute

Durch die Erweiterung der Metadaten von [SAML2Meta] [CMPM05] können nun auch ODER-Verknüpfungen automatisch, durch ein Skript, bearbeitet werden.

5.3.3. Bewertung

Die Konvertierungsregeln sind durch das Conversion Rule Management wiederverwendbar. Gleichzeitig können sie großteils automatisiert in die Konfiguration eingefügt werden, während der Grad der Automatisierung konfigurierbar ist. Zudem kann eine Art Qualitätskontrolle für die Konvertierungsregeln eingesetzt werden. Zwar werden nicht alle Regeln automatisch erstellt, aber es eine Verbesserung zum Ist-Zustand. Soweit weitere Definition für die Implementierungen verfügbar sind, können diese in das Conversion Rule Management eingefügt und genutzt werden. Insgesamt werden durch das Conversion Rule Management folgende Funktionalitäten erbracht:

- Hochladen, Ändern, Löschen von Konvertierungsregeln.
- Validieren von Konvertierungsregeln.
- Wiederverwenden von Konvertierungsregeln durch andere Entitäten.
- Implementierungsunabhängigkeit bzw. Formatierung in das benötigte Format.
- Automatische Integrieren von Konvertierungsregeln.
- Konfiguration des Automatisierungsgrades, damit auch halbautomatische Integration möglich ist.
- Logging.
- Validierung von Konvertierungsregeln durch Föderationen.

Tabelle 5.5 stellt die Erfüllung der Anforderungen im Vergleich mit bisherigen Ansätzen dar.

Anforderung	aktuelle Praxis	eCCS	FSCS	Ontologien	Repository
[Konv-Wiederverwendbarkeit]	-	+	+	+	+
[Konv-Automatisierung]	-	-	-	-	+
[Konv-Abdeckung]	+	-	+	+	+
[Konv-Implementierungsunabhängigkeit]	+	-	+	o	+
[Konv-Konvertierungen]	+	+	+	+	+
[Konv-Modularität]	o	-	+	o	+
[Konv-Qualität]	-	+	-	-	+

Tabelle 5.5.: Bewertung der entwickelten Lösung im Vergleich zu den Ansätzen zur Konvertierung

5.3.4. Anwendung

Um das Conversion Rule Management zu verdeutlichen, wird das Werkzeug am Beispiel des IdPs Leibniz-Rechenzentrum angewandt. Das LRZ verwendet die SAML-Implementierung Shibboleth. Durch den Einsatz von Shibboleth können die meisten Konvertierungsregeln automatisch erzeugt werden. Neben der nationalen Föderation DFN-AAI, nimmt das LRZ an der internationalen Inter-Föderation eduGAIN teil. Während eduGAIN keine Konvertierungsregeln verwaltet, wird dieses Werkzeug intensiv durch die DFN-AAI eingesetzt. MdFIM wurde so implementiert, dass bei der verzögerten Nutzbarkeit eines Dienstes der Nutzer informiert wird, sobald der Dienst vollständig nutzbar ist. Falls dies nicht möglich ist, beispielsweise weil Konvertierungsregeln fehlen oder die benötigte Regel Skripte enthält, wird der Nutzer ebenso informiert.

Ein Mitarbeiter des IdPs LRZ will einen neuen Dienst nutzen. Nach dem Metadatenaustausch werden die benötigten Konvertierungsregeln ermittelt. Nachdem der Dienst zwei Attribute vorschreibt, die das LRZ nicht liefern kann, wird MdFIM nach passenden Regeln gefragt. Für eines der Attribute, `mailName`, ist bereits eine generische Konvertierungsregel vorhanden. Diese Regel besteht aus dem target `mailName` und den Source-Attributen `sn` und `givenName`. Da basierend aus diesen Informationen eine Regel automatisch erzeugt werden kann, wie in Listing 5.34 zu sehen, ist eine Erstellung möglich. Da diese Regel jedoch Skript enthält, wird der Administrator informiert, der diese Regel nach einer manuellen Überprüfung herunterlädt und in die Konfiguration einfügt.

```
<resolver:AttributeDefinition id="mailName" xsi:type="Script"
 xmlns="urn:mace:shibboleth:2.0:resolver:ad">
    <resolver:Dependency ref="sn" />
    <resolver:Dependency ref="givenName" />
    <resolver:AttributeEncoder xsi:type="SAML1String"
        xmlns="urn:mace:shibboleth:2.0:attribute:encoder" name="{{ targeturn1
     }}"/>
    <resolver:AttributeEncoder xsi:type="SAML2String"
        xmlns="urn:mace:shibboleth:2.0:attribute:encoder" name="{{ targeturn2
     }}"
        friendlyName="mailName" />
    <Script><![CDATA[
       importPackage(Packages.edu.internet2.middleware.shibboleth.common.
           attribute.provider);
       mailName = new BasicAttribute("mailName");
       merge = sn.getValues().get(0) + " " + givenName.getValues().get(0);
       mailName.getValues().add(merge);
    ]]></Script>
```

Listing 5.34: Merging von Attributen als generische Shibboleth-Konvertierung

Für das zweite, neu hinzugekommene Attribut `personalDisplayName` ist noch keine Regel vorhanden. Folglich wird der Administrator über E-Mail über eine fehlende Konvertierungsregel informiert. Da `personalDisplayName` vermutlich eine interne Bezeichnung ist, erstellt der Administrator die folgende Regel (vgl. Listing 5.35).

```
<resolver:AttributeDefinition xsi:type="Simple"
 xmlns="urn:mace:shibboleth:2.0:resolver:ad"
 id="personalDisplayName" sourceAttributeID="displayName">
<resolver:Dependency ref="displayName" />
  <resolver:AttributeEncoder xsi:type="enc:SAML1ScopedString"
   name="{{ targeturn1 }}" />
  <resolver:AttributeEncoder xsi:type="enc:SAML2ScopedString"
   name="{{ targeturn2 }}" friendlyName="personalDisplayName" />
</resolver:AttributeDefinition>
```

Listing 5.35: Umbenennung eines Attributs als generische Shibboleth-Konvertierung

Der Nutzer wird über die nun mögliche Nutzung des Dienstes per E-Mail informiert. Automatisch über das Skript wird diese Regel zur Umbenennung hochgeladen, im Repository abgelegt und die Information über `source` und `target` in der Datenbank gespeichert. Nachdem diese Regel die DFN-AAI betrifft, wird der zuständige Administrator informiert. Dieser akzeptiert diese Regel und fragt gleichzeitig den SP nach den noch benötigten Encodings an, die ergänzend eingetragen werden. Kurz darauf will ein Nutzer einer anderen Universität, beispielsweise von der Universität Ulm, ebenfalls den Dienst verwenden. Da nun alle benötigten Regeln vorhanden sind, kann der Nutzer unmittelbar den Dienst nutzen.

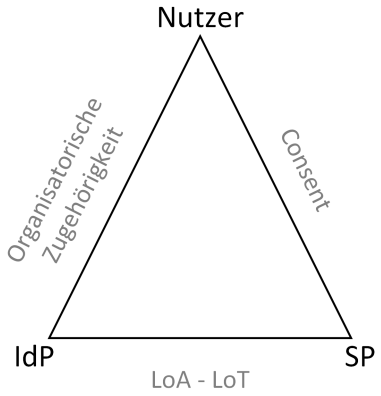

Abbildung 5.19.: Vertrauen zwischen den Akteuren

5.4. Trust Management

Das bisher als Black Box beschriebene Trust Management behandelt das Thema Vertrauen. Während in Latifa Boursas [Bou09] Dissertation dynamisches Vertrauen über Trust Based Access Control (TBAC) im Vordergrund steht, geht es hier um ein eher statisches Konzept, was allerdings um Latifa Boursas Ansatz erweitert werden kann. In einem statischen Konzept müssen verschiedene Sichtweisen und Akteure beachtet werden, wodurch sich die folgenden Fragen stellen:

- Wie sieht ein Nutzer, dass er dem SP vertrauen kann?
- Wie sieht ein Nutzer, dass er seinem IdP vertrauen kann?
- Wie sieht ein SP, dass er dem IdP vertrauen kann?
- Wie sieht ein SP, dass er dem Nutzer vertrauen kann?
- Wie sieht ein IdP, dass er dem SP vertrauen kann?
- Wie sieht ein IdP, dass er seinem Nutzer vertrauen kann?

Die unterschiedlichen Vertrauensabhängigkeiten sind in Abbildung 5.19 dargestellt. Sampath et al. [SG06] bezeichnen dieses Geflecht an Vertrauensbeziehungen als *System trust*. Die erste Frage kann durch Consent beantwortet werden, wobei hier nur sichtbar ist, welche Attribute an den SP gesendet werden. Eine zusätzliche Visualisierung, welche Attribute üblich für diese Art von Dienst sind, wäre sinnvoll. Dafür wird zunächst eine Kategorisierung erforderlich. Anhand dieser Einordnung können dann übliche angefragte Attribute ermittelt werden. Eine große Abweichung vom Standard kann entweder auf zu viel abgefragte Benut-

zerinformationen oder einen einfachen Dienst hinweisen. Die letztliche Entscheidung bleibt jedoch dem Nutzer überlassen. Diese Art von Visualisierung kann zusammen mit der Usability von Consent und Discovery im Rahmen einer Nachfolgearbeit genauer analysiert werden. Ein SP sieht im Gegenzug anhand der Attribute und des IdPs, inwiefern der Nutzer vertrauenswürdig ist. SPs können IdPs durch ein *Level of Assurance* einordnen und ihnen dadurch in gewissen Rahmen vertrauen. Wie bereits in Kapitel 3 aufgezeigt, erfüllen die meisten LoAs den Anforderungen der Föderationen und Communities nicht oder nur bedingt, weswegen diese Lösungen nur als suboptimal gelten. Folglich soll nachfolgend im Abschnitt 5.4.1 ein Konzept eines LoA entwickelt werden, was den Anforderungen entspricht, aber auch außerhalb der jetzigen Föderationen eingesetzt werden kann. Dieses Konzept soll vor allem dazu dienen, verschiedene Level of Assurance vergleichbar zu machen. Für die Einordnung von SPs gibt es bisher keine geeignete statische Einordnung, wodurch diese als *Level of Trust* bezeichnete Kategorisierung in Abschnitt 5.4.2 eingeführt wird. Sowohl für LoA als auch LoT ist die Risikobewertung wichtig, damit die jeweilige Organisation einen Mindestwert angeben kann. Die Beziehung zwischen IdP und Nutzer ist geschäftlicher Natur. In den meisten Fällen sind sie innerhalb einer Organisation angesiedelt, wodurch ihr gegenseitiges Vertrauen in dieser Arbeit vorausgesetzt wird.

Bei der Spezifizierung eines LoA bzw. LoT soll beachtet werden, dass trotz der Automatisierung eine gewisse Sicherheit gewährleistet wird. Diese soll durch eine einfache Art von Vertrauen für alle Parteien möglich sein, wobei Gewichtungen möglich sein sollen. Die Auswirkung durch Föderationsverwaltung soll ebenso erläutert werden. Somit die folgenden Teilaspekte beachtet werden:

- Einschätzung von IdP und SP,
- Vergleich von Verlässlichkeitsklassen,
- Feingranulare Einteilung,
- Parallelität von mehreren Verlässlichkeitsklassen pro Organisation und
- Einordnung von neuen Mitgliedern ermöglichen.

Neben der inhaltlichen Spezifikation von Trust Levels ist die technische Spezifikation relevant. Hier muss geklärt werden, wie ein Trust Level dargestellt werden kann. Dabei müssen die Protokolle OpenID Connect und SAML betrachtet werden, um möglichst eine Spezifikation zu wählen, die leicht durch die beiden Protokolle angewandt werden kann. Ferner spielen weitere Aspekte im Rahmen des Trust Managements eine Rolle:

- Welche Werkzeuge werden für das Trust Management benötigt?
- Welche Funktionalitäten muss MdFIM bieten, um ein effizientes Trust Management zu ermöglichen?

- Welche Erweiterungen benötigen IdP und SP?

Diese Fragen werden im Folgenden ebenfalls beantwortet.

Johannes Meier [Mei15] hat in seiner Bachelorarbeit über die Evaluation und Konzeption von Verlässlichkeitsklassen für den Einsatz in föderierten Umgebungen geschrieben. Das Thema wurde von der Autorin dieser Arbeit vorgeschlagen und zusammen mit einem Kollegen betreut. Nachdem in dieser Arbeit nicht alle Anforderungen und Randbedingungen beachtet wurden, wird das Trust Management in dieser Arbeit von Grund auf neu konzipiert. Das Grundkonzept für dynamisches Vergleichen von LoA in ihren Aspekten wurde in der Veröffentlichung [GHMP16] erläutert.

5.4.1. Level of Assurance

Auch wenn Level of Assurance wenig bis nichts über das tatsächliche vorhandene Vertrauen zwischen zwei Entitäten aussagt, dienen LoAs dazu IdPs in eine bestimmte Klassifikation statisch einzuordnen und dementsprechend den Nutzern die Verwendung eines Dienstes zu erlauben oder diese abzulehnen. Dies ist abhängig von der Risikoeinschätzung eines Dienstes. Durch die Beschreibung der Verlässlichkeitsklasse in den Metadaten bzw. als Attribut in einer Assertion werden keine weiteren Daten übertragen, wodurch auch kein Bottleneck entstehen kann im Gegensatz zu dynamischen Varianten. Zudem können auch neue Mitglieder eingeordnet werden. Bisher bestehen verschiedene Verlässlichkeitsklassen der Föderationen parallel zu den Normen. Im Folgenden werden zunächst die Anforderungen kurz wiederholt, bevor auf die Bewertung der Verlässlichkeitsklassen eingegangen wird. Der Aufbau der Klassifikation bildet den Grundstein für eine eigene Klassifikation. Im Anschluss wird eine eigene Klassifikation erstellt, die anhand eines Beispiels angewandt wird.

Anforderungen

Die Anforderung [FA-LoA] wurde mit Priorität 2 bewertet und besteht aus mehreren Unter-Anforderungen. Durch [FA-LoA] soll dem Service Provider angezeigt werden, welche Datenqualität der Identity Provider liefern kann. Zugleich soll die Einteilung in eine bestimmte Klasse transparent erfolgen. Falls unterschiedliche Klassifikationen verwendet werden, sollen diese vergleichbar werden können. Wie bereits in Kapitel 2 beschrieben, unterteilt sich die Anforderung [FA-LoA] in die folgenden Teilaspekte, die bei einer Konzeption beachtet werden müssen:

- [LoA-Auditing],
- [LoA-Automatisierung],
- [LoA-Einordnung],

- [LoA-Gewichtung],
- [LoA-Implementierungsunabhängigkeit],
- [LoA-Koexistenz],
- [LoA-Konfiguration],
- [LoA-Mapping],
- [LoA-Protokollunabhängigkeit],
- [LoA-Realisierbarkeit],
- [LoA-Registrierung] und
- [LoA-SelfAsserted].

Bewertung der Level of Assurance

Die Einordnung der Föderationen der NRENs in die LoA der Normen ist teils nicht eindeutig, da nicht alle Informationen gegeben sind. Einzig bei InCommon ist die Einordnung, insbesondere in NIST und STORK QAA, eindeutig, da sich die beiden Profile Bronze und Silver an dem NIST Standard orientieren, der wiederum STORK QAA inspiriert hat. Während STORK QAA die geringsten Voraussetzungen für QAA 2 hat, ist bei NIST LoA die Stufe 2 nur für InCommon Silver möglich, wie in Tabelle 5.6 zu sehen. Nachdem nicht immer auf Information Security Management geachtet wird, ist hier ebenfalls eine niedrige Einordnung zu erkennen.

	Norm			
NREN	NIST LoA	STORK QAA	ISO/IEC 29115	Kantara
DFN-AAI	bis 1	2	1	1-2
InCommon	1/2	1/2	1/2	1-2
Haka	1	2	2	1-2
SWAMID	1	2	1	1-2
UK federation	bis 1	1	1	1

Tabelle 5.6.: Einordnung der NREN Föderationen in die LoA der Normen

Beim Vergleich der Normen mit der Anwendung von Level of Assurance in den ausgewählten Föderationen ist festzustellen, dass

- die Föderationen allgemein mehr Wert auf die organisatorischen Abläufe, sichere Software und Hardware legen. In diesem Zusammenhang fällt auf, dass ISO/IEC 29115:2013 als einzige Norm Security Management betrachtet.

- zusätzlich der Datenschutz in den Vorgaben der Föderationen wichtiger ist als in den Normen.
- Normen verstärkt auf technischen Schutz gegen Angriffe setzen.
- Kantara einen Schwerpunkt auf fest definierte organisatorische Abläufe legt.

eIDAS legt höhere Anforderungen als die anderen Normen und kann nicht durch jede Föderation abgedeckt werden. VoT bietet dahingegen einen flexiblen Aufbau, betrachtet jedoch nicht alle Aspekte, die die LoAs der Föderationen als wichtig erachten.

Die Ansätze der Forschung, beschrieben in Abschnitt 3.4, konzentrieren sich zum einen auf dynamische Verfahren, zum anderen setzen sie entweder kein Vertrauen oder auf eine vereinfachte Version von NIST. Während dynamische Verfahren ein Bottleneck darstellen können und daher nicht allein eingesetzt werden sollen, spiegeln andere Ansätze die Verhältnisse nicht wieder und erfüllen somit auch nicht die gestellten Anforderungen. Bei Betrachtung der oben aufgestellten Anforderungen wird insbesondere [LoA-Gewichtung] von keinem LoA eingehalten. Somit wird ein eigenes Level of Assurance erstellt.

Inhaltlicher Aufbau eines Level of Assurance

Während Normen vier verschiedene Levels aufweisen, sind die Verlässlichkeitsklassen der Föderationen meist auf 2 Levels beschränkt. Um die Anforderung [LoA-Gewichtung] zu unterstützen, wird zunächst der Aufbau betrachtet. Es ist nirgendwo beschrieben, wieso Verlässlichkeitsklassen genau diese Anzahl an Levels aufweisen und wieso ganze Levels verwendet werden. Daher werden nachfolgend die Entscheidungsfindung und Charakteristika beschrieben, um zukünftig anhand dieser Arbeit eine Anleitung zu bieten.

Zunächst werden unterschiedliche Metriken betrachtet. Es gibt qualitative und quantitative Metriken, wobei quantitativ in diskret und kontinuierlich unterschieden wird. Diskrete Werte haben den Vorteil, dass

- sie leichter verglichen werden können als kontinuierliche Werte.
- sie leichter angegeben werden können.
- es eine endliche, überschaubare Anzahl an möglichen Werten gibt.

Dies ist für die Angabe des Vertrauens wichtig. Dabei ist zu beachten, dass es so viele Werte bzw. Stufen geben soll, dass die Einordnung möglichst genau möglich ist. Ebenso sollen die Anforderungen an den Gegenüber möglichst genau wiedergegeben werden. Es sollen jedoch nicht zu viele Stufen oder Werte zur Verfügung gestellt werden, damit eine Übersichtlichkeit und eine leichte Einordnung noch möglich ist. Es bietet sich eine gerade Anzahl an Stufen an, damit die Entscheider tätig werden müssen und nicht die mittlere Lösung wählen können.

In Hinblick auf die vorhandenen Normen haben sich 4 Stufen etabliert und als benutzbar gezeigt.

Eine Klassifikation soll die Stufen/Werte für mehrere Aspekte vergeben. Dies ist ähnlich wie der Ansatz VoT, an dem die Autorin mitarbeitet. Um eine Gewichtung oder eine feine Einteilung für die Föderationen zu erlauben, ergeben sich verschiedene Möglichkeiten:

- Zwischenstufen, um die einzelnen Abstufungen zu erlauben. Dies erlaubt jedoch nur bedingt eine Gewichtung und wird nachfolgend nicht weiter betrachtet.
- Maturity Levels einführen, was Abstufungen und Gewichtungen erlaubt.
- Levels als Art Vektoren aufstellen, womit jeder einzelne Aspekt separat bewertet werden kann, was wiederum beides, also Gewichtung und Einteilung, erlaubt.

Christian Richter [Ric13] hat in seiner Dissertation Reifegradmodelle für Werkzeuglandschaften zur Unterstützung von IT Service Management (ITSM)-Prozessen entwickelt. Das Resultat ist eine gewichtete Liste mit Anforderungen an die Werkzeuglandschaft, um eine möglichst optimale Prozessunterstützung zu erhalten. Durch eine Priorisierung muss die Werkzeuglandschaft nicht in einem Schritt angepasst werden, sondern kann sukzessive vorgenommen werden. Das Prinzip kann für LoA übertragen werden. So sind grundlegende Anforderungen, untergliedert in verschiedene Aspekte, höher gewichtet, als spezielle und höhere Anforderungen, die beispielsweise nur für Finanzinstitute essentiell sind. Durch eine Priorisierung kann der eigene LoA schrittweise erhöht werden. Zudem ist es möglich verschiedene Stufen pro Föderation festzulegen, die zu erreichen sind.

Die Darstellung in einem Vektor hat ähnliche Vorteile wie Maturity Level. Beispielsweise können 4 Aspekte als `(x1.x2.x3.x4)` dargestellt werden, wie im Ansatz VoT im Kapitel 3 aufgezeigt. Eine sukzessive Verbesserung erlaubt es bei einem Punkt ein höheres Level zu erreichen. Föderationen können bestimmte Vektoren vorgeben, die zu erreichen sind. Ebenso können SPs Vektoren festlegen, die als Mindestanforderung für ihren Dienst gelten. Maturity Level und Vektoren können vom Prinzip ähnlich verwendet werden. Im Gegensatz zu Vektoren spiegelt der Name Maturity Level wieder, dass sich Organisationen verbessern können.

Als nächstes müssen die betrachteten Aspekte festgelegt werden. Diese sollen zum einen alle relevanten Aspekte enthalten, zum anderen orthogonal angeordnet sein. Die relevanten Aspekte werden durch Szenarien bzw. Anforderungen erfasst und entsprechend gruppiert oder voneinander abgegrenzt. Hierfür bieten die Szenarien aus Kapitel 2 eine repräsentative Auswahl an einfachen, strengen, gewichteten und unterschiedlichen Anforderungen. Zudem zeigen die unterschiedlichen Schemata von LoA Gemeinsamkeiten der betrachteten Aspekte. Um für ein möglichst universal einsetzbares Schema und auch für die Umrechnung von LoA bei der zentralen Komponente MdFIM alle relevanten Aspekte zu erreichen, wird eine erweiterte Schnittmenge verwendet. Neben den Aspekten, die in allen Schemata eingesetzt werden, werden auch Aspekte einbezogen (*Organisation*), die in mehreren Schemata verwendet werden und die sich damit von anderen Schemata abgrenzen. Daraus ergeben sich sechs

verschiedene Aspekte, die ähnlich sind zu denen aus Benjamin Meiers Bachelorarbeit [Mei15]:

Identification: Identifizierung und Registrierung, Art und Sicherheit der Identifizierung.

Data Management: Aktualität der Daten, Genauigkeit der Daten.

Authentication: Art und Sicherheit der Authentifizierung.

Assertions: Sicherheit der Assertions.

Accountability: Technische Maßnahmen.

Organizational Management: Organisatorische Anforderungen, wie Audits.

Für diese Aspekte müssen im abschließenden Schritt vier verschiedene Levels, Maturity Levels, erstellt werden. Level 1 soll jeweils die geringsten Anforderungen ausdrücken. Level 4 ist das höchst möglichste Level. Daraus ergibt sich die nachfolgend beschriebene Spezifikation.

Spezifikation eines LoA

Für die Identifikation wird *Maturity Level Identification (MLI)* verwendet. Eine einfache Registrierung erfolgt ohne Überprüfung der Identität, jedoch wird bereits eine eindeutige ID vergeben, die über mehrere Sessions dieselbe bleibt. Für MLI2 und MLI3 wird eine Verifikation erforderlich, die von einer einfachen Version, beispielsweise über E-Mail oder Social Media, bis hin zu einer eindeutigen Verifikation über Personalausweis oder ähnliche amtliche Dokumente reicht. Durch die verbesserte Verifikation, ist die Wahrscheinlichkeit höher, das die Identität dieselbe ist als vorgegeben. Ab MLI3 wird zudem eine Dokumentation benötigt, um die Registrierung auch nachvollziehen zu können. MLI4 fügt diesen Anforderungen zudem ein Vertrag zwischen IdP und Person hinzu. Damit sind beide Parteien über einen Vertrag miteinander verbunden, was wiederum eine höhere Sicherheit darstellt. Ebenso soll die Dokumentation zur Registrierung mindestens 3 Jahre bzw. der Dauer der landesüblichen oder internationalen Fristen aufbewahrt werden. Eine ähnliche Abstufung findet sich auch in verschiedenen Schemata und somit bildet dies eine Schnittmenge.

Maturity Level Identification

MLI1: Registrierung, eindeutige ID.

MLI2: Plus einfache Verifikation (Person, Social, E-Mail).

MLI3: Plus eindeutige Verifikation (Personalausweis oder gleichwertige Methode), Dokumentation.

MLI4: Plus Vertrag, Aufbewahrung 3 Jahre bzw. innerhalb der landesüblichen Fristen.

Damit SPs sicher gehen können, dass nur berechtigte Benutzer Zugriff auf ihre Dienste haben, ist nicht nur die Überprüfung der Identität, sondern auch die Aktualität der Daten entscheidend (*Maturity Level Data (MLD)*). Die Aktualität der Daten ist ein fester Bestandteil des Identity Lifecycle Managements. Dies ist auch im Sinn von IdPs, wenn der entsprechende Dienst monetär abgerechnet wird. Eine im Rahmen des Projektes GÉANT 4 Phase 1 durchgeführte Umfrage[1] bei IdPs zeigte hierbei eine Varianz von weniger als 2 Wochen, teils unter 2 Tage, bis hin zu mehr als 6 Monaten. Basierend auf den verschiedenen Fristen bei Normen und Verlässlichkeitsklassen der Föderationen ergibt sich folgende Abstufung:

Maturity Level Data

MLD1: Daten müssen vorgehalten werden.

MLD2: Plus Aktualität von 2 Monaten, Dokumentation.

MLD3: Reduzierung auf 2 Wochen.

MLD4: Reduzierung auf 2 Tage.

Ein weiterer Schritt zur Absicherung ist die Authentifizierung, die jedoch im Gegensatz zum alltäglichen Gebrauch nur ein Faktor ist. Wenn die Authentifizierung aus zwei Faktoren besteht, aber die Identität der Person nicht sicher ist, resultiert dies nicht in mehr Sicherheit. Mikael Linden [Lin09] zeigt den Zusammenhang zwischen Identität des Nutzers und Authentifizierung auf. Je stärker die Vernetzung, desto mehr Schaden kann durch einen Identitätsdiebstahl eintreten. Dies ist auch in Abbildung 5.20 dargestellt. Authentifizierung besteht grundlegend aus einem oder mehreren der folgenden Dinge:

- etwas, was die Benutzer wissen, wie Passwort.
- etwas, was die Benutzer besitzen, wie ein Token.
- etwas, was die Benutzer sind, wie Fingerabdruck.

Basierend auf den Normen und Verlässlichkeitsklassen der Föderationen ergeben sich folgende vier *Maturity Level Authentication (MLAuth)*, die eine Schnittmenge bisheriger Anforderungen bilden:

Maturity Level Authentication

MLAuth1: Selbe Person, mindestens verschlüsseltes Passwort.

[1] Daniela Pöhn, Tangui Coulouarn und Nicole Harris: Service Aspects of Assurance`https://docs.google.com/document/d/13Ru2_eRIpJoRl_9Phm6FyLg2eVgnzFCKn4kat50c-uw/edit?pref=2&pli=1` [Online, abgerufen am 06.01.2016].

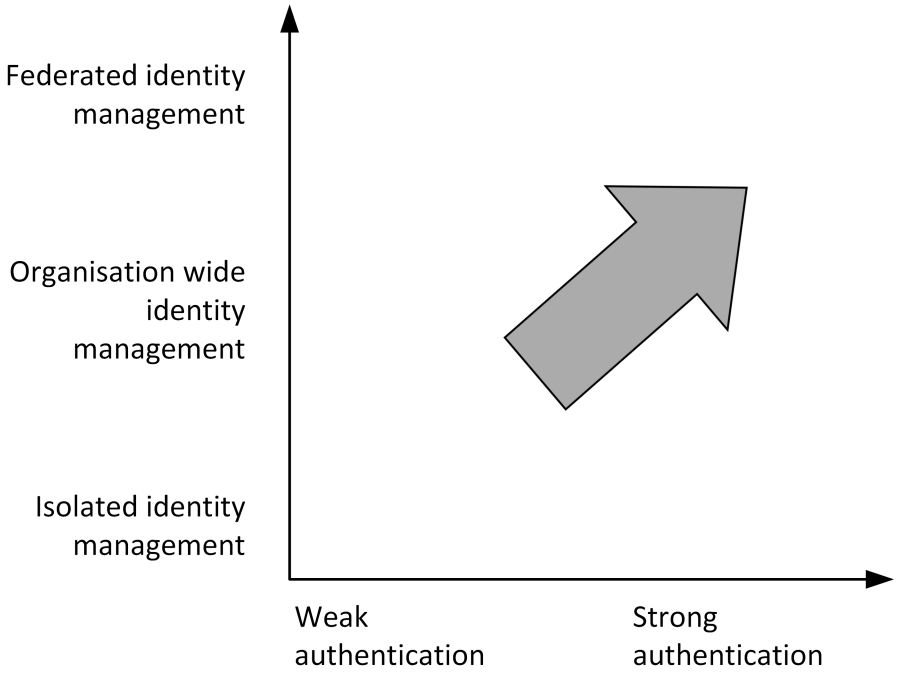

Abbildung 5.20.: Zusammenhang von Identität des Nutzers und Authentifizierung nach [Lin09]

MLAuth2: Persönliches Benutzerkonto, sicheres Passwort, mindestens Challenge-Response.

MLAuth3: Plus Token, einfacher Schutz vor Angriffen.

MLAuth4: Einschränkung auf Hard Token, guter Schutz vor Angriffen.

Assertions sind die Grundlage zur Übermittlung von Benutzerinformationen. Damit diese nicht verändert werden, gibt es verschiedene Schutzmechanismen, die unterschiedlich aufwändig sind. Hier wurde die Abstufung von VoT übernommen, nachdem sie im Gegensatz zu NIST eine Abstufung aufweist und zum anderen verschiedene Schutzmechanismen einsetzt. Zunächst wird von keinem Schutz ausgegangen, während als nächstes ein signierter und verifizierter Token eingesetzt wird. Ein Token über einen anderen Kanal ausgetauscht verbessert den Schutz, während ein verschlüsselter Token am optimalsten ist. Darauf basierend wird *Maturity Level Assertion (MLA)* spezifiziert:

Maturity Level Assertion

MLA1: Kein Schutz.

MLA2: Signierter und verifizierter Token.

MLA3: Signierter und verifizierter Token, über anderen Kanal.

MLA4: Verschlüsseltes Token.

Ein Aspekt, der je nach Norm und Verlässlichkeitsklasse eine unterschiedliche Rolle spielt, ist die technische Komponente. Wenn Hardware und Software nicht sicher sind, können beispielsweise Daten verändert oder gelöscht werden. Zudem kann es sein, dass ein IdP nicht verfügbar ist. Für einen einfachen Dienst ist das weniger wichtig, als wenn ein kommerzieller Dienst missbräuchlich benutzt oder nicht verwendet werden kann. Mehrere Schemata verwenden die Begriffe niedrige und hohe Sicherheit, wobei diese Abstufung nicht definiert ist. Um dieses Fehlen zu vermeiden, werden konkrete Maßnahmen bzw. Prozesse definiert. Bei der Abstufung von *Maturity Level Technical (MLT)* ist zudem berücksichtigt, dass es problematisch sein kann einen externen Auditor zu finden, der sich mit der Materie auskennt. Die University of Chicago musste über ein Jahr warten, bis sie einen geeigneten Auditor fand[2].

Maturity Level Technical

MLT1: Hardware muss gewartet werden, Logs.

MLT2: Plus Monitoring und weitere technische Maßnahmen (wie Firewall und IDS).

MLT3: Plus Computer Security Incident Response Team (CSIRT) sowie entsprechende Prozesse bei Security Incidents, Verfügbarkeit und (interne) Audits.

MLT4: Einschränkung auf externe Audits.

Zusätzlich wird die Organisation in Form von Datenschutz und Nutzungsrichtlinien betrachtet. Dieser Aspekt hat im Vergleich zu den vorher genannten Maturity Level eine geringere Wichtigkeit, kann jedoch bei internationalen Zusammenarbeiten trotzdem wichtig sein. Ferner wird dieser Aspekt im nachfolgend beschriebenen LoT ebenfalls verwendet. Ein impliziter Datenschutz und eine Nutzungsrichtlinie werden in *Maturity Level Organizational (MLO)* 1 eingeordnet. Eine explizite Akzeptanz des Datenschutz des jeweiligen Landes wird bereits höher bewertet, während eine explizite Datenschutzerklärung für den Dienst spezialisiert in MLO3 eingeordnet wird. Ein expliziter hoher Datenschutz ergibt das höchste Maturity Level.

Maturity Level Organizational

MLO1: Impliziter Datenschutz und Nutzungsrichtlinie.

MLO2: Expliziter Datenschutz des Landes.

MLO3: Expliziter Datenschutz des Dienstes.

[2]Laut Gespräch mit Tom Barton, University of Chicago, InCommon Technical Advisory Committee und Senior Director – Architecture, Integration, & CISO.

MLO4: Expliziter Datenschutz des Dienstes, der höher als der Datenschutz des Landes ist.

Die Tabelle 5.7 fasst die Maturity Level zusammen. Die eben aufgeführten Maturity Level können, wie nachfolgend beschrieben, in verschiedenen Profilen innerhalb von Föderationen und Nationen auf die jeweiligen Begebenheiten angepasst bzw. genau festgelegt werden.

Technische Spezifikation eines LoA

In den oben aufgeführten Level of Assurance wird ein Level als ein Paar Level:Wert angegeben, der fest definiert ist. Die verfügbaren Werte und ihre Semantik sind üblicherweise in einem für Menschen lesbaren Dokument verfasst. Um eine Automatisierung auf Basis von Maturity Level zu erlauben, müssen die Werte vergleichbar sein und die Anforderungen der niedrigeren Level miterfüllen.

Es gibt zwei Arten, um den LoA eines IdPs anzugeben. LoAs können in den Metadaten der Organisationen beschrieben sein. Das im Kapitel 3 beschriebene SAML Identity Assurance Profile [SAMLIAP] [KHM+10] erlaubt die Verwendung von mehreren URI-Referenzen, um erfüllte LoAs auszudrücken. Eine URI im Format `http://foo.example.com/assurance/loa1` beschreibt somit genau einen Level. Im Gegensatz dazu verwendet Vectors of Trust einen URN-String, wodurch alle benötigten LoA-Informationen hierdurch ausgedrückt sind. Das eliminiert die zusätzliche Verwendung eines externen Dokuments, welches nicht automatisch verarbeitet werden kann. Eine URN bei VoT beinhaltet die Zuordnung von Vektor und Wert: `urn:ietf: param:[TBD]:P1.Cc.A3`. Zusätzlich können Nutzer-spezifische LoA als Attribute, z. B. dem `eduPersonAssurance` Attribut, an den SP gesendet werden. Dies ist theoretisch auch für VoT möglich, auch wenn bisher (Stand Januar 2016) keine Implementierung dessen besteht.

Um die Parallelität von bisherigen LoA und den Maturity Levels zu erlauben, müssen mit dem Format eigene Levels sowie mehrere Aspekte übertragen werden können. Hierfür wird eine URI gesetzt, die als Parameter `loa` oder `ml` für Maturity Level enthält. Mindestens einer der Parameter muss mit einem Wert gefüllt sein. Wenn beide Parameter gesetzt werden sollen, können zwei URIs eingesetzt werden.

```
https://loa.mdfim.net/mdfim?loa=
        http%3A%2F%2Ffoo.example.com%2Fassurance%2Fml=I1.D2.Auth3.A4.T3.O2

https://loa.mdfim.net/mdfim?loa=
        http%3A%2F%2Ffoo.example.com%2Fassurance%2Floa=loa1
```

Der Name des LoA kann zudem in Klammern `%5Bprofile%5D` hinter dem Wert stehen. Das URI-Schema und der `hier`-Part bezeichnen die Implementierung dieses Ansatzes, um den Unterschied zu traditionellen SAML-Implementierungen anzuzeigen:

`https://loa.mdfim.net/mdfim`.

Art des MLs	ML1	ML2	ML3	ML4
MLI	eindeutige ID	einfache Verifikation	eindeutige Verifikation, Dokumentation	Vertrag, Aufbewahrung
MLD	Speicherung	Aktualität 2 Monate	Aktualität 2 Wochen	Aktualität 2 Tage
MLAuth	selbe Person, Passwort	persönliches Konto, sicheres Passwort	Token, Schutz	Hard Token, guter Schutz
MLA	kein Schutz	signiert	anderer Kanal	verschlüsselt
MLT	Wartung, Logs	Monitoring, Sicherheit	Verfügbarkeit, Audits	externe Audits
MLO	implizit	explizit Land	explizit Dienst	explizit, höherer Datenschutz

Tabelle 5.7.: Zusammenfassung des Level of Assurance als Maturity Level

Um mehrere LoAs anzuzeigen, müssen unterschiedliche URIs verwendet werden. Mehrere URIs mit unterschiedlichen Parametern für denselben `LoA` zeigen an, dass die Nutzer des IdPs unterschiedliche Levels besitzen. Der erst genannte Level ist dabei der gebräuchlichste. Für einen genauen Vergleich muss jedoch der Nutzer-LoA herangezogen werden, der als Attribut übertragen werden muss.

Solange keine unabhängige Instanz diese Informationen verifiziert, muss der SP den Informationen des IdPs glauben. Bis jetzt wird laut Wissen des Autors keine Methode verwendet, um den gewünschten LoA auf Seiten des SPs auszudrücken. Dieser Wert hängt von der Risikobewertung des eigenen Dienstes ab. Für eine Automatisierung muss diese Information ebenso transportiert werden, um automatisch einen Abgleich durchführen zu können. Dies geschieht im selben Element der Metadaten. Die Mindestanforderung des LoAs gilt nur dann erfüllt, wenn alle Werte mindestens gleichwertig, wenn nicht höher sind. Erreicht ein Aspekt nicht den benötigten Wert, gilt der Trust als nicht genügend.

Um einen automatischen Abgleich durchführen zu können, müssen Workflows und Werkzeuge definiert werden. Dies erfolgt in Abschnitt 5.4.3, nachdem ein LoT spezifiziert wurde, der diese Workflows und Werkzeuge mitbenutzen soll.

Anwendung

Um das Trust Management, Level of Assurance und Level of Trust zu verdeutlichen, soll von einer Organisation ausgegangen werden, die sowohl einer Föderation als auch einer Community zugehört. Zudem hat die Organisation Kooperationen innerhalb von Forschungsprojekten. Die Organisation ist eine Universität, die sowohl IdP als auch einen SP mit mehreren Diensten betreibt. Zunächst bestimmt die Universität den einen Maturity Level für den IdP:

- MLI: Einfache Verifikation bei Studenten vor der ersten Prüfung, eindeutige Verifikation für alle Studenten nach der ersten Prüfung, Dokumentation und Vertrag sowie Aufbewahrung für alle Mitarbeiter.
- MLD: Aktualität 2 Wochen für Studenten und Mitarbeiter.
- MLAuth: Persönliches Benutzerkonto und sicheres Passwort für Studenten und Mitarbeiter, in Spezialfällen auch Token.
- MLA: Signatur.
- MLT: Monitoring und eine gewisse Sicherheit.
- MLO: Expliziter Datenschutz für den Dienst, den jeder Nutzer Anfangs akzeptieren muss.

Daraus ergibt sich ein Mindest-Level (I2.D3.Auth2.A3.T2.O3), welches als eine Art Array geschrieben wird. Als URI beschrieben sieht das wie folgt aus:

```
https://loa.mdfim.net/mdfim?loa=http%3A%2F%2Ffoo.example.com
        %2Fassurance%2Fml=I2.D3.Auth2.A3.T2.O3
```

Für Mitarbeiter kann dieses Level auf (I4.D3.Auth3.A3.T2.O3) steigen. Dies ergibt folgende URI:

```
https://loa.mdfim.net/mdfim?loa=http%3A%2F%2Ffoo.example.com
        %2Fassurance%2Fml=I4.D3.Auth3.A3.T2.O3
```

Nachdem die allermeisten Benutzer den LoA (I3.D3.Auth2.A3.T2.O3) besitzen, wird dieser in den Metadaten als Erstes angegeben, gefolgt von den weiteren Maturity Levels, und anschließend von der zuständigen Föderation akzeptiert. Zudem muss, auf Grund der Unterschiede, der Benutzer-bezogene LoA im I&AM bzw. im darunter liegenden LDAP gespeichert werden. Für Mitarbeiter wird grundsätzlich nach der Einstellung (I4.D3.Auth2.A3.T2.O3) vergeben. Wenn der Mitarbeiter ein Token anfordert, wird das Level auf (I4.D3.Auth3.A3.-T2.O3) automatisch erhöht. Bei Studenten wird nach der ersten Prüfung das Level ebenso erhöht. Wenn die Überprüfung des Personalausweises nicht erfolgreich war, läuft ein neuer Prozess an, um die Identität des Studenten gesondert zu überprüfen.

Die Föderation hat zwei Profile angelegt. Das niedrige Profil hat die Mindestanforderung (I2.D2.Auth2.A1.T1.O1). Um die Übereinstimmung zu dokumentieren, soll in diesem Fall sowohl `loa` als auch `ml` verwendet werden:

```
https://loa.mdfim.net/...%5C%2Fassurance%5C%2Floa
        %5C%3Dloa1%5federation%5D%2Fml=I2.D2.Auth2.A1.T1.O1
```

Diese Anforderung erfüllt die Universität ohne Probleme. Das höhere Profil hat die Mindestanforderung (I3.D3.Auth2.A3.T3.O3).

```
https://loa.mdfim.net/...%5C%2Fassurance%5C%2Floa
        %5C%3Dloa2%5federation%5D%2Fml=I3.D3.Auth2.A3.T3.O3
```

Zwar kann die Universität eine hohe Verfügbarkeit aufweisen, jedoch werden keine Audits durchgeführt. Durch das Self-Assessment mithilfe eines zentralen Tools können die Administratoren diese Verbesserungsmöglichkeit sehen. Zudem befinden sich hier Informationen, wie ein Audit aussehen kann und was beachtet werden soll. Nachdem der SP einer Community, den eine kleine Gruppe von Forschern verwenden möchte, genau diesen Maturity Level MLT auf 3 gesetzt hat, versuchen die Zuständigen Audits zu realisieren. Alternativ kommen Verträge in Frage.

Der SP einer Kooperation benutzt den LoA des jeweiligen Industriesektors. Dieser ist, genauso wie die Maturity Level und die Profile der Föderation in der IANA Registry eingetragen.

```
https://loa.mdfim.net/...%5C%2Fassurance%5C%2Floa
        %5C%3Dloa%5industry%5D%2Fml=I2.D2.Auth2.A3.T2.O3
```

Mithilfe des Vergleichstools des MdFIM können beide Level of Assurance verglichen werden. Ein Beispiel für einen solchen Vergleich ist im Anhang A.3 gegeben. Es zeigt sich, dass der IdP die Anforderungen erfüllt.

5.4.2. Level of Trust

Definition 8. Level of Trust

Level of Trust ist die schriftliche Repräsentation in das Vertrauen, welches ein Identity Provider in einen Service Provider haben kann.

Im Gegensatz zu Level of Assurance gibt es bisher keine äquivalente Klassifikation von SPs. Da Identity Provider häufig keine oder nicht alle benötigten Attribute an Service Provider senden, soll eine Klassifikation helfen Service Provider einzuteilen. Dieses Level of Trust soll ähnlich wie der Level of Assurance aufgebaut sein. Nachfolgend werden zunächst die Anforderungen wiederholt werden, bevor der LoT spezifiziert wird. Der definierte LoT soll im letzten Schritt angewandt werden.

Anforderungen

Die Anforderung [FA-LoT] mit Priorität 2 soll IdPs die Möglichkeit geben die Vertrauenswürdigkeit von SPs zu überprüfen und geeignet darzustellen. Hierfür gibt es, ähnlich wie bei Level of Assurance, Unteranforderungen, die bei der Spezifikation zu beachten sind.

- [LoT-Auditing],
- [LoT-Automatisierung],
- [LoT-Datenschutz],
- [LoT-Einordnung],
- [LoT-Gewichtung],
- [LoT-Implementierungsunabhängigkeit],
- [LoT-Konfiguration],
- [LoT-Protokollunabhängigkeit],

- [LoT-Realisierbarkeit],
- [LoT-Registrierung]und
- [LoT-SelfAsserted].

Nachdem keine bisherigen Level of Trust vorhanden sind, werden die Anforderungen [Mapping] und [Koexistenz] nicht dringend benötigt. Trotzdem sollen sie beachtet werden, um zukünftige LoT zu unterstützen. Die neu hinzugekommene Anforderung [LoT-Datenschutz] umfasst den Datenschutz. Dazu gehören die Datenschutzrichtlinien des Landes, Nutzungsrichtlinien und Auswirkungen auf die Privatsphäre durch den Dienst. Dies ist auch im Einklang mit dem NIST Internal/Interagency Reports (NISTIR) 8062 Draft Privacy Risk Management for Federal Information System der NIST [BN15], der Planen, Monitoring, Risiko ermitteln und Handeln in Rahmen von Plan-Do-Check-Act empfiehlt.

Spezifikation

Das Vorgehen zur Spezifikation des LoT ist ähnlich wie das für LoA. Es wird dabei versucht Maturity Levels wiederzuverwenden, um eine möglichst hohe Konsistenz zu erreichen. Bei Betrachtung der oben genannten Aspekte zeigt sich, dass *Identification*, *Authentication* und *Assertions* für SPs irrelevant sind. Hingegen werden die folgenden Aspekte weiterhin benötigt:

Data Management: Speicherung der Daten.

Accountability: Technische Maßnahmen.

Organizational Management: Organisatorische Anforderungen, wie Audits.

Dabei ist insbesondere die Privatsphäre der Nutzer stärker zu betrachten. Dies betrifft die Datenschutzgesetze der entsprechenden Länder, die Anzahl der verwendeten Benutzerinformationen und die Nutzungsrichtlinien der Dienste. Die Richtlinien und Gesetze sowie die benötigten Benutzerinformationen können dabei schwierig quantifiziert werden, wie bei der geänderten Klassifikation von *MLO* zu sehen:

Maturity Level Organizational

MLO1: Implizite Nutzungsrichtlinie, implizit Datenschutz.

MLO2: Plus explizite Nutzungsrichtlinie, Datenschutz des Landes, nur wenige, benötigte Benutzerinformationen.

MLO3: Plus expliziter Datenschutz des Dienstes, hoher Datenschutz durch Land.

MLO4: Einschränkung auf hoher Datenschutz des Dienstes, der höher ist als der des Landes.

Die Verfügbarkeit des Dienstes sowie die weitere Sicherheit sind insbesondere bei kommerziellen Diensten interessant, wodurch sich *MLT* ergibt:

Maturity Level Technical

MLT1: Hardware muss gewartet werden, Logs.

MLT2: Plus Monitoring und technische Maßnahmen (wie Firewall und IDS), auch für Logfiles.

MLT3: Plus CSIRT, Verfügbarkeit und Audits.

MLT4: Einschränkung auf externe Audits.

Nachdem bei SPs ebenfalls Benutzerdaten speichern, jedoch nicht alle Benutzerinformationen haben, sind die Ausprägungen entsprechend angepasst. Service Provider fordern von IdPs Benutzerdaten an, die möglichst vom Nutzer zudem freigegeben werden. Zudem kann es bei Diensten vorkommen, dass sie Daten weiter geben. Dies soll ebenfalls erst nach expliziter Zustimmung erfolgen, wie beim Datenschutz in Kapitel 2 zu sehen. Somit ergibt sich folgendes *MLD*:

Maturity Level Data

MLD1: Daten müssen sicher vorgehalten werden, vor Weitergabe Einwilligung.

MLD2: Plus explizite Einwilligung, Erklärung für was Daten benötigt, Dokumentation.

MLD3: Plus einfache organisatorische Maßnahmen, um Daten zu schützen.

MLD4: Plus organisatorische Maßnahmen nach Normen.

Die Tabelle 5.8 fasst die Maturity Level für Level of Trust zusammen.

Ebenso wie bei LoA werden bei Level of Trust verschiedene Maturity Level eingesetzt, die möglichst identisch sind. Dazu gehören Maturity Level Organizational, Maturity Level Technical und Maturity Level Data. Maturity Level Authentication, Maturity Level Identification und Maturity Level Assertion lassen sich nicht auf SPs anwenden und sind daher im LoT weggelassen. MLO umfasst, im Gegensatz zu LoA stärker den Datenschutz, benötigte Benutzerinformationen und Richtlinien. MLT ist in beiden Klassifikationen identisch. MLD unterscheidet sich am stärksten. Während MLD bei LoA auf die Aktualität der Daten eingeht, wird bei LoT die Weitergabe von Daten, die Abfrage von Daten sowie organisatorische Komponente betrachtet.

Art des MLs	ML1	ML2	ML3	ML4
MLO	implizit	explizit Land	explizit Dienst	explizit, höherer Datenschutz
MLT	Wartung, Logs	Monitoring, Sicherheit	Verfügbarkeit, Audits	externe Audits
MLD	Speicherung	explizit	organisatorisch	Normen

Tabelle 5.8.: Zusammenfassung des Level of Trust als Maturity Level

Anwendung

Die oben beschriebene Universität betreibt auch mehrere Dienste. Darunter befindet sich ein Gitlab-Dienst, der sowohl von internen Mitarbeitern, Studenten als auch von Externen innerhalb von gemeinsamen Projekten genutzt wird. Nachdem der Dienst nur nach Einladung von Internen benutzt werden kann und es kaum Risiken für die Betreiber gibt, wird (I1.D2.Auth1.A1.T1.O1) als Mindestanforderung angesetzt.

```
https://loa.mdfim.net/mdfim?loa=http%3A%2F%2Ffoo.example.com
        %2Fassurance%2Fml=I1.D2.Auth1.A1.T1.O1
```

Der Dienst erfüllt selbst den LoT von (O3.T2.D2). Dieser LoT wird in den Metadaten des SPs angegeben.

```
https://lot.mdfim.net/mdfim?lot=http%3A%2F%2Ffoo.example.com
        %2Fassurance%2Fml=O3.T2.D2
```

Innerhalb der Föderation ist das niedrigere Profil mit den Mindestanforderung (I2.D2.-Auth2.A1.T1.O1) ausreichend.

```
https://loa.mdfim.net/...%5C%2Fassurance%5C%2Floa
        %5C%3Dloa1%5federation%5D%2Fml=I2.D2.Auth2.A1.T1.O1
```

Somit können alle Teilnehmer der Föderation den Dienst nutzen, wenn sie dazu eingeladen wurden.

5.4.3. Technische Realisierung des Werkzeugs

Damit die Maturity Level von LoA und LoT eingesetzt und zum Vergleich vorhandener LoA Schemata verwendet werden können, müssen MdFIM und die dazugehörigen Erweiterungen entsprechende Funktionalitäten bereitstellen. Diese sowie die technischen Werkzeuge hierfür werden nachfolgend, aufgeteilt in MdFIM, Entitäten und Föderationen, beschrieben.

MdFIM-seitige Realisierung

Latifa Boursas [Bou09] hat in ihrer Dissertation einen TrustBroker beschrieben. Dieser zentrale Dienst sammelt alle relevante Daten, errechnet Trust Level und aggregiert bzw. vergleicht den Trust Level mit inter-organisationalen Trust Level Schemata. Die Entscheidung über Zugriff bzw. Vertrauen trifft die Access Decision Engine. Eine ähnliche Funktionalität übernimmt das MdFIM-seitige Werkzeug.

Die Vertrauensinformationen in Form von LoA oder LoT werden im Datenmodell verlinkt

über eine Umsetzungstabelle zur Tabelle `Entity` gespeichert. Diese Vertrauenswerte spiegeln alle verwendeten Vertrauensgrad innerhalb der Entität wieder. Wenn Benutzer beispielsweise unterschiedlich geprüft werden, muss dies in den Assertions sowie in den Metadaten vermerkt sein. In den Metadaten sind neben dem Standard Level auch alle weiteren möglichen Levels anzugeben, während beim Benutzer das jeweils verwendete Level steht. Ist es nicht möglich, den jeweiligen Vertrauensgrad im lokalen I&AM zu speichern, muss hierfür eine Attribute Authority verwendet werden. Der Vertrauensgrad im Datenmodell kann durch den jeweilig Zuständigen der Entität geändert werden. Dies ist durch die API über ein Skript oder eine Webanwendung möglich.

In MdFIM gibt es die Möglichkeit die Vertrauensgrade auf Basis der Maturity Level im `TrustHandler` zu vergleichen. Dies ist insbesondere dann wichtig, wenn einzelne Aspekte nicht betrachtet oder unterschiedliche Schemata verwendet sowie wenn bisherige LoA Schemata eingesetzt werden. Zunächst wird der Vergleich der Trust-Werte durch die Funktion `compareTrust()` beschrieben. Dieser wird nachfolgend am Beispiel von LoA durchgeführt, er lässt sich jedoch genauso auf LoT übertragen. Nachdem sowohl IdP als auch SP mehrere LoA URIs besitzen können, muss jede URI miteinander verglichen werden. Die Mindestanforderungen des SPs sind erfüllt, wenn wenigstens ein LoA-Pärchen passt. Zudem muss in diesem Fall der individuelle LoA des Nutzers auf SP-Software-Seite überprüft werden. Verwenden IdP und SP unterschiedliche LoA-Schemas, muss die Vergleichstabelle von MdFIM zunächst beide LoAs in die Maturity Level umrechnen, um sie dann vergleichbar zu machen. Der grundsätzliche Ablauf des Vergleichs ist in Listing 5.36 in Pseudocode dargestellt.

```
1  typedef enum {NOT_FULFILLED, FULFILLED} comparison_result;
2
3  comparison_result compare_LOA_URIs(sp_loa_uris, idp_loa_uris) {
4    foreach sp_loa_uri in sp_loa_uris do {
5        foreach idp_loa_uri in idp_loa_uris do {
6          foreach sp_loa_aspect in all_sp_loa_aspects do {
7              if (idp_loa_aspect of same type exists
8                  or can be derived via mapping table) {
9                if (idp_loa_aspect.value < sp_loa_aspect.value) {
10                 // IDP guarantee does not fulfill SP requirement
11                 continue with next idp_loa_uri;
12               }
13             }
14             else { // LoA aspect requested by SP is not known to IDP
15               continue with next idp_loa_uri;
16             }
17         }
18         return FULFILLED; // All relevant LoA aspects had suitable values
19       }
20   }
21   return NOT_FULFILLED; // No suitable (SP LoA, IDP LoA)-pair was found
22 }
```

Listing 5.36: Vergleich der LoA-Werte von IdP und SP in Pseudocode

Dieser grundsätzliche Workflow kann nicht nur für LoA-Angaben in Metadaten, sondern

auch bei Benutzerattributen eingesetzt werden. Die relationale Datenbank von MdFIM benötigt zusätzliche Informationen, um einen Vergleich der LoAs durchführen zu können. Metadaten enthalten, wie bereits erklärt, Informationen zum LoA. Der Vergleich zweier LoA-Werte soll sowohl für Maturity Level als auch für bisherige LoA möglich sein. Diese Information muss gelesen, geparst und in der Datenbank gespeichert werden.

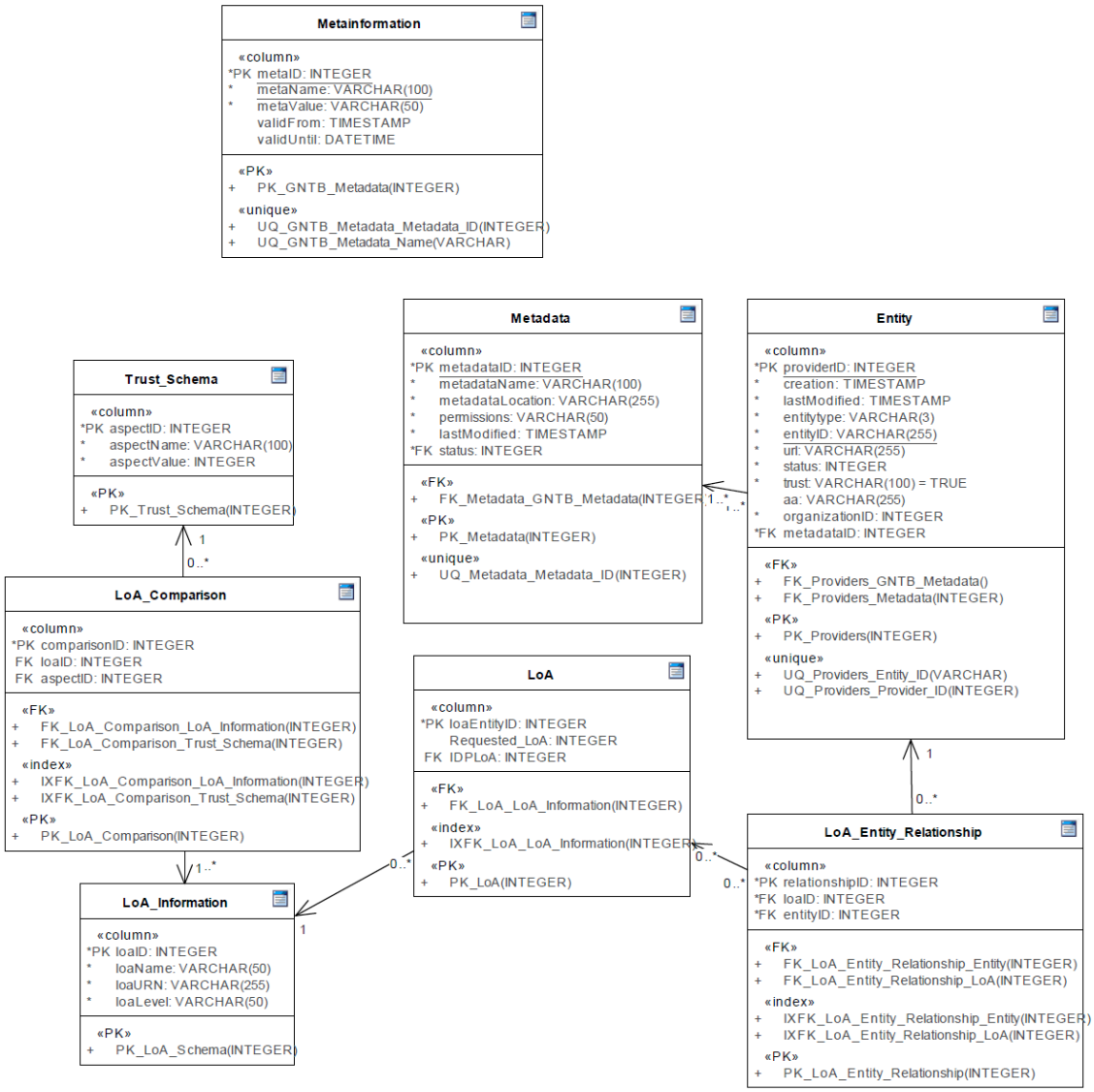

Abbildung 5.21.: Datenbank für das Trust Management

Um alle Trust-Informationen zu speichern und einen Vergleich möglich zu machen, gibt es die folgenden Tabellen, die in Abbildung 5.21 dargestellt sind:

- `LoA` enthält den oder die LoA-Werte der IdPs sowie die gewünschten LoAs der SPs.

- `LoA_Entity` ist die Verknüpfungstabelle zwischen den Tabellen `LoA` und `Entity`.
- `LoA_Information` enthält Informationen zu den unterschiedlichen LoAs, ihre URI bzw. URN sowie weitere Informationen.
- `LoA_Comparison` ist die Vergleichstabelle, um einen LoA in die unterschiedlichen Aspekte umzurechnen.
- `Trust_Schema` enthält die unterschiedlichen Aspekte.

Die Trust-Information, d. h. der LoA der Metadaten, wird in der Variable `IDP_LoA` der zusätzliche Tabelle `LoA` gespeichert. Der Eintrag mit der Variable `IDP_LoA` ist mit der Tabelle `LoA_Entity` verknüpft, die wiederum mit der `Entity`-Tabelle verbunden ist. Durch diese Tabellen und die Umsetzungstablle, können IdPs mehrere LoAs verwenden. Ebenso können die Mindestanforderungen der SPs in der Tabelle `LoA` gespeichert werden. Hierfür existiert die Variable `RequiredLoA`.

Eine weitere Tabelle, `LoA_Comparison`, beschreibt die Konvertierung eines LoAs in die unterschiedlichen Aspekte. Basierend auf der Fragmentierung in diese eindeutigen Aspekte, kann das Tool einen Vergleich zwischen unterschiedlichen LoA Schemata durchführen. Hierzu ist es wichtig, dass die Tabelle erstmalig befüllt wird, wenn keine weiteren automatischen Methoden, wie Ontologien verwendet werden. Um zwei LoA Schemata zu vergleichen, liest die Funktion `compareTrustSchema` beide LoA-Werte, sucht nach dem passenden LoA Schema und vergleicht anschließend, wenn nötig, jeden Aspekt mit Hilfe der Tabelle. Dies ist als Pseudocode in Listing 5.37 dargestellt. Wenn das Ergebnis des Vergleichs `false` bzw. `NOT_FULLFILLED` ist, werden die Metadaten nicht ausgetauscht.

```
typedef enum {NOT_FULFILLED, FULFILLED} comparison_result;

comparison_result compare_LOA_URIs(sp_loa_uris, idp_loa_uris) {
  foreach sp_loa_uri in sp_loa_uris do {
    sp_loa_schema := parse_LoA_URI(sp_loa_uri);
    foreach idp_loa_uri in idp_loa_uris do {
      idp_loa_schema := parse_LoA_URI(idp_loa_uri);
      foreach sp_loa_uri in sp_loa_uris do {
        if (sp_loa_schema /= idp_loa_schema) {
          foreach sp_loa_uri in sp_loa_uris do {
            sp_loa_aspect := search_ml(sp_loa_uri);
            foreach idp_loa_uri in idp_loa_uris do {
              idp_loa_aspect := search_ml(idp_loa_uri);
                if (idp_loa_aspect.value < sp_loa_aspect.value) {
                // IDP guarantee does not fulfill SP requirement
                  continue with next idp_loa_uri;
                }
              }
            }
          }
          else { // LoA schema is the same
            if (idp_loa_uri.value < sp_loa_uri.value) {
            // IDP guarantee does not fulfill SP requirement
              continue with next idp_loa_uri;
            }
          }
        }
      return FULFILLED; // All relevant LoA aspects had suitable values
    }
  }
  return NOT_FULFILLED; // No suitable (SP LoA, IDP LoA)-pair was found
}
```

Listing 5.37: Vergleich der LoA-Werte von IdP und SP basierend auf der Vergleichstabelle in Pseudocode

Latifa Boursas hat in ihrer Arbeit [Bou09] zudem die Darstellung von Vertrauensbeziehungen in Vertrauensgraphen (Trust Graphs) beschrieben. Vertrauensgraphen, angereichert mit den geographischen Positionen, beispielsweise als Bestandteil der Metadaten, können helfen die Vertrauensbeziehungen zu visualisieren. Eine Unterscheidung in Aktualität, Föderationen, Maturity Level, Schemata und verwendete Attribute ist hierbei möglich. Zudem können Informationen über nicht erfolgreiche Vertrauensaufbaue eine weitere Sicht auf die Organisationen erlauben. Hierbei können z. B. Maturity Level-Inkompatibilitäten, technische Fehler und manuelle Ablehnung als Kennzahlen verwendet werden. Weitere, interessante Kennzahlen sind u. a. Anzahl der Teilnehmer, Anzahl der Vertrauensbeziehungen, Dauer für einen Metadatenaustausch und Anzahl der Föderationen. Die Darstellung von Vertrauensbeziehungen in Vertrauensgraphen kann ferner als Grundlage für inter-organisationelle Security Incident Response Prozesse verwendet werden.

Als Beispiel für eine mögliche Visualisierung wird in Abbildung 5.22 ein Globus mit möglichen Verbindungen zwischen IdPs und SPs gezeigt.

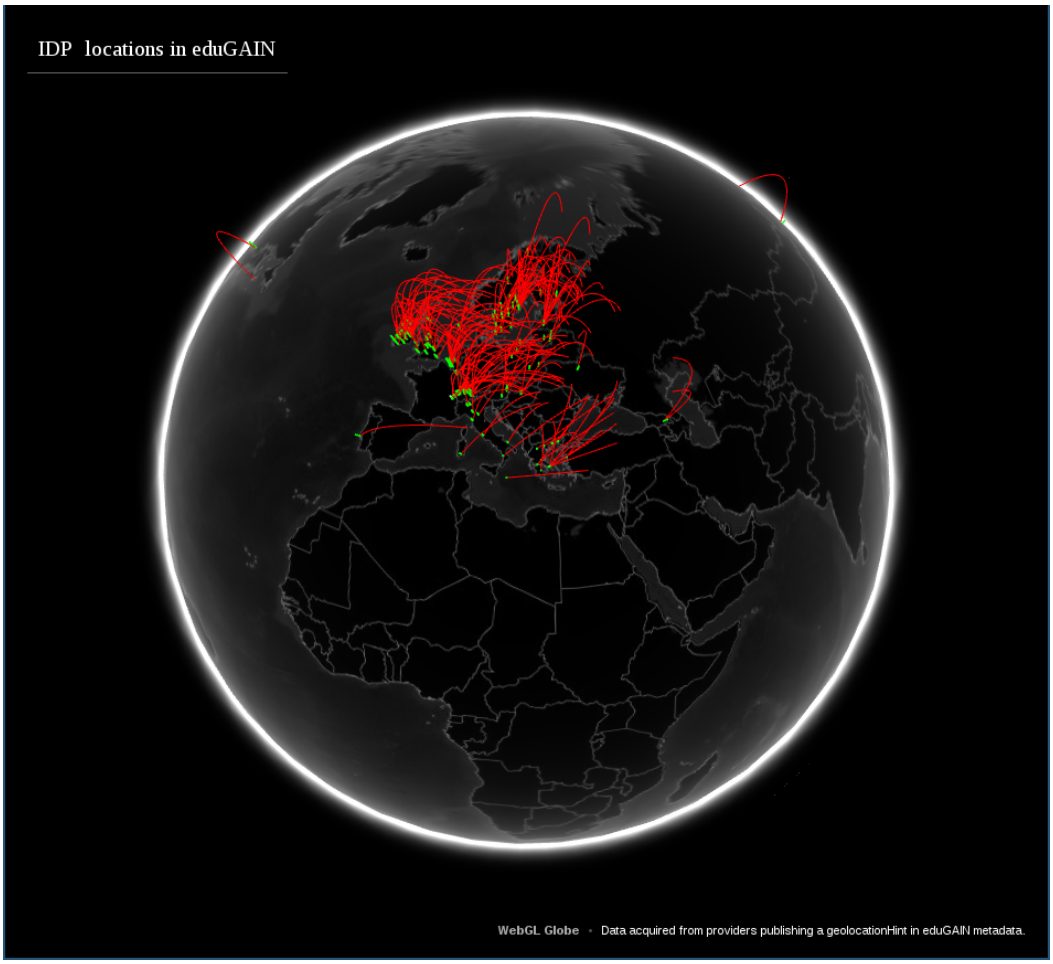

Abbildung 5.22.: Prototypische Visualisierung von IdP-SP-Verbindungen

Entitäten-seitige Realisierung

Damit Entitäten Maturity Level sinnvoll einsetzen können, müssen bestimmte Voraussetzungen erfüllt werden. Neben der Angabe des LoA bzw. LoT in den Metadaten, muss die Konfiguration angepasst werden. Dies wird nachfolgend beschrieben. Hierbei wird vom Protokoll SAML ausgegangen, jedoch kann die Umsetzung auch mit OpenID Connect realisiert werden.

Um eine implementierungsunabhängige und benutzerfreundliche Lösung zu erhalten, besitzen die Erweiterungen der SP und IdP Software eine zusätzliche Konfiguration zum Vertrauen. Diese MdFIM-Erweiterung erlaubt die Konfiguration eines Mindestgrades und weiteren Bedingungen der gegenüberliegenden Partei. Zudem wird der eigene Vertrauensgrad festgelegt. Das Schema der Konfiguration ist in Listing 5.38 abgebildet.

```
1  LoA
2          requestedLoA:
3                  EntityID1:
4                  EntityID2:
5          ownLoA:
6          profile:
7          federation:
8          checkedByFederation:
9          category:
10         authentication:
11 LoT
12         requestedLoT:
13         ownLoT:
14         profile:
15         federation:
16         checkedByFederation:
17         category:
```

Listing 5.38: Konfiguration für Trust-Aufbau und -Abgleich

Eine Funktion, `configTrust()`, passt die oben genannten Konfigurationen entsprechend an und wird für die Erweiterung um Vertrauen verwendet, um neben dem eigentlichen LoA auch andere Kategorien zuzulassen. Beispielsweise soll hiermit ermöglicht werden, nur überprüfte bzw. signierte LoA zu erlauben.

Nachdem manche Organisationen mehrere LoAs besitzen, beispielsweise wenn bestimmte Nutzer eine Zwei-Faktor-Authentifizierung verwenden, muss der LoA-Wert auch lokal überprüft werden können. Wenn kein einheitlicher LoA vorhanden ist, muss der LoA des Nutzers als Attribut `eduPersonAssurance` oder einem vergleichbaren Attribut gesendet werden. Hierfür muss in der SAML Implementierung Shibboleth der Attribute Resolver und der Attribute Filter angepasst werden, wie nachfolgend beschrieben. Um den LoA einzuschränken, muss die SP Software entsprechend konfiguriert werden. SPs schränken mögliche IdPs durch die Filterung des Tags in den Metadaten oder durch das Attribut bei dem Nutzer ein. In der Erweiterung der SP-Software ist der `TrustHandlerSP` für den Trust zuständig.

`TrustHandlerIdP` ist das Gegenstück auf IdP-Seite. Die Funktion `compareTrust()` prüft die Metadaten des SPs bzw. IdPs gegenüber dieser allgemeinen Trust-Konfiguration. Die Funktion ruft anschließend über `compareTrustMdfim()` die Vergleichstabelle der MdFIM auf, wenn nicht dasselbe Schema verwendet wird. Der Aufruf basiert auf den API Call `ttp_compareTrust()`. Entsprechend des Ergebnisses wird entschieden, ob der Nutzer den Dienst nutzen darf oder nicht. Die Überprüfung erfolgt äquivalent zum bereits aufgezeigten Pseudocode. Neben dieser Erweiterung um den Vergleich des Vertrauens, muss die IdP und SP Software zudem angepasst werden.

Identity Provider Software

Um den IdP entsprechend anzupassen, muss zum einen der LoA in den Metadaten stehen. Der SAML `AuthenticationContext` beschreibt ebenso den LoA, wie das XML-Element in den SAML Metadaten, zu sehen in Listing 5.39. In OpenID Connect wird das LoA-Profil über `acr` beschrieben, wie in Listing 5.40 zu sehen.

```
<md:EntityDescriptor xmlns="urn:oasis:names:tc:SAML:2.0:metadata"
 xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
 xmlns:attr="urn:oasis:names:tc:SAML:metadata:attribute"
 xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
 entityID="https://www.lrz.de/SAML">
  <md:Extensions>
    <attr:EntityAttributes>
      <saml:Attribute
            NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
            Name="urn:oasis:names:tc:SAML:attribute:assurance-certification">
        <saml:AttributeValue>
                    https://loa.mdfim.net/mdfim?loa=
                    http%3A%2F%2Ffoo.example.com%2Fassurance
                    %2Fml=I1.D2.Auth1.A1.T1.O1
                </saml:AttributeValue>
          </saml:Attribute>
    </attr:EntityAttributes>
  </md:Extensions>
   ...
</md:EntityDescriptor>
```

Listing 5.39: XML-Element in SAML Metadaten für LoA-Profil

```
{
  "iss": ...,
  "sub": ...,
  "aud": ...,
  "exp": ...,
  "iat": ...,
  "auth_time": ...
  "acr": "https://loa.mdfim.net/mdfim?loa=
    http%3A%2F%2Ffoo.example.com%2Fassurance%2Fml=1.2.1.1.1.1"
}
```

Listing 5.40: JSON Web Token mit dem LoA-Profile

Neben den Metadaten muss gegebenenfalls auch die lokale Konfiguration angepasst werden. Wenn kein allgemein gültiger LoA in einer Organisation existiert, muss der jeweilige LoA des Benutzers zwingend als Attribut `eduPersonAssurance` versendet werden. Dies ist entsprechend zu konfigurieren, so dass das Attribut bei Nachfrage gesendet wird. OpenID Connect bietet kein festes Vokabular für Attribute. Für die Interoperabilität sollte dieses jedoch festgelegt werden. Ein wichtiger Bestandteil hierfür ist die Assurance des Nutzers.

Um den LoT der SPs abzugleichen, muss neben der Erweiterung der Software auch die Information, welche Mindestanforderungen der IdP hat, in der Konfiguration stehen.

Service Provider Software

Um den akzeptierten LoA einzuschränken, muss die SP-Software entsprechend konfiguriert werden. Dies geht zum einen über die SAML-Software, über `.htaccess` bei Apache und über die Applikation selbst. Wenn die SAML-Implementierung die Überprüfung des `<EntityAttributes>` Elements unterstützt, kann diese Erweiterung der Metadaten [`SAML2-MetadataAttr`] [LCC09a] genutzt werden, um IdPs zu validieren. IdPs können hiermit über Meta-Tags den eigenen Identity Provider beschreiben. SPs können bestimmte Beschränkungen durchsetzen, indem sie den Tag in den Metadaten oder das entsprechende Attribut der Benutzer filtern und basierend darauf eine Entscheidung treffen. Dieses Prinzip kann in fast allen Fällen verwendet werden.

Ebenso wie beim IdP soll der LoT in den Metadaten des SPs stehen. Der `EntityDescriptor` kann ebenso für LoT eingesetzt werden. Ähnliches gilt für OpenID Connect und OAuth.

Föderationsseitige Realisierung

Föderationen und Communities hatten bisher die Möglichkeit bestimmte Voraussetzungen in Form eines LoA zu bestimmen. Um von den nicht-maschinenlesbaren Dokumenten weg zu kommen und stärker automatische Lösungen einzusetzen, sollen so genannte Profile beschrieben werden können. Eine Föderation oder Community kann Profile festlegen, die als

Mindestanforderungen gelten. Diese Mindestanforderungen hängen von den Risiken für die Beteiligten ab. Langfristig sollen Profile ebenso verschwinden wie feste Föderationen.

```
https://loa.mdfim.net/...%5C%2Fassurance%5C%2Floa
        %5C%3Dloa1%5federation%5D%2Fml=I2.D2.Auth2.A1.T1.O1
```

Für LoA besteht ein Profil aus den Kriterien MLI, MLD, MLAuth, MLA, MLT und MLO. Diese Kriterien sind jeweils mit einer Zahl verknüpft, die die entsprechenden Maturity Level wiedergeben. Bei LoT besteht ein Profil aus den Kriterien MLD, MLT und MLO. Wenn ein Aspekt nicht zutrifft, kann er auch weggelassen werden.

```
https://loa.mdfim.net/...%5C%2Fassurance%5C%2Floa
        %5C%3Dloa1%5federation%5D%2Fml=I2.D2.Auth2.A3.T1
```

Über die Funktion `addProfile(federationID, profileLoA[], profileLoT[])` können Föderationsverwaltungen bestimmte Mindestvoraussetzungen als Profile erstellen. Zugleich können Föderationsverwaltungen Maturity Level bestätigen. Die Bestätigung durch eine unabhängige, vertrauenswürdige Instanz sichert eine höhere Wahrscheinlichkeit zu, dass die Angabe stimmt. Die Bestätigung wird in MdFIM gespeichert und kann für den Vertrauensaufbau verwendet werden, wie in der Architektur zu sehen. Zudem kann lokal die URI zur öffentlichen Seite der verifizierten Trust-Entitäten überprüft werden, die in den Metadaten stehen soll. Diese URI soll angelehnt an die Trust-URI wie folgt aussehen:

```
https://lot.federation.net/mdfim?lot=http%3A%2F%2Ffoo.example.com
        %2Fassurance%2Fml=O3.T3.D2
```

Zunächst wird die Föderation bzw. unabhängige Stelle über das URI-Schema und dem `hier`-Part angegeben. Als Parameter wird der LoT bzw. LoA angegeben mit der EntityID der Entität als Wert. Die Angabe eines unabhängigen Dritten ist äquivalent zum Vorgehensweise in OpenID Connect. Um sie in SAML nutzen zu können, müssen die Metadaten um ein Element `saml:AttributeCertification` erweitert werden, wie in Listing 5.41 zu sehen.

```
<md:EntityDescriptor xmlns="urn:oasis:names:tc:SAML:2.0:metadata"
 xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
 xmlns:attr="urn:oasis:names:tc:SAML:metadata:attribute"
 xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
 entityID="https://www.lrz.de/SAML">
  <md:Extensions>
    <attr:EntityAttributes>
      <saml:Attribute
            NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
            Name="urn:oasis:names:tc:SAML:attribute:assurance-certification">
        <saml:AttributeValue>
                  https://loa.mdfim.net/mdfim?loa=
                  http%3A%2F%2Ffoo.example.com%2Fassurance
                  %2Fml=I1.D2.Auth1.A1.T1.O1
                </saml:AttributeValue>
                <saml:AttributeCertification>
                  https://loa.federation.net/mdfim?
                  loa=http%3A%2F%2Fwww.lrz.de%2SAML
                </saml:AttributeCertification>
          </saml:Attribute>
    </attr:EntityAttributes>
  </md:Extensions>
  ...
</md:EntityDescriptor>
```

Listing 5.41: XML-Element in SAML Metadaten für LoA-Profil mit Erweiterung um Bestätigung des Levels

Um einen LoA oder einen LoT zu akkreditieren und somit zu überprüfen und akzeptieren, erhalten Föderationsverwaltungen eine Web-Schnittstelle. Hierfür ist der `FederationTrustHandler` zuständig. Diese informiert die zuständigen Rollen und folglich Benutzer über E-Mail, wenn eine Entität ihre Maturity Level bestätigt haben will (`informConfirmTrust()`). Beim Einloggen in die Web-Anwendung erscheinen als Erstes ihnen zugeordnete Aufgaben (`displayTasks(userID/federationID)`). In diesem Fall ist es die Überprüfung einer Entität. Neben der Entität und deren Metadaten wird der selbst festgelegte Maturity Grad angezeigt. Die Föderationsbenutzer haben nun die Möglichkeit alle Informationen sich anzuzeigen (`displayInformationLevel(entityID)`), den Level zu bestätigen bzw. nicht zu bestätigen (`confirmTrust(entityID, boolean)`) sowie weitere Informationen anzufragen (`requestInformation(entityID, information)`). Wenn der Level bestätigt wurde und durch die Föderation weitere Aktionen ausstehen, wie eine Signatur, erscheinen diese Aktionen als nächste Aufgaben. Eine Workflow Engine überprüft auch hier den Ablauf der Workflows.

Werkzeug Trust-Assessment

Um die Bestimmung der eigenen Maturity Level zu erleichtern und Verbesserungspotential leichter kennen zu können, hilft eine Webanwendung des MdFIM den Entitäten. Auf Basis der gemachten Antworten, werden die jeweiligen Maturity Level festgelegt. Zudem wird in einer

Übersicht aufgezeigt, in welchen Bereichen Verbesserungen möglich sind, um beispielsweise die Mindestanforderungen einer Föderation zu entsprechen. Zugleich ist eine Umrechnung von bisherigen LoA zu Maturity Level möglich.

Um dies zu realisieren, benötigt das Trust-Assessment mehrere Elemente:

- Fragenkatalog, basierend auf den oben ausgeführten Maturity Level, der pro Aspekt den Grad anhand der Antworten bestimmt.
- Ein Framework für die Visualisierung.
- Informationen über die Mindestanforderungen einer Föderation.
- Umrechnung von LoA zu Maturity Level.

Der Fragenkatalog fragt systematisch das Einhalten der einzelnen Aspekte ab, um anschließend das Ergebnis anzuzeigen (`displayTrustResult(entityID)`). Um einen Vergleich mit den Mindestanforderungen einer Föderation machen zu können (`compareTrustProfile(entityID, federationID)`), muss diese Mindestanforderung erst einmal abgefragt werden (`queryFederationProfile(federationID)`). Für die Umrechnung von LoA zu Maturity Level werden die API (`ttp ttp_compareTrust()`) und damit die Funktion `compareTrust(loa)` aufgerufen. Das Ergebnis wird über `displayCompareTrust(loa)` angezeigt.

5.4.4. Bewertung

In diesem Abschnitt wurden sowohl die Maturity Level für Level of Assurance als auch für ein Level of Trust definiert. Dabei wurde eine Anleitung geschrieben, welche Aspekte bei der Erstellung eines LoA zu beachten sind. Um Maturity Level für Föderationen verwendbarer zu machen, wurde der Begriff Profile eingeführt, der eine Mindestanforderung beschreibt. Zugleich ermöglichen Profile einen Wechsel von Dokumenten zu maschinenlesbaren URIs. Im Rahmen des Qualitätsmanagements können Föderationsverwaltungen die Maturity Level ihrer Mitglieder validieren und ihnen damit ein höheres Gewicht verleihen. Ein Assessment Tool hilft zudem das eigene Maturity Level zu bestimmen, Verbesserungspotential zu erkennen und, falls vorhanden, Anleitungen hierfür zu erhalten. Das Tool ist ferner eine visuelle Hilfe für Benutzer, um sich zu informieren.

Bezogen auf die oben genannten Anforderungen ergibt sich folgendes Bild:

- [LoA/LoT-Auditing]: Beide Verlässlichkeitsklassen können validiert werden und betrachten zudem unterschiedliche Ausprägungen von Audits.
- [LoA/LoT-Automatisierung]: Beide Verlässlichkeitsklassen erlauben die Automatisierung durch Konfiguration und Eintragung in den Metadaten.

- [LoA-Einordnung]: Die Maturity Level für LoA sind so gewählt, dass jede Entität leicht ihre Mindestanforderungen und ihre eigene Einordnung tätigen kann. Zudem hilft das zentrale Werkzeug bisherige LoA in die Maturity Level zu konvertieren.

- [LoT-Datenschutz]: LoT betrachtet den Datenschutz als einen Aspekt.

- [LoA/LoT-Einordnung]: In beide Verlässlichkeitsklassen können Entitäten eingeordnet werden. Zudem hilft ein Werkzeug bei der Einordnung.

- [LoA/LoT-Gewichtung]: Durch die unterschiedlichen Maturity Level kann eine Art Gewichtung durchgeführt werden.

- [LoA/LoT-Implementierungsunabhängigkeit]: Beide Konzepte sowie das zentrale Werkzeug wurden unabhängig von der Implementierung und vom Protokoll entwickelt und können daher universal eingesetzt werden.

- [LoA-Koexistenz]: Durch die Verwendung von Metadaten und einer lokalen Speicherung bzw. einer AA können unterschiedliche Maturity Level für die Benutzer gesetzt werden. Nachdem immer die URN/URI bzw. die Bezeichnung des LoA angegeben werden muss, sind auch unterschiedliche LoA innerhalb einer Entität möglich, wenn auch nur in der Übergangszeit zu befürworten.

- [LoA/LoT-Konfiguration]: Beide Verlässlichkeitsklassen können über die Konfiguration festgesetzt werden. Zudem können Mindestanforderungen an andere Teilnehmer über die Konfiguration gemacht werden. Dasselbe Prinzip gilt für Föderationen, die Profile erstellen und konfigurieren können.

- [LoA/LoT-Protokollunabhängigkeit]: Beide Konzepte sowie das zentrale Werkzeug wurden unabhängig von der Implementierung und vom Protokoll entwickelt und können daher universal eingesetzt werden.

- [LoA/LoT-Realisierbarkeit]: Durch die bereits mögliche Verwendung von LoA sind die Maturity Level für LoA schnell anwendbar. Durch eine Erweiterung ist dies ebenso bei LoT machbar. Im Prinzip müssen die Daten in den Metadaten mit der lokalen Konfiguration abgeglichen werden. Die Maturity Level sind so gewählt, dass jede Entität ihre Mindestanforderungen und ihre eigene Einordnung tätigen kann.

- [LoA/LoT-Registrierung]: Beide Verlässlichkeitsklassen und die dazugehörigen Profile sollen und können in der IANA Registry registriert werden.

- [LoA/LoT-SelfAsserted]: Beide Verlässlichkeitsklassen erlauben es, dass Entitäten ihre Werte selbst festsetzen und bestimmen können. Durch eine Verifikation durch eine vertrauenswürdige Instanz erhöht sich die Wertigkeit dieser Einordnung.

5.4.5. Anwendung

Die Universität hat, wie oben beschrieben, den allgemeinen LoA (I3.D3.Auth2.A3.T2.O3). Nachdem interne Audits eingeführt wurden, erhöht sich der LoA auf (I3.D3.Auth2.A3.T3.O3). Dieser wird in den Metadaten auch angegeben, wie in Listing 5.42 zu sehen.

```
<md:EntityDescriptor xmlns="urn:oasis:names:tc:SAML:2.0:metadata"
 xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
  xmlns:attr="urn:oasis:names:tc:SAML:metadata:attribute"
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  entityID="https://www.lrz.de/SAML">
  <md:Extensions>
    <attr:EntityAttributes>
      <saml:Attribute
            NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
            Name="urn:oasis:names:tc:SAML:attribute:assurance-certification">
        <saml:AttributeValue>
                  https://loa.mdfim.net/mdfim?loa=
                   http%3A%2F%2Ffoo.example.com%2Fassurance
                   %2Fml=I3.D3.Auth2.A3.T3.O3
                  https://loa.mdfim.net/mdfim?loa=
                   http%3A%2F%2Ffoo.example.com%2Fassurance
                   %2Fml=I2.D3.Auth2.A3.T2.O3
                  https://loa.mdfim.net/mdfim?loa=
                   http%3A%2F%2Ffoo.example.com%2Fassurance
                   %2Fml=I4.D3.Auth3.A3.T2.O3
                </saml:AttributeValue>
          </saml:Attribute>
    </attr:EntityAttributes>
  </md:Extensions>
  ...
</md:EntityDescriptor>
```

Listing 5.42: XML-Element in SAML Metadaten für LoA-Profil

Die LoA von Studenten und Mitarbeitern werden zudem in LDAP gespeichert und über das Attribut `eduPersonAssurance` ausgegeben. Hierfür wurde die Konfiguration angepasst (vgl. Listing 5.43).

```
<resolver:AttributeDefinition xsi:type="ad:Simple"
 id="eduPersonAssurance" sourceAttributeID="loa">
    <resolver:Dependency ref="assurance" />
    <resolver:AttributeEncoder xsi:type="enc:SAML1String"
          name="urn:mace:dir:attribute-def:eduPersonAssurance" />
    <resolver:AttributeEncoder xsi:type="enc:SAML2String"
          name="urn:oid:1.3.6.1.4.1.5923.1.1.1.11"
          friendlyName="eduPersonAssurance" />
</resolver:AttributeDefinition>

<resolver:DataConnector xsi:type="dc:RelationalDatabase"
 xmlns="urn:mace:shibboleth:2.0:resolver:dc"
 id="assurance">
```

Listing 5.43: XML-Element im AttributeResolver für LoA

Der SP für den Gitlab-Dienst hat den LoA (I1.D2.Auth1.A1.T1.O1) als Mindestanforderung und erfüllt selbst den LoT von (O3.T2.D2).

```
https://loa.mdfim.net/...%2Fassurance
```

```
https://lot.mdfim.net/mdfim?lot=http%3A%2F%2Ffoo.example.com
         %2Fassurance%2Fml=O3.T2.D2
```

Dieser LoT wurde auch in den Metadaten des SPs angegeben. Die Konfiguration für den IdP ist wie folgt (vgl. Listing 5.44).

```
LoA
        ownLoA: https://loa.mdfim.net/mdfim?loa=
         http%3A%2F%2Ffoo.example.com%2Fassurance
         %2Fml=I3.D3.Auth2.A3.T3.O3
        profile: https://loa.mdfim.net/...%2Fassurance
        %2Floa=loa1%5Bprofile%5D
        federation: federation
        checkedByFederation: true
        category: trusted
        authentication:
LoT
        requestedLoT: https://lot.mdfim.net/...%2Fassurance
        %2Fml=O2.T2.D2
        profile: https://lot.mdfim.net/...%5C%2Fassurance
        %5C%2Flot%5C%3Dlot1%5Bprofile%5D
        federation: federation
        checkedByFederation: true or category
        category: R&S
```

Listing 5.44: Konfiguration des IdPs

Beim SP zeigt sich folgendes Bild (vgl. Listing 5.45):

```
LoA
        requestedLoA: https://loa.mdfim.net/...%2Fassurance
        %2Fml=I1.D2.Auth1.A1.T1.O1
                EntityID1: https://loa.mdfim.net/...%2Fassurance
                %2Fml=I1.D1.Auth1.A1.T1.O1
                EntityID2: https://loa.mdfim.net/...%2Fassurance
                %2Fml=I1.D2.Auth1.A1.T1.O2
        profile: https://loa.mdfim.net/...%2Fassurance
        %2Floa=profile1%5Bfederation%5D
LoT
        ownLoT: https://lot.mdfim.net/mdfim?lot=
          http%3A%2F%2Ffoo.example.com%2Fassurance%2Fml=O3.T3.D2
        profile: https://lot.mdfim.net/...%2Fassurance
        %2Flot=profile1%5Bfederation%5D
        federation: federation
        checkedByFederation: true
        [https://lot.federation.net/mdfim?lot=
          http%3A%2F%2Ffoo.example.com%2Fassurance
          %2Fml=O3.T3.D2]
        category: R&S
```

Listing 5.45: Konfiguration des SPs

Basierend auf der Konfiguration wird durch MdFIM überprüft, ob die Mindestanforderungen, die in den Metadaten stehen, eingehalten werden. Ist dies der Fall, werden die Metadaten ausgetauscht. Ist dies nicht der Fall, wird dem Nutzer eine Fehlermeldung angezeigt und die Aktion geloggt. Wenn eine Entität mehrere Maturity oder Trust Level akzeptiert, findet zudem lokal eine Überprüfung statt.

5.5. Bewertung

In diesem Kapitel wurde der Vertrauensaufbau mittels MdFIM und den zusätzlichen Werkzeugen Conversion Rule Management und Trust Management beschrieben. Die Unterteilung in Metadatenaustausch und Vertrauensinformationen basiert darauf das Vertrauen als verschiedene Schichten (Layer) zu betrachten, wie in Abbildung 5.23 zu sehen. Neben dem technical trust, bestehend aus dem Metadatenaustausch, werden Informationen in den Metadaten kodiert, die den LoA bzw. LoT betrachten. Durch diese Informationen können Entitäten das Vertrauen in die gegenüberliegende Partei besser einschätzen und ihre eigenen Risiken als Mindestanforderungen festhalten. Zusätzlich, über den bisherigen beiden Schichten, liegt der behavioural trust.

Zur Bewertung der entwickelten Werkzeuge wird der Anforderungskatalog aus Kapitel 2 herangezogen. Anforderungen, die bereits durch die Architektur und das Konzept aus dem vorherigen Kapitel erfüllt werden, werden nicht weiter betrachtet.

Die essentiellen Anforderungen wurden wie folgt erfüllt:

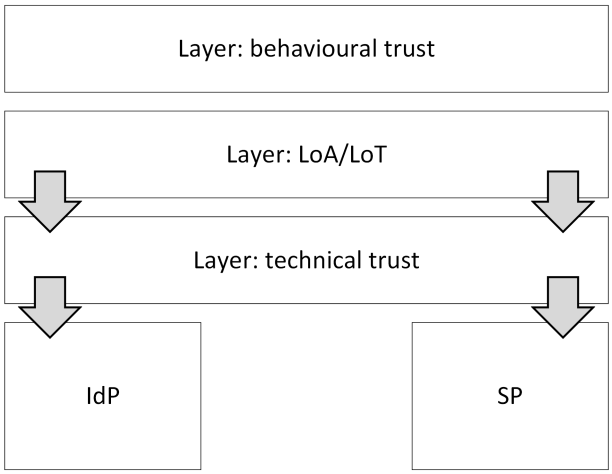

Abbildung 5.23.: Vertrauen basierend auf Schichten

[FA-Realisierbarkeit]: Priorität 1

- Beschreibung: Die Implementierung und die Teilnahme an Föderationen müssen in akzeptabler Zeit möglich sein.
- Bewertung: Die Realisierbarkeit von MdFIM wird durch die Erweiterung von etablierten Lösungen vereinfacht und insbesondere durch die prototypische Anwendung aufgezeigt. Daher kann diese Anforderung als erfüllt angesehen werden.

[NFA-OpenSource]: Priorität 1

- Beschreibung: Die Lösung muss auf offenen Standards basieren und unter den entsprechenden Lizenzen veröffentlicht werden.
- Bewertung: Diese Anforderung wird durch diese Arbeit mit dem Protokoll SAML und der Implementierung basierend auf Shibboleth erfüllt.

[ORG-Realisierbarkeit]: Priorität 1

- Beschreibung: Der Aufwand für Realisierung und Betrieb muss angemessen sein.
- Bewertung: Diese Anforderung kann als erfüllt angesehen werden.

[ORG-Registrierung]: Priorität 1

- Beschreibung: Die Organisationen müssen sich in der Lösung registrieren können. Hierfür sollen Föderationen auch bestimmte Anforderungen für die Aufnahme aufstellen können.

- Bewertung: Die Organisationen können sich bei MdFIM registrieren. Durch die Föderationsverwaltung existiert ein festgeschriebener Aufnahmeprozess.

[DSA-ARPs]: Priorität 1

- Beschreibung: Die Daten, die an einen SP gesendet werden, müssen vorher gefiltert werden. Benutzer müssen dabei kontrollieren können, welche Daten an einen Service Provider gesendet werden.

- Bewertung: Benutzerdaten werden bei SAML-Implementierungen durch ARPs gefiltert. Diese Möglichkeit wird bei der automatischen Konfiguration verwendet. Zudem kann der Nutzer durch uApprove oder ein anderes Consent-Modul kontrollieren, welche Daten an den jeweiligen SP gesendet werden.

Zusammenfassend zeigt sich, dass alle essentiellen Anforderungen, soweit technisch möglich, umgesetzt wurden. Bei den wichtigen Anforderungen zeigt sich folgendes Bild:

[FA-LoA]: Priorität 2

- Beschreibung: Dem SP soll angezeigt werden, welche Datenqualität der IdP liefern kann. Zugleich soll die Einteilung in eine bestimmte Klasse transparent erfolgen. Falls unterschiedliche Klassifikationen verwendet werden, sollen diese gemappt werden können.

- Bewertung: Es wurde ein Trust Management mit LoA definiert.

[FA-LoT]: Priorität 2

- Beschreibung: Die Vertrauenswürdigkeit von SPs soll überprüft und geeignet dargestellt werden.

- Bewertung: Es wurde ein Trust Management mit LoT definiert.

[FA-Schema]: Priorität 2

- Beschreibung: Es soll alle benötigten Attribute unabhängig vom Schema der Inter-Föderation oder Föderation versendet und interpretiert werden können.

- Bewertung: Diese Anforderung wird durch Konvertierungsregeln erfüllt.

[ORG-LoA]: Priorität 2

- Beschreibung: Die Verlässlichkeitsklasse betrifft interne Abläufe und die Konfiguration.

- Bewertung: Diese Anforderung kann als erfüllt angesehen werden.

[ORG-LoT]: Priorität 2

- Beschreibung: Die Vertrauenswürdigkeit von SPs soll organisatorisch realisiert werden können.
- Bewertung: Diese Anforderung kann als erfüllt angesehen werden.

[ORG-Schema]: Priorität 2

- Beschreibung: Ein organisationsübergreifendes Modell zum Datenaustausch soll verwendet werden.
- Bewertung: Diese Anforderung kann als erfüllt angesehen werden.

[DSA-LoT]: Priorität 2

- Beschreibung: Durch bzw. trotz der Funktion Level of Trust soll der Datenschutz eingehalten werden.
- Bewertung: Diese Anforderung kann als erfüllt angesehen werden.

Die nachfolgende Tabelle 5.9 visualisiert die Umsetzung der Anforderungen durch die Architektur und die dazu gehörenden Werkzeuge.

Die Tabelle zeigt auf fast alle Anforderungen erfüllt wurden. Die Aspekte TTP, LoA bzw. LoT und ein universales Schema wurden im vorherigen Kapitel bewusst ausgeklammert und als Black Box betrachtet. Um den Erfüllungsgrad der Anforderungen weiter zu erhöhen, wurden in diesem Kapitel Werkzeuge für die Architektur bereitgestellt. Ebenso wurden die Anforderungen [FA-Realisierbarkeit], [FA-Schema], [NFA-OpenSource], [ORG-Realisierbarkeit], [ORG-Registrierung], [ORG-Schema] und [DSA-ARPs] detaillierter spezifiziert.

Verbesserungen sind bei den nur teilweise erfüllten Anforderungen [FA-Attributswahl], [FA-Identitätswahl], [FA-Monitoring], [FA-SelfAsserted], [NFA-Usability] sowie der nicht erfüllten Anforderung [FA-Entscheidungshilfe] möglich. Monitoring bezieht sich hierbei auf die Informationen aus Nutzersicht. Während über Consent-Management dem Nutzer zwar die gesendeten Benutzerinformationen angezeigt werden, ist eine Verwaltung und ein späteres Widerrufen supoptimal gestaltet. Durch die Bündelung der Anforderungen ist eine Nachfolgearbeit sinnvoll, die sich auf die Verbesserung aus Nutzersicht konzentriert. Dies ist insbesondere dann nützlich, wenn UMA für SAML existiert und diese Arbeit miteinbezogen werden kann.

Anforderung	Priorität	Bewertung	Anforderung	Priorität	Bewertung
Funktionale Anforderungen					
[FA-Aktualisierung]	2	+	[FA-Konnektor]	2	+
[FA-Attributswahl]	2	o	[FA-Kontext]	3	+
[FA-Automatisierung]	2	+	[FA-Langlebigkeit]	1	+
[FA-Datenkategorisierung]	1	+	[FA-LoA]	2	+
[FA-Dynamik]	2	+	[FA-Lokalisierung]	1	+
[FA-Entscheidungshilfe]	3	-	[FA-LoT]	2	+
[FA-Fehlermanagement]	2	+	[FA-Metadaten]	2	+
[FA-Föderation]	1	+	[FA-Monitoring]	2	o
[FA-Grenzüberschreitend]	1	+	[FA-Pull&Push]	1	+
[FA-Homeless]	3	+	[FA-Realisierbarkeit]	1	+
[FA-Identitätswahl]	3	o	[FA-Reichweite]	2	+
[FA-Initiierung]	2	+	[FA-Rollen]	2	+
[FA-Integration]	1	+	[FA-Schema]	2	+
[FA-Interaktion]	1	+	[FA-SelfAsserted]	3	o
[FA-Konfiguration]	1	+	[FA-SLA]	3	+
Nichtfunktionale technische Anforderungen					
[NFA-Dokumentation]	1	+	[NFA-Portabilität]	3	+
[NFA-Implementierungsunabhängigkeit]	2	+	[NFA-Protokollunabhängigkeit]	2	+
[NFA-Koexistenz]	1	+	[NFA-Skalierbarkeit]	1	+
[NFA-OpenSource]	1	+	[NFA-Usability]	2	o
[NFA-Performanz]	2	+			
Sicherheitsanforderungen					
[SEC-ARPs]	1	+	[SEC-Kontext]	3	+
[SEC-Auditing]	2	+	[SEC-LoA]	2	+
[SEC-Authentifizierung]	1	+	[SEC-LoT]	2	+
[SEC-Automatisierung]	2	+	[SEC-Metadaten]	3	+
[SEC-Datenübertragung]	1	+	[SEC-Multilateral]	1	+
[SEC-Initiierung]	2	+	[SEC-Systemsicherheit]	1	+
[SEC-Integration]	2	+			
Organisatorische Anforderungen					
[ORG-Automatisierung]	2	+	[ORG-Realisierbarkeit]	1	+
[ORG-Föderation]	2	+	[ORG-Registrierung]	1	+
[ORG-Konfiguration]	2	+	[ORG-Schema]	2	+
[ORG-LoA]	2	+	[ORG-SLA]	3	+
[ORG-LoT]	2	+	[ORG-Supportprozesse]	2	+
[ORG-Metadaten]	3	+	[ORG-Validierung]	1	+
[ORG-Migration]	2	+			
Datenschutzanforderungen					
[DSA-ARPs]	1	+	[DSA-Interaktion]	2	+
[DSA-CoCo]	3	+	[DSA-LoT]	2	+
[DSA-Datenschutz]	1	+	[DSA-Selbstbestimmung]	1	+
[DSA-Initiierung]	3	+	[DSA-Zustimmung]	2	+

Tabelle 5.9.: Bewertung des Konzepts

Kapitel 6

Prototypische Implementierung

Inhalt dieses Kapitels

In den Kapiteln 4 und 5 wurden eine Architektur für dynamischen Metadatenaustausch und die hierfür benötigten Werkzeuge konzipiert. Um die Realisierbarkeit und Wirksamkeit zu zeigen, wird in diesem Kapitel zunächst die prototypische Implementierung beschrieben, bevor die Architektur und ihre Werkzeuge anhand eines umfassenden Anwendungsbeispiels im nächsten Kapitel erläutert werden.

Zunächst werden die hier vorgestellten *Werkzeuge und Komponenten* in Abschnitt 6.1 begründet. Nachdem durch den dynamischen Metadatenaustausch grundlegende Eingriffe in die Architektur notwendig sind, müssen bestehende Komponenten angepasst und zusätzliche Werkzeuge entwickelt werden. Diese Erweiterungen sollen, genauso wie die Software selbst, Open Source sein [NFA-OpenSource]. Basierend auf dem Implementierungsrahmen und der Auswahl der Komponenten wird die tatsächliche *Implementierung* gezeigt.
Die *Evaluierung der Skalierbarkeit* in Abschnitt 6.2 erfolgt anhand von Szenarien. Das Kapitel wird durch eine Zusammenfassung abgerundet. Die Evaluierung der Performance wurde weggelassen, nachdem hierfür die Testumgebung zu klein war bzw. es stärkere Änderungen bei der Implementierung nach sich gezogen hätte. Diese Vorgehensweise ist auch in Abbildung 6.1 dargestellt.

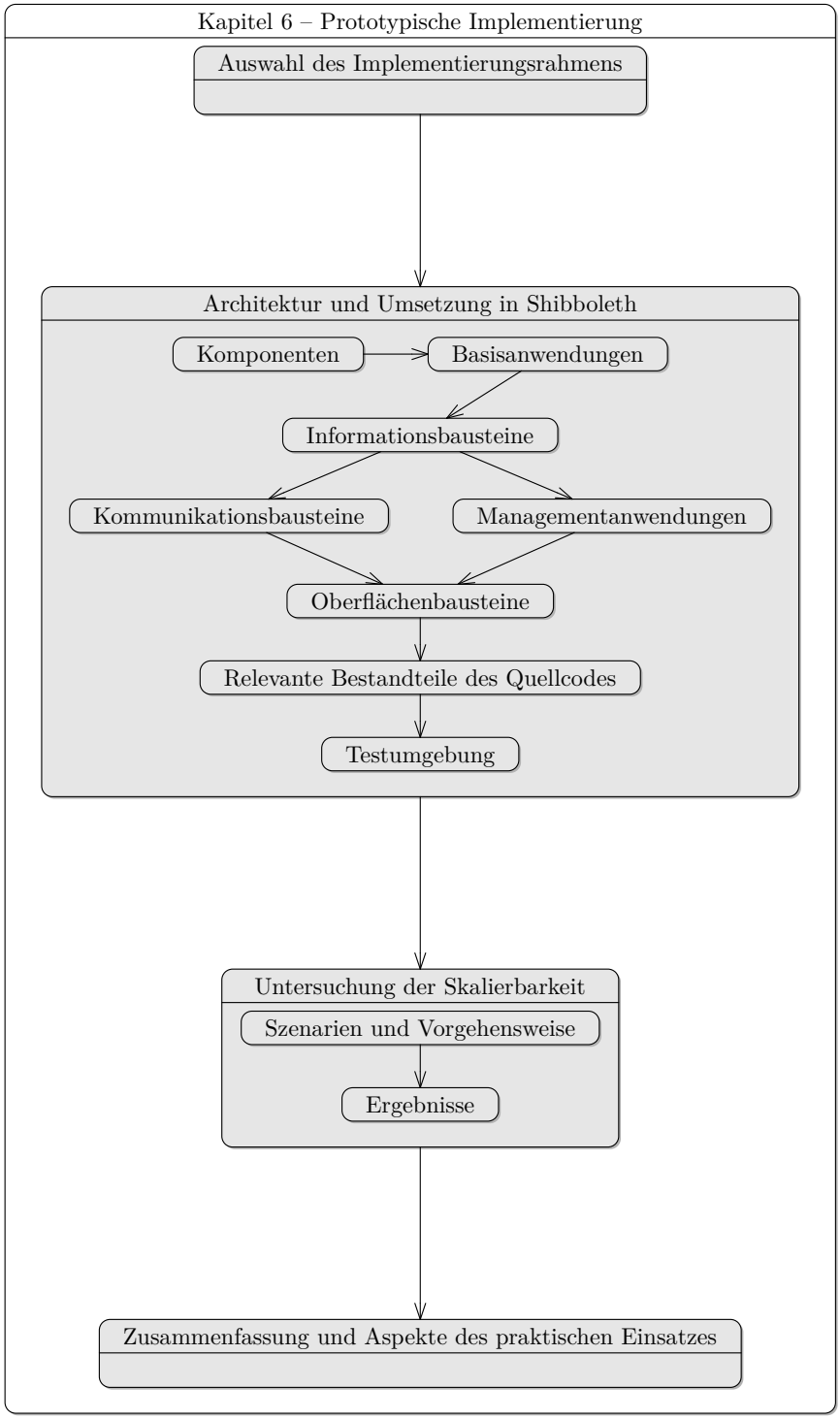

Abbildung 6.1.: Vorgehensmodell in diesem Kapitel

Die prototypische Implementierung wurde im Rahmen der Masterarbeit von Michael Grabatin [Gra14] implementiert und im Rahmen des GÉANT Projektes für das Projekt bezogen weiter entwickelt (vgl. [PMH14e], [PGM+15] und [PvWP16]), so dass hier zum großen Teil die Ergebnisse vorgestellt werden. Anhand einer Testumgebung wird der beispielhafte Aufbau gezeigt. Für die beschriebene Architektur wurde eine rudimentäre, erweiternde Implementierung vorgenommen, um die Skalierbarkeit zu beurteilen und die praktische Nutzbarkeit zu demonstrieren.

6.1. Auswahl des Implementierungsrahmens und Umsetzung in Shibboleth

Der Implementierungsrahmen legt fest, welche Konzepte und Werkzeuge implementiert werden und begründet diese Auswahl. Gleichzeitig werden Ausnahmen gemacht, die großteils anschließend theoretisch besprochen werden.

Eine vollständige Umsetzung aller vorgeschlagenen Änderungen war im Rahmen dieser Arbeit auf Grund des Implementierungsaufwandes nicht möglich und wäre nur sinnvoll gewesen, wenn eine produktionsreife Lösung benötigt wird. Zudem wurde eine vereinfachte, spezialisierte Lösung der in dieser Arbeit definierten Architektur für das GÉANT-Projekt implementiert.

Als Konsequenz wurde eine Priorisierung vorgenommen, die auf den folgenden Aspekten beruht:

- Tragfähigkeitsnachweis der hier vorgestellten Architektur und Werkzeuge.
- Konzentration auf wesentliche Neuerungen.
- Modifikationen der Kernfunktionalitäten von IdP und SP Software sollten vermieden werden.
- Implementierte Komponenten sollen sinnvoll eingesetzt werden können, ohne gesamte Föderationen und Inter-Föderationen ändern zu müssen.
- Vorhandene Implementierungen im Rahmen des GÉANT-Projekts sind auf Grund des Aufwands vorzuziehen.

Die *prototypische Implementierung* bildet die Basis für die Evaluation der erstellten Architektur und der zugehörigen Werkzeuge. Die Implementierung des Proof of Concepts konzentriert sich SAML, da SAML eine gute Grundlage für Erweiterungen bildet und bestehende Systeme in R&E sowie in der Wirtschaft häufig auf SAML aufbauen. Dabei wird insbesondere auf Shibboleth aufgesetzt, weil diese SAML-Implementierung die meisten Anforderungen erfüllt, die im Kriterienkatalog aufgestellt wurden. Zwar zeigt SimpleSAMLphp eine leichtere

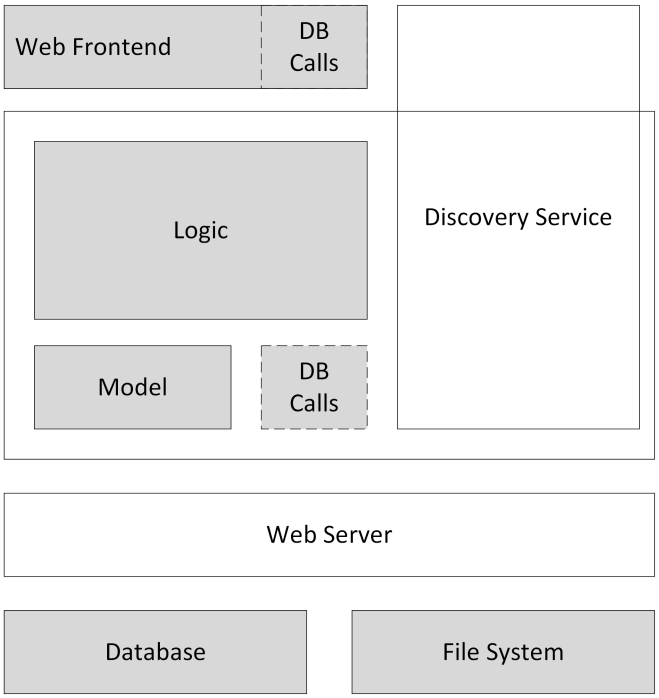

Abbildung 6.2.: Grundlegende Architektur der MdFIM aus [PMH14e]

Erweiterung durch die Modularität und die Unterstützung mehrerer Protokolle. Dafür bietet Shibboleth bessere Anleitungen sowie eine feingranulare, verständliche Konfiguration. Ferner soll die Erweiterbarkeit mit Shibboleth Version 3.x nachgezogen werden. In Shibboleth vorhandene Funktionalitäten, wie beispielsweise [FA-Datenkategorisierung], [FA-Konnektor] und [FA-Rollen], können dadurch weiter genutzt werden.
Die SAML-Implementierung Shibboleth besteht, wie in Kapitel 3 aufgezeigt, aus mehreren Komponenten:

- Software für IdP, wobei die IdP-Software ebenfalls für AAs eingesetzt werden kann. Sie basiert auf Java und läuft auf einem Webserver, wie Tomcat.
- Software für den SP, die in C/C++ geschrieben ist.
- Embedded Discovery Service, der optional ist und beim SP läuft.
- Centralized Discovery Service, der für die zentrale Auswahl des IdPs zuständig ist. Dieser Lokalisierungsdienst ist in Java geschrieben und läuft, wie der IdP, auf einem Webserver.

Die prototypische Implementierung fand auf Basis von Shibboleth 2 statt, wobei die IdP-Erweiterung für Version 3 nachgezogen wurde. Um die Managementplattform MdFIM zu

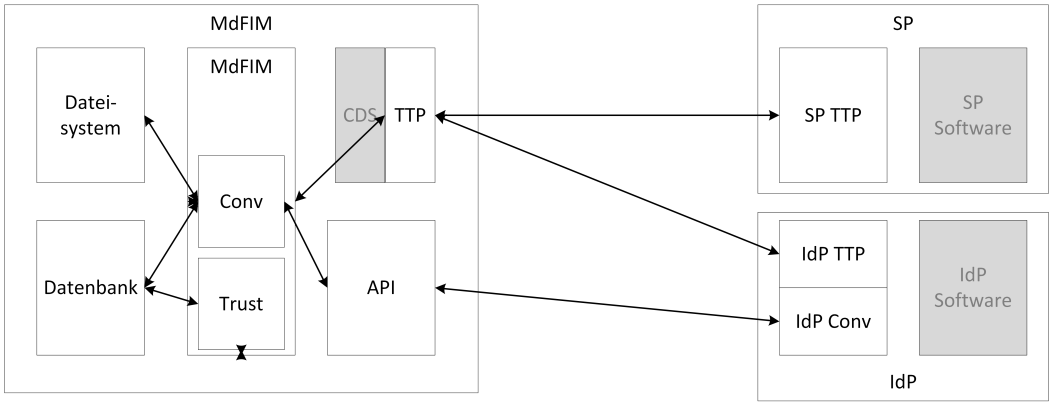

Abbildung 6.3.: Übersicht über die implementierten Komponenten

realisieren, wurde der Centralized Discovery Service um eine Managementanwendung erweitert, die auf Java Server Pages (JSP) beruht. Datenbank und ein Dateisystem dienen zur Speicherung der Metadaten und Konvertierungsregeln, während eine Webanwendung die Verwaltung ermöglicht. Die Logik beschreibt die interne Verarbeitung inklusive der Einhaltung der Workflows. Diese grundlegende Architektur ist in Abbildung 6.2 dargestellt.

Nachfolgend werden zunächst bestehende Komponenten kurz erklärt, die für die Erweiterung relevant sind, bevor die prototypische Implementierung basierend auf den Bausteinen Basisanwendung, Information, Kommunikation, Management und Oberfläche beschrieben wird.

6.1.1. Komponenten

Bei der Auswahl der zu implementierenden Komponenten wurde darauf geachtet, dass sie die Kernfunktionen sowie die wichtigsten Änderungen zeigen. Diese Kernfunktionen demonstrieren die Machbarkeit des in Kapitel 4 aufgestellten Konzeptes und der damit verbundenen, in Kapitel 5 beschriebenen Werkzeuge:

Dynamische Metadatenaustausch und MdFIM: Um die Realisierbarkeit des dynamischen Metadatenaustausches und der Managementplattform zu zeigen, wird der Metadatenaustausch mit einer einfachen Managementfunktionalität für Shibboleth implementiert. Damit eine lokale Erweiterung auf Seiten von IdP, SP und AA inklusive Konfigurationsmöglichkeiten gezeigt wird, wird hierbei eine Erweiterung für den dynamischen Metadatenaustausch implementiert, der grundlegend konfiguriert werden kann.

Conversion Rule Management: Um das Conversion Rule Management zu skizzieren, wird eine Speicherung der generischen Konvertierungsregeln ermöglicht und die Implementierung des Conversion Rule Managements in Shibboleth prototypisch realisiert.

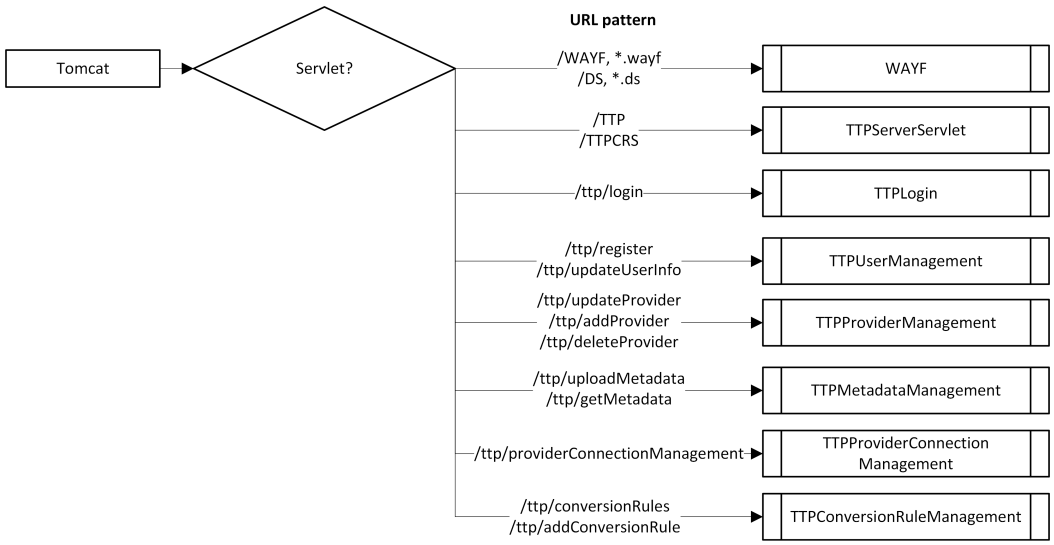

Abbildung 6.4.: Übersicht über die URL-Patterns der TTP [Gra14]

Trust Management: Der Kern des Trust Managements ist der Vergleich der verschiedenen Trust Levels anhand der definierten Maturity Level. Hierfür gibt es die Komponenten MdFIM mit der zentralen Konvertierungstabelle und die lokale Erweiterungen, die ebenfalls die Trust Levels überprüfen. Die Realisierbarkeit des Trust Managements wird durch die prototypische Implementierung des Vergleichs für LoA auf Seite der MdFIM gezeigt.

Diese Auswahl spiegelt sich auch in der Selektion der zu implementierenden FIM-Komponenten wieder, die in Abbildung 6.3 dargestellt ist.

6.1.2. Basisanwendungen

Um MdFIM nutzen zu können, werden unterschiedliche Basisanwendungen benötigt. Die entsprechenden Servlets der TTP werden über URL-Pattern ausgewählt, vgl. Masterarbeit von Michael Grabatin [Gra14]. Basierend auf der URL wird ein HTTP POST Request geschickt, um eine bestimmte Aktion auszuführen. Die in der prototypischen Implementierung verwendeten URL-Pattern sind in der Abbildung 6.4 dargestellt.

Bei der Registrierung [ORG-Registrierung] muss überprüft werden, ob die Entität mit der EntityID der neuen Entität schon existiert. Das Hinzufügen wie auch das Modifizieren von bereits vorhandenen Entitäten geschieht über das Entitätenmanagement. Als URL-Pattern wird bei dieser Basisanwendung `ttp/addProvider` eingesetzt. Nach einer Validierung auf Einhaltung der Längenbegrenzung und der möglichen Existenz der Entität wird die Entität in die Datenbank eingefügt. Damit eine neue Entität erfolgreich hinzugefügt werden kann,

muss auch ein neuer Benutzer registriert werden. Für den HTTP POST Request zum Anlegen eines Benutzers wird das URL-Pattern `ttp/register` verwendet. Nachdem sich der Benutzer registriert hat, kann er sich erneut einloggen. Hierfür ist das Login-Servlet zuständig, welches das URL-Pattern `/ttp/login` verwendet. Ist die Überprüfung des Passwortes erfolgreich, wird eine neue Sitzung für den Benutzer initialisiert. Hierfür wird ein Benutzer-Objekt gespeichert, welches die Benutzer ID und den Namen des Benutzers enthält.

Um zu verifizieren [ORG-Validierung], dass der Benutzer die Entität verwenden darf, muss dieser eine Datei auf der Domain der Entität hosten. Um den URL-Pfad zu generieren, wird zunächst die Domain aus der EntityID der Entität extrahiert. Anschließend wird über `java.security.SecureRandom.SecureRandom()` eine 128 Byte Zufallszahl generiert und als Hexadezimalzahl dargestellt. Beispielsweise muss die Entität Example mit der EntityID `https://idp.example.com/idp/shibboleth` eine Datei für die URL `http://idp.example.com/1o1gh7lfd4fe4kac5gaa4ksahd` erstellen. Diese URL wird in der Sitzung des Benutzers gespeichert und anschließend verifiziert. Der Benutzer muss, nach Erstellen der Datei, bestätigen, dass er diese angelegt hat. Das Servlet versucht die Datei abzurufen und akzeptiert die Verknüpfung, wenn als Antwort der Status `200 OK` gesendet wird. Andernfalls wird der Vorgang abgebrochen und gegebenenfalls wiederholt.

6.1.3. Informationsbaustein

Neben der Datenbank und dem Dateisystem auf Seiten der MdFIM, müssen bei Identity Provider und Service Provider die heruntergeladenen Metadaten gespeichert werden. Diese Erweiterungen sind in Java für die IdP-Software bzw. in C/C++ für die SP-Software implementiert.

Informationsbaustein beim Identity Provider

Die Shibboleth Identity Provider Software verwendet bei Version 3 das Framework Spring, wodurch Workflows leichter definiert werden können. Dies eignet sich somit insbesondere bei der Realisierung des kompletten Funktionsumfanges von MdFIM, da an dieser Stelle viele Workflows, zum Beispiel für Föderationsverwaltungen und der Verwaltung von Konvertierungsregeln, benötigt werden. Im Folgenden (vgl. Listing 6.1) ist die `dame-flow.xml` in Ausschnitten dargestellt. Der Dynamic Automated Metadata Exchange (DAME) Flow bezieht sich auf den Workflow von DAME, d. h. des dynamischen Metadatenaustausches über die TTP. DAME ist ein I-D bei der IETF zum orchestrierten Metadatenaustausch zwischen Identity Provider und Service Provider, der durch die Autorin dieser Arbeit und Kollegen innerhalb des Projektes GNTB spezifiziert wurde und folglich auch für die Interoperabilität verwendet wird.

```
<flow xmlns="http://www.springframework.org/schema/webflow"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.springframework.org/schema/webflow http://www
    .springframework.org/schema/webflow/spring-webflow.xsd">

  <action-state id="Preparation">
    <evaluate expression="InitializeProfileRequestContext" />
    <evaluate expression="InitializeDameContext" />
    <evaluate expression="CheckAccess" />
    <evaluate expression="'proceed'" />
    <transition on="proceed" to="DameDownload" />
  </action-state>

  <action-state id="DameDownload">
    <evaluate expression="DownloadMetadata" />
    <evaluate expression="'proceed'" />
    <transition on="proceed" to="DameRefreshMetadataResolver" />
  </action-state>

  <action-state id="DameRefreshMetadataResolver">
    <evaluate expression="UpdateMetadataResolver" />
    <evaluate expression="'proceed'" />
    <transition on="proceed" to="GetConversionRules" />
  </action-state>

  ...
  <end-state id="replyOk">
    <on-entry>
      <evaluate expression="RecordResponseComplete" />
    </on-entry>
  </end-state>

  <end-state id="replyError">
    <on-entry>
      <evaluate expression="HandleError.handleEvent(currentEvent)">
        <!-- <attribute name="event" value="currentEvent">
        </attribute> -->
      </evaluate>
      <evaluate expression="RecordResponseComplete" />
    </on-entry>
  </end-state>
  ...
</flow>
```

Listing 6.1: Ablauf des Metadatenaustausches bei IdP3

Damit der IdP Metadaten dynamisch herunterlädt, wird ein Metadaten Provider benötigt. Das Metadata Synchronization Servlet speichert die ausgetauschten Metadaten in dem Ort, der durch das Metadata Storage Directory vorgegeben wurde.

Informationsbaustein beim Service Provider

Die Erweiterung des SPs benötigt ebenfalls einen `MetadataProvider`, der die von MdFIM synchronisierten Metadaten verwaltet. Der `MetadataProvider` wird von Handlern, wie dem `SAML2SessionInitiator`, aufgerufen. Ein `MetadataSyncHandler` ist anschließend für die Synchronisation der Metadaten zuständig. Der `MetadataProvider` für die MdFIM heißt `TTPMetadataProvider`. Der Basis `MetadataProvider` besitzt zusätzliche Funktionen für

- das Hinzufügen und Entfernen von Filtern und
- das Abrufen der Metadaten bzw. `EntityDescriptoren`.

Ein Beobachter (Observer), `ObservableMetadataProvider`, ist beim `MetadataProvider` registriert und wird bei Änderungen an den Metadaten benachrichtigt. Damit nach dem Herunterladen einer neuen Metadatendatei der Benachrichtungsprozess angestoßen wird, wird die Funktion `void metadataDownloaded()` aufgerufen. Der `AbstractMetadataProvider` definiert zwei neue `resolve()` Funktionen, um nach Metadaten suchen zu können. Der Dateiname wird durch die Funktion `TTPUtils::getMetadataFilePath` generiert. Wird die Datei gefunden, wird diese durch einen `Unmarshaller` aus dem XML-Dokument in ein `EntityDescriptor`-Objekt umgewandet und ein Zeiger auf das erstellte Objekt zurückgegeben. Der `DynamicMetadataProvider` besitzt einen Locking-Mechanismus, um bei anderen Prozessteilen einen Lese-Lock zu setzen. Dies ist relevant, um konsistente Ergebnisse bei Anfragen zu erreichen. Die Konfiguration des `MetadataProviders` erfolgt über die Shibboleth-Konfigurationsdatei `shibboleth2.xml`, die für SPs zuständig ist. Diese enthält durch die Erweiterung ein neues `MetadataProvider`-Element vom Typen `TTPMetadataProvider`.

6.1.4. Kommunikationsbaustein

Der eben beschriebene Speicherort für die Metadaten sowie die Registrierung der Entitäten ist entscheidend für den dynamischen Metadatenaustausch, dessen Implementierung in diesem Abschnitt gezeigt wird. Zunächst wird die Erweiterung beim SP skizziert, der den ersten Authentication Request an die MdFIM schicken muss.

Kommunikationsbaustein beim Service Provider

Um den Request abzuändern, wird beim SP das Modul `mod_shib` in den Apache2 Prozess des Webservers eingebunden. Das Modul wird in der Datei `mod_apache.cpp` implementiert. Dies erlaubt dem Modul die Behandlung eingehender Requests in diesem geschützten Bereich, wie auch die Auswahl der entsprechenden Handler. Für den Authentication Request ist der `SAML2SessionInitiator` verantwortlich, der Teile der Verarbeitung an den Shibboleth Prozess `shibd` weiterleiten kann. Die `run()` Methode ist zuständig für die Entscheidung der

Weiterleitung.
Die Erstellung eines neuen Authentication Requests wird durch den `shibd` Prozess in einem eigenen Thread verarbeitet. Der `ListenerService` ist für die Wahl des passenden Handlers verantwortlich, an den die Nachricht über die Methode `receive()` weitergeleitet wird. Durch `receive()` wird die Antwort, ein als Pointer übergebenes Response-Objekt, für den Apache2 Prozess modifiziert.

Entscheidend für den ersten Request an MdFIM ist der entwickelte `SessionInitiator`. Der `SessionInitiator` erweitert die `run` Methode des `AbstractHandlers` um den für den dynamischen Metadatenaustausch benötigten Parameter `entityID`. Der `TTPSessionInitiator`, abgeleitet von den Klassen `SessionInitiator`, `AbstractHandler` und `RemoteHandler`, ist dafür zuständig, der TTP den ersten Request an den Identity Provider zu senden, ohne die Metadaten des IdPs zu kennen. Hierfür wird ein Request an die `run` Funktion mit dem Parameter `entityID` übergeben. Der dadurch generierte Request wird verpackt und an den `shibd` Prozess geschickt. Der Prozess

- überprüft, ob die Metadaten bereits vorhanden sind,
- verarbeitet den generierten Request und
- erstellt daraus einen Authentication Request.

Die Zieladresse des Requests, d. h. die TTP, wird aus der Konfiguration bestimmt und mit dem Parameter `entityID` konkateniert.

Um nach der erfolgreichen Authentifizierung des Benutzers die Metadaten des IdPs herunter zu laden und zu integrieren, wird die Funktion `processMessage` verwendet. Diese Funktion extrahiert die Parameter `entityID` und `location` aus dem Request und speichert diese Werte. Basierend daraus wird der Pfad zur lokalen Metadatendatei berechnet und über die Funktion `downloadFile` heruntergeladen.

Kommunikationsbaustein bei MdFIM

MdFIM bzw. die TTP übernimmt die Orchestrierung des Metadatenaustausches. Zunächst muss der Request des SPs gespeichert werden, bevor ein neuer Request versendet wird. Anschließend werden die Metadaten ausgetauscht, ehe der ursprüngliche Request an den IdP weitergeleitet werden kann. Der Request des SPs wird zunächst überprüft, um anschließend zusammen mit Relaystate und den EntityIDs von SP und IdP in der Tomcat-Session gespeichert zu werden. Dies ist notwendig, um den ursprünglichen Request nach dem Metadatenaustausch an den Identity Provider weiterleiten zu können. Wenn ein anderer Webserver verwendet wird als Apache Tomcat, wird dessen Session eingesetzt. Zudem wird mit Hilfe der Datenbank überprüft, ob die Verbindung überhaupt erlaubt ist. An dieser Stelle können Level of Assurance verglichen werden. Die Überprüfung der Erlaubnis wird in beide Richtungen durchgeführt, wie in Listing 6.2 zu sehen.

```
1  private void handleAuthentication (HttpServletRequest req,
2   HttpServletResponse res)
3     throws WayfRequestHandled,
4
5     WayfException {
6        String idpEntityID, spSamlRequestBase64, spSamlRequestRelayStateStr,
7        spSamlRequestSigAlgStr, spSamlRequestSigBase64, spSamlRequestStr;
8        byte[] spSamlRequestDecoded;
9
10       idpEntityID = req.getParameter(SP_AUTH_REQ_IDP_ENTITYID_ATTRIBUTE_NAME);
11       spSamlRequestSigAlgStr = req.getParameter(
     SP_AUTH_REQ_SIG_ALG_ATTRIBUTE_NAME);
12       spSamlRequestSigBase64 = req.getParameter(SP_AUTH_REQ_SIG_ATTRIBUTE_NAME
     );
13       ...
14       verifyRequestSig(spSamlRequestSigAlgStr,spSamlRequestSigBase64);
15       spSamlRequestStr = TTPUtils.gzdeflate(spSamlRequestDecoded);
16       ...
17
18       AuthnRequest spAuthnRequest = getAuthnRequestFromString(spSamlRequestStr
     );
19
20           // Test if the LoAs are compatible
21       TTPProviderLoA requestedLoA = db.getProviderLoAByID(idp.
     getRequestedLoAID());
22       TTPProviderLoA availableLoA = db.getProviderLoAByID(sp.getProviderLoAID
     ());
23       if(requestedLoA != null && availableLoA != null){
24       if(!requestedLoA.isCompatibleWith(availableLoA)){
25                ....
26       ...
27       if (!db.isProviderConnectionAllowed(sp.getId(), idp.getId())){
28           ...
29           }
30           ...
31    }
```

Listing 6.2: Verifizierung des Authentication Requests [Gra14]

Um den neuen Authentication Request an den IdP zu schicken, wird das HTTP-Redirect-Binding eingesetzt. Nach der erfolgreichen Authentifizierung des Nutzers, wird der Metadatenaustausch durch die TTP initiiert. Die Methode `doMetadataExchange` überprüft zudem, ob die Metadaten heruntergeladen wurden (vgl. Listing 6.3).

```
1  private void doMetadataExchange(String idpEntityID,
2   String spEntityID, HttpServletRequest req)
3   throws WayfException, IOException {
4
5   if (initiateMetadataDownload(idpEntityID, spEntityID, req) !=
       HttpURLConnection.HTTP_OK) {
6     log.warn("The␣remote␣({})␣did␣not␣successfully␣download␣the␣metadata",
       idpEntityID);
7   } else {
8     log.info("The␣remote␣({})␣successfully␣downloaded␣the␣metadata",
       idpEntityID);
9   }
10
11  if (initiateMetadataDownload(spEntityID, idpEntityID, req) !=
       HttpURLConnection.HTTP_OK) {
12    log.warn("The␣remote␣({})␣did␣not␣successfully␣download␣the␣metadata",
       spEntityID);
13  } else {
14    log.info("The␣remote␣({})␣successfully␣downloaded␣the␣metadata", spEntityID
       );
15  }
16 }
```

Listing 6.3: Methode für den Metadatenaustausch [Gra14]

Die Methode `doMetadataExchange` ruft wiederum die Methode `initiateMetadataDownload` auf, die den Metadatenaustausch initiiert.

Kommunikationsbaustein beim Identity Provider

Um die Metadaten lokal beim IdP zu integrieren, wird das Event `handleDownload` benötigt. Der eigentliche Download geschieht über `doDownload`. Der Request zum Download wird dahingegen überprüft, ob die EntityID angegeben wurde und ob eine URL vorhanden ist. Dies ist in Listing 6.4 dargestellt.

```
1  private Event handleDownload(HttpServletRequest req,
2          ProfileRequestContext profileRequestContext, DameContext dameContext)
       {
3
4          ...
5          String entityID = req.getParameter(DOWNLOAD_ENTITY_PARAM_NAME);
6          if (entityID == null || entityID.isEmpty()) {
7                  log.error("Could␣not␣download␣metadata,
8  ␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣no␣entityID␣supplied.");
9                  return ActionSupport.buildEvent(profileRequestContext,
10                         EventIds.INVALID_MESSAGE);
11         }
12         ...
13         String location = req.getParameter(DOWNLOAD_URL_PARAM_NAME);
14         if (location == null || location.isEmpty()) {
15                 log.error("Could␣not␣download␣metadata,
16 ␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣no␣location␣supplied.");
17                 return ActionSupport.buildEvent(profileRequestContext,
18                         EventIds.INVALID_MESSAGE);
19         }
20
21         URL url;
22         try {
23                 url = new URL(location);
24         } catch (MalformedURLException e) {
25                 log.error("Could␣not␣create␣URL:␣{}", e.getMessage());
26                 return ActionSupport.buildEvent(profileRequestContext,
27                         EventIds.INVALID_MESSAGE);
28         }
29         dameContext.setSubjectMetadataUrl(url.toString());
30
31         Path metadataFilePath = Paths.get(
32                 dameContext.getMetadataStorageDirectory(), entityHash);
33         long fileSize = doDownload(url, metadataFilePath.toFile());
34         if (fileSize > 0) {
35                 dameContext.setSubjectEntityId(entityID);
36                 dameContext.setSubjectEntityIdHash(entityHash);
37                 log.info("Downloaded␣{}␣bytes␣of␣metadata␣for␣entityId␣{}.",
38                         fileSize, entityID);
39                 return ActionSupport.buildProceedEvent(profileRequestContext);
40         } else {
41                 log.error("Download␣unsuccessful");
42                 return ActionSupport.buildEvent(profileRequestContext,
43                         EventIds.INVALID_MESSAGE);
44         }
45 }
```

Listing 6.4: Handling des Requests zum Download

6.1.5. Managementanwendungen

Der eben beschriebene Kommunikationsbaustein ist die Basis für die Managementanwendungen. Ohne die Kommunikation kann weder die Managementplattform MdFIM, noch das Trust Management oder das Conversion Rule Management funktionieren. Im Folgenden wird die Implementierung dieser drei Werkzeuge

- MdFIM,
- Trust Management und
- Conversion Rule Management

an Beispielen gezeigt, um die Realisierbarkeit zu demonstrieren.

Managementanwendung MdFIM

Nachdem mit dem Centralized Discovery Service der Einstieg für den Metadatenaustausch wiederverwendet werden kann [FA-Initiierung], basiert die Implementierung der Managementplattform auf Java und JSP. Der Discovery Service selbst ist nicht modifiziert. Durch diesen modularen Aufbau kann er ausgetauscht und aktualisiert werden. In der Erweiterung sind die folgenden neuen Klassen und Handler für das Entgegennehmen von Anfragen zuständig:

- `TTPServerServlet`: Entgegennehmen von Anfragen.
- `TTPServerMetadataSyncHandler`: Synchronisieren von Metadaten.
- `TTPServerConversionRuleSyncHandler`: Synchronisieren von Konvertierungsregeln.

Das Servlet ist zugleich Einstiegspunkt, um die Konfiguration zu laden und eingehende Anfragen an Handler weiterzuleiten. Im Fall des Metadata Managements erlaubt es dem Administrator Metadaten hochzuladen und zu aktualisieren. Durch die Versionierung ist das Aktualisieren ähnlich wie das Hochladen neuer Metadaten. Als Parameter wird `metadata` benötigt, also die Metadaten der Entität. Die EntityID wird automatisch aus den Metadaten gefiltert und mit der EntityID des MdFIM-Nutzers verglichen. Wenn die Validierung erfolgreich war, werden die Metadaten gespeichert. Im Fall dieser prototypischen Implementierung wurde auf ein Repository, wie in Kapitel 4 beschrieben, verzichtet und stattdessen eine Dateiablage `ttpMetadataStorageDirectory` verwendet. Die Dateiablage hat zwar im Gegensatz zum Repository den Nachteil, dass eine Versionierung nur durch Nummerierung möglich ist, jedoch ist eine Dateiablage einfacher zu verwenden und einzubinden. Sie genügend zudem, um das Speichern und Abrufen von Dateien, d. h. Metadaten und Konvertierungsregeln, zu zeigen, wie im Listing 6.5 dargestellt.

```
if(metadata != null && !metadata.isEmpty()){
    if(storageDirectory != null && !storageDirectory.isEmpty()){
        // validation
        EntityDescriptor entityDescriptor = parseStringToEntityDescriptor(
    metadata);
        if(entityDescriptor == null) {
                throw new TTPWebError("Could not parse metadata");
        }
        provider = db.getProviderByEntityID(entityDescriptor.getEntityID());
        if(provider == null){
                throw new TTPWebError("Metadata did not match any provider");
        }
        if(!db.canUserEditProvider(user.getId(), provider.getId())){
            throw new TTPWebError("Permission denied");
        }
        // write metadata into filesystem
        BufferedWriter writer = null;
        try {
            // filename is SHA-1 hash with timestamp
            String filename =
                    TTPUtils.getSHA1Sum(entityDescriptor.getEntityID())+"."
                      +System.currentTimeMillis();
            File f = new File(ttpMetadataStorageDirectory+"/"+filename);
                    if(f.exists()){
                        log.error("File already exists");
                            throw new TTPWebError("Internal Server Error");
                    }
            ...
                    // insert Metadata
                    java.util.Date now = new java.util.Date();
                    TTPMetadata md =
                      new TTPMetadata(entityDescriptor.getEntityID(),
                    filename, comment,
                    ((TTPUser) session.getAttribute("userObj")).getId(),
                    ...
                    new Date(now.getTime()));
                    if(md != null){
                        // use BufferedWriter for insert
                            if(db.insertMetadata(md) > 0){
                                writer = new BufferedWriter(new FileWriter(f));
                                    writer.write(metadata);
                            } else {
                                throw new TTPWebError("Could not insert metadata
                                            to database");
                            }
                  ...
```

Listing 6.5: Beispiel des Metadata Managements [Gra14]

Managementanwendung Trust Management

Wie in Listing 6.2 gezeigt, wird vor dem Metadatenaustausch bereits überprüft, ob

- der SP genügend Vertrauen, LoT, für den IdP hat und ob
- der IdP genügend Vertrauen, LoA, für den SP hat.

Diese Überprüfung findet im Fall der prototypischen Implementierung mit Hilfe der Datenbank statt, in der gewünschte und verfügbare Vertrauenswerte, LoA, stehen. Die Überprüfung findet während der Funktion `handleAuthentication` im `TTPServerMetadataSyncHandler` statt, bevor die Metadaten überhaupt ausgetauscht werden. Dies ist im Listing 6.6 dargestellt.

```
1  // Test if the LoAs are compatible
2      TTPProviderLoA requestedLoA = db.getProviderLoAByID(idp.getRequestedLoAID
       ());
3      TTPProviderLoA availableLoA = db.getProviderLoAByID(sp.getProviderLoAID())
       ;
4      if(requestedLoA != null && availableLoA != null){
5        if(!requestedLoA.isCompatibleWith(availableLoA)){
6          throw new WayfException(
7            "Providers have incompatible level of assurance.");
8        } else {
9          log.info("Providers have compatible level of assurance");
10       }
11     } else if(requestedLoA != null && availableLoA == null){
12       throw new WayfException(
13           "The IdP does not provide a LoA, but the SP requests one.");
14     } else {
15       log.info("No LoAs are defined");
16     }
```

Listing 6.6: Beispiel der LoA-Überprüfung in handleAuthenticaton

An dieser Stelle kann eine Umrechnung der Trust-Werte einbezogen werden.

Managementanwendung Conversion Rule Management

Die Verwaltung der Konvertierungsregeln ist äquivalent der Verwaltung der Metadaten. Die Anwendung selbst ist in Python geschrieben und verwendet das Tornado-Framework, welches selbst bei sehr vielen offenen Verbindungen gut skaliert. Das Conversion Rule Management verwendet das URL-Pattern `# /api/conversionrule/\(?P<rule_id>[a-zA-Z0-9]+)/?$` für die Verwaltung von Konvertierungsregeln. Als Parameter werden bei der prototypischen Implementierung

- `name` für den Namen,
- `target` für das Zielattribut,
- `source` für ein Array an Quellattributen und

- `type` für die Art der Konvertierung sowie
- `parameter` als Parameter für die eigentliche Konvertierung verwendet.

Nach der erfolgreichen Validierung, wird die Regel in die Datenbank eingefügt. Bedingt durch das feste Schema einer Konvertierungsregel ist die Validierung vor allem an das Schema gebunden, wie in Listing 6.7 zu sehen.

```
@schema.validate(
        input_schema={
            "type": "object",
            "properties": {
                "name": {"type": "string"},
                "source": {"type": "array", "items": {"type": "integer"}},
                "target": {"type": "integer"},
                "type": {"type": "string"},
                "parameter": {"type": "string"}
            },
            "required": ["name", "source", "target", "type"]
        },
        input_example={
            "name": "cn",
            "source": [1],
            "target": 2,
            "type": "scope",
            "parameter": "@example.com"
        },
        output_schema={"type": "null"}
    )
```

Listing 6.7: Validierung einer neuen Konvertierungsregel

In Listing 6.8 ist ein Beispiel von der Speicherung einer Konvertierungsregel gegeben. Sowohl Quell- als auch Zielattribute müssen für die Datenbank bekannt sein. Zudem werden die Parameter Name, Typ und Parameter angegeben.

```
1  def post(self):
2          source_attributes = [a for a in self.db_conn.query(Attribute).filter(
       Attribute.id.in_(self.body['source']))]
3          target_attribute = self.db_conn.query(Attribute).filter(Attribute.id
       == self.body['target']).one()
4          new_rule = ConversionRule(
5                  name=self.body['name'],
6                  source=source_attributes,
7                  target_id=target_attribute.id,
8                  type=self.body['type'],
9                  parameter=bytearray(self.body['parameter'], "UTF-8")
10             )
11         try:
12             self.db_conn.add(new_rule)
13             self.db_conn.commit()
14             self.set_status(201)  # Created
15             self.add_header("Location", "api/conversionrule/{}".format(
       new_rule.id))
16         except exc.IntegrityError as err:
17             self.db_conn.rollback()
18             self.error("Could not add conversion rule.", str(err.orig), 409)
```

Listing 6.8: Speicherung einer neuen Konvertierungsregel in der Datenbank

Für eine Umbenennung mit Shibboleth wird zunächst das JSON File

`convRuleMgmt/test/inputs/rename-test.json`

geöffnet, das folgendes Format hat (vgl. Listing 6.9).

```
1  {
2    "source": ["source1"],
3    "transformation": {
4      "action": "rename"
5    },
6    "target": {
7      "name": "fooTarget",
8      "urn1": "targetUrn1",
9      "urn2": "targetUrn2"
10   }
11 }
```

Listing 6.9: Format einer Umbenennung in JSON

Jede Implementierung hat unterschiedliche Templates, die im Fall der prototypischen Implementierung der Einfachheit halber bei der TTP gespeichert sind. Je nach Implementierung wird die Variable `self.env`, die den Ort der Templates angibt, gesetzt. Über `self.rename_renderer = render.Render(shibboleth, file_in)` wird die Datei erstellt. Im Beispiel der SAML-Implementierung Shibboleth werden die Werte aus der Datenbank in das Template für Shibboleth eingetragen und als Datei versendet. Das Template sieht wie folgt aus (vgl. Listing 6.10).

```
<resolver:AttributeDefinition xsi:type="Script" xmlns="urn:mace:shibboleth
    :2.0:resolver:ad" id="{{target}}">
    <resolver:Dependency ref="{{source1}}"/>
    <resolver:Dependency ref="{{source2}}"/>
    <resolver:AttributeEncoder xsi:type="SAML1String"
                           xmlns="urn:mace:shibboleth:2.0:attribute:encoder"
                           name="{{targetUrn1}}" />
    <resolver:AttributeEncoder xsi:type="SAML2String"
                           xmlns="urn:mace:shibboleth:2.0:attribute:encoder"
                           name="{{targetUrn2}}"
                           friendlyName="{{target}}" />
    <Script>
        <![CDATA[
            importPackage(Packages.edu.internet2.middleware.shibboleth.
                           common.attribute.provider);
            target = new BasicAttribute("{{target}}");
            merge = {{source1}}.getValues().get(0) + " " + {{source2}}.
                           getValues().get(0);
            target.getValues().add(merge);
        ]]>
    </Script>
</resolver:AttributeDefinition>
```

Listing 6.10: Format einer Umbenennung in XML für Shibboleth

6.1.6. Oberflächenbausteine

Die vorher beschriebenen Bausteine sind grundlegend für die Managementplattform. Die Oberflächengestaltung, insbesondere die Visualisierung der Managementanwendungen, ist für die Administratoren relevant. Die oben genannten Basisanwendungen, Informationsbaustein und Kommunikationsbaustein sowie die Managementanwendungen werden über die Oberfläche visuell dargestellt und somit als Funktion dem Kunden bereitgestellt.

Die Oberflächenbausteine gruppieren sich in zwei Anwendungsbereiche: Nutzersicht mit den Lokalisierungsdiensten und die Oberflächenbausteine der Managementplattform MdFIM. Während die Lokalisierungsdienste keine Änderungen der Oberflächengestaltung erhielten, wurde die Managementplattform MdFIM implementiert und gestaltet.

Oberflächenbausteine der Lokalisierungsdienste

Der Nutzer stößt den dynamischen Metadatenaustausch durch die Wahl seines IdPs an [FA-Initiierung]. Die Initiierung geschieht bei der Lokalisierung durch die Erweiterungen der Lokalisierungsdienste [FA-Lokalisierung].

Um diese Initiierung zu starten, wurde beim *Embedded Discovery Service* eine Weiterleitung zur TTP mit dem zentralen Lokalisierungsdienst eingerichtet. Der Embedded Disco-

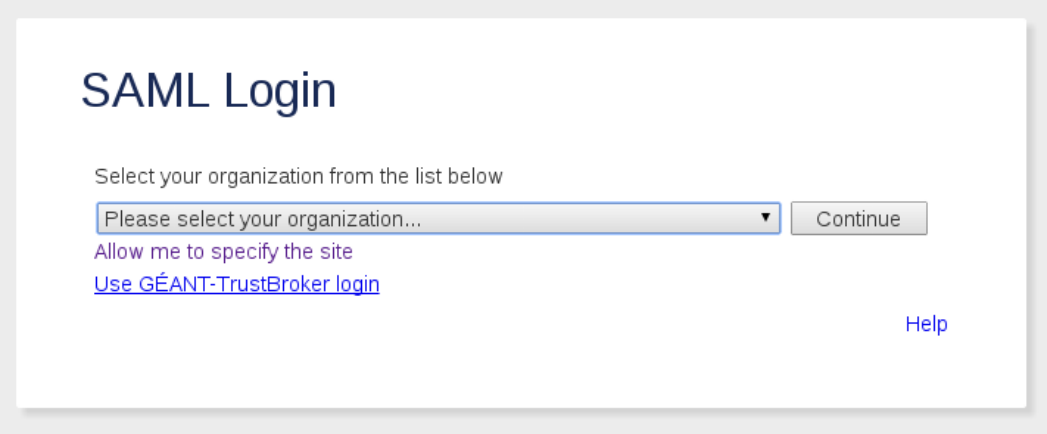

Abbildung 6.5.: Embedded Discovery Service

very Service läuft beim Service Provider und listet alle IdPs auf, denen der SP vertraut. Die Weiterleitung für den dynamischen Metadatenaustausch, die somit eine Erweiterung der Vertrauensliste hervorrufen kann, ist in Abbildung 6.5 dargestellt.

Beim *Centralized Discovery Service* der TTP werden wiederum all diejenigen Identity Provider gezeigt, die bei der TTP registriert sind. Das ist in der Abbildung 6.6 zu sehen. Beim Metadatenaustausch kann zudem eine Information über den gerade durchgeführten Metadatenaustausch erscheinen. Diese Benutzerinformation wurde jedoch im Rahmen der prototypischen Implementierung nicht realisiert.

Oberflächenbausteine der MdFIM

Die Managementplattform MdFIM realisiert die Verwaltung der Entitäten, Benutzer, Metadaten und Konvertierungsregeln in der prototypischen Implementierung. Die Implementierung baut auf dem Centralized Discovery Service, der in Java und JSP implementiert wurde, auf und basiert auf JSP mit dem Framework Bootstrap. Nach der Anmeldung werden über den in Listing 6.11 beispielhaft gezeigten Code die Berechtigungen des Nutzers in Hinblick auf die Entität abgefragt. Basierend auf den Rollen und Berechtigungen werden in der darauf folgenden Übersichtsseite unterschiedliche Informationen dargestellt.

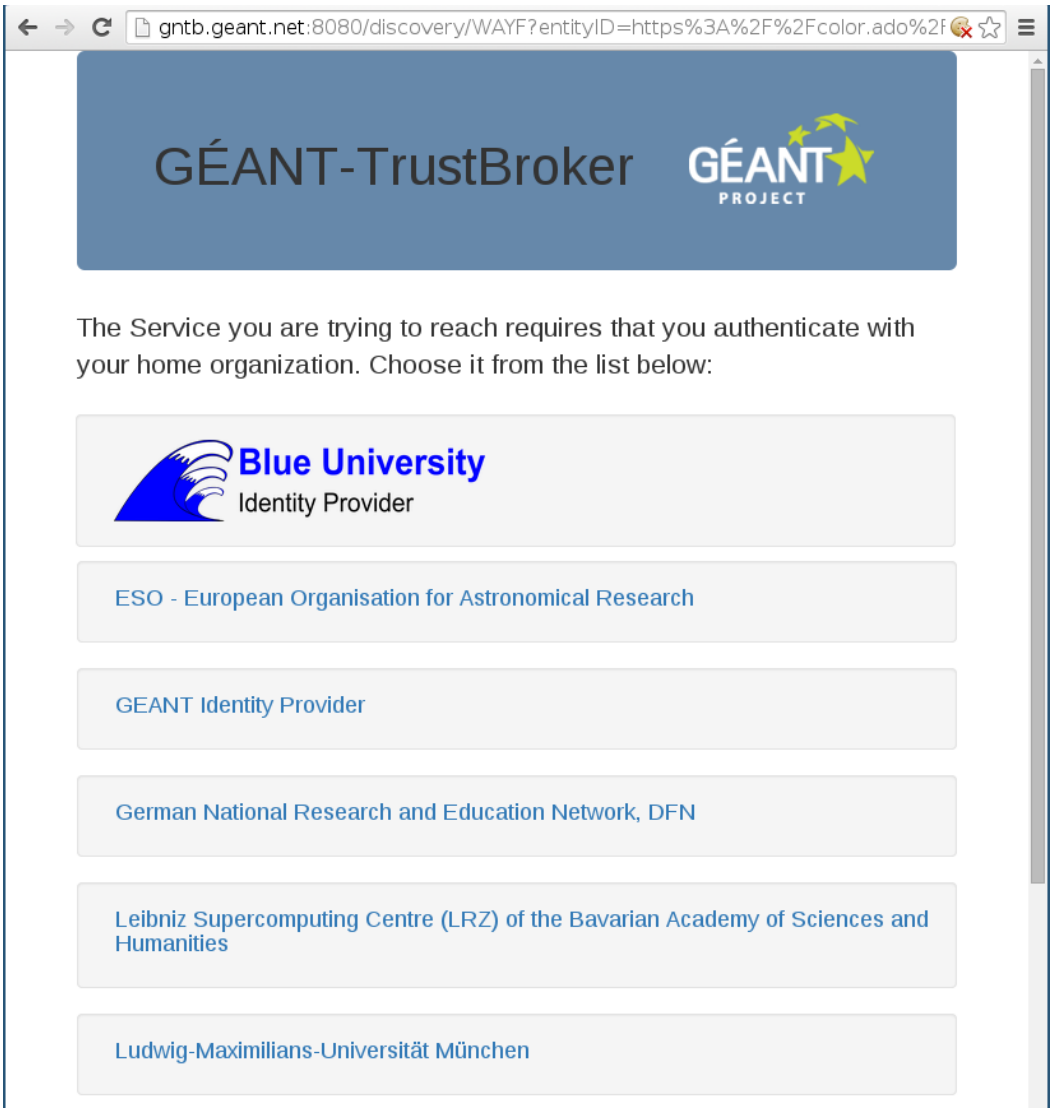

Abbildung 6.6.: Centralized Discovery Service der TTP

```
1 <%
2     List<TTPProvider> providers = db.getProvidersForUserID(user.getId());
3     for (TTPProvider entry : providers) {
4         List<TTPProviderRole> roles = db.getProviderRolesForUserID(
5         user.getId(), entry.getId());
6             ...
7 %>
```

Listing 6.11: Abfrage der Berechtigungen für den Benutzer

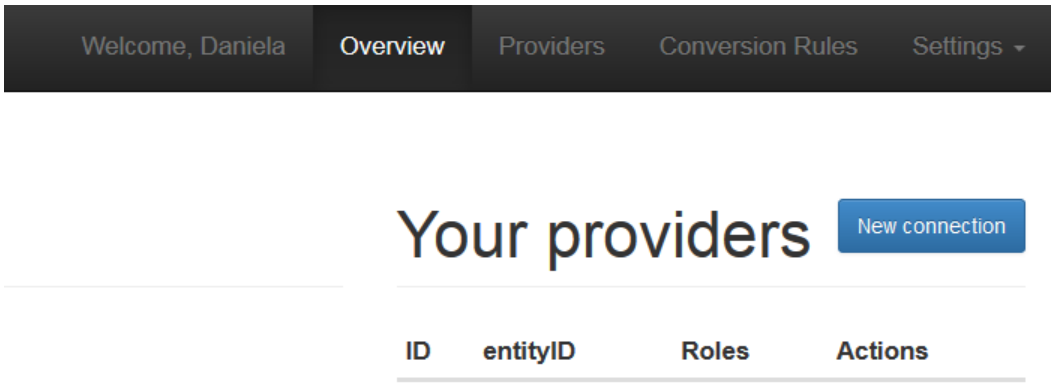

Abbildung 6.7.: Übersichtsseite der Managementplattform

Dies zeigt die Abbildung 6.7. Nachdem der Nutzer `dpoehn` keine mit ihm verknüpften Entitäten besitzt, kann er dies im nächsten Schritt nachholen. Zunächst wird eine neue Entität, Provider genannt, angelegt. Hierfür sind EntityID, Typ, Organisation und eine Beschreibung relevant. Metadaten können über die Webanwendung ebenso hochgeladen werden. Beim Hochladen werden die Metadaten verifiziert, beispielsweise bezüglich der Wohlgeformtheit und ob die EntityIDs identisch sind.

Im nächsten Schritt muss der Benutzer seine Berechtigungen für die Entität beweisen, indem er eine Datei auf dem Webserver mit einem zufällig generierten Namen anlegt. Der Name der anzulegenden Datei ist in Abbildung 6.8 zu sehen. In JSP wird dies über den in Listing 6.12 beschriebenen Aufruf realisiert. Zukünftig können auch Verifizierungen über E-Mail-Adressen und Zertifikate eingesetzt werden.

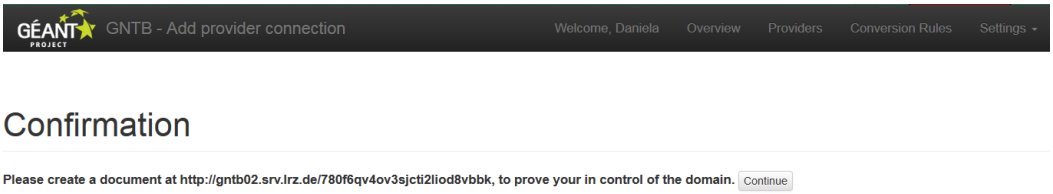

Abbildung 6.8.: Verifikation der Berechtigung für eine bestimmte Entität

```
1  <h1 class="page-header">Confirmation</h1>
2  <p>
3  <form role="form" action="./providerConnectionManagement"
4          method="post">
5          <label>Please create a document at <%=Encode.forHtml(checkURL)%>,
6                  to prove your in control of the domain.
7          </label> <input type="hidden" name="action" value="validate"></input>
8          <button class="btn btn-xs btn-default">Continue</button>
9  </form>
10 </p>
```

Listing 6.12: Abfrage der Berechtigungen des Nutzers für eine Entität

Über den Reiter `Providers` kann der Administrator verschiedene Entitäten mit ihren Informationen und Metadaten sehen. Vertrauensinformationen werden ebenso auf dieser Seite dargestellt. Eine weitere Funktionalität ist das Herunterladen der Metadaten. Wenn der Benutzer der Managementplattform die Berechtigung hat, die Entität zu ändern, kann dies ebenso auf dieser Seite geschehen.

In einem weiteren Reiter der Webanwendung ist die Verwaltung der Konvertierungsregeln möglich. Abbildung 6.9 zeigt einen Überblick über alle auf MdFIM gespeicherten Konvertierungsregeln. Diese werden dynamisch aus der Datenbank abgerufen und angezeigt. Abhängig von der Berechtigung des Nutzers können Konvertierungsregeln auch geändert werden.
Äquivalent zum Vorgehen bei Entitäten kann eine einzelne Konvertierungsregel bzw. die Metainformationen dieser angeschaut werden. Entscheidende Informationen für eine einzelne Konvertierungsregel sind `target` und `source`. Die Art der Umwandlung steht in diesem Fall in der `description`. Durch diese Informationen kann der Benutzer entscheiden, ob er die Konvertierungsregel verwenden kann oder nicht. Herunterladen ist hier ebenfalls möglich. Falls der Benutzer die entsprechenden Berechtigungen hat, kann er die Konvertierungsregel auf dieser Seite ebenfalls ändern.

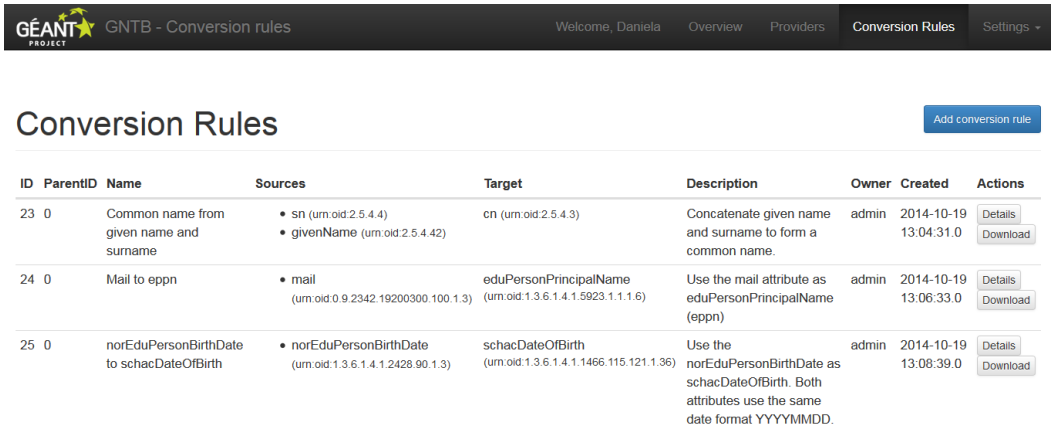

Abbildung 6.9.: Übersichtsseite der Konvertierungsregeln

Findet der Benutzer keine passende Regel, kann er eine neue Regel erstellen. Dies ist in der prototypischen Implementierung durch die Parameter `Name`, `Target`, `Sources`, `Description` und `File` realisiert. Die Webanwendung kennt durch gespeicherte Attribute und ihre URN bereits wichtige Elemente von Konvertierungsregeln. Zusätzliche Attribute können ebenso hinzugefügt werden. Über `File` lädt der Administrator die erstelle Konvertierungsregel hoch. Vor dem Speichern wird diese verifiziert. Wenn die Verifizierung erfolgreich war, kann die Konvertierungsregel durch andere Benutzer wiederverwendet werden. Hierdurch werden unterschiedliche Schemata realisiert [FA-Schema].

6.2. Untersuchung der Skalierbarkeit

Da die Skalierbarkeit eine essentielle Anforderung ist, werden die benötigten Schritte sowie die Größe des integrierten Metadatensatzes zunächst betrachtet. Die Ergebnisse der Untersuchung basieren auf [GHMP15].

Die technischen Daten der acht virtuellen Maschinen der Testumgebung sind die Folgenden:

- ESXi 5.5 und höher (VM-Version 10),
- 2048 Megabyte (MB) Arbeitsspeicher,
- 11 Gigabyte (GB) Festplatte,
- Domain Name System (DNS) Name `gntb01` bis `gntb08`,
- IP-Adressen `129.187.163.161` bis `129.187.163.168`,

- CPU Intel Nehalem bzw. Ivy Bridge und
- ProLiant BL460c Gen8 als Blades mit je zwei CPUs Intel(R) Xeon(R) CPU E5-2660 v2 2.20GHz (10 Cores).

Auf diesen VMs lief zunächst die stabile Version 7.6 *Wheezy* der Linux-Distribution Debian, bevor auf 8.x *Jessie* aktualisiert wurde. Um verschiedene Szenarien durchspielen zu können, werden die Maschinen aufgeteilt, so dass die Testumgebung neben einer TTP mehrere IdPs und SPs besitzt. Um die Kompatibilität zu prüfen und um reale Bedingungen zu simulieren, werden verschiedene Software-Versionen auf den Maschinen betrieben:

- Shibboleth Identity Provider: Versionen 2.4.0, 2.4.2 und 3.1.
- Shibboleth Service Provider: Versionen 2.4.3 und 2.5.3.

Zudem werden die Tomcat-Versionen 6 und 7 eingesetzt. Die Performanz ist eine wichtige Anforderung, die im Kapitel 2 definiert wurde. Nachdem sich im Vergleich zu normalem FIM dynamisch Metadaten ausgetauscht und zudem die Vertrauenswerte überprüft werden, erhöht sich folglich die Verarbeitungszeit. Nachdem jedoch mit dieser Testumgebung keine Untersuchung der Performanz unter Last ohne größere Veränderungen am Code möglich ist, wurde auf eine Performanzuntersuchung verzichtet. Innerhalb der gegebenen Umgebung wurden jedoch keine Unterschiede zum statischen FIM festgestellt.

Um die Skalierbarkeit und notwendigen Schritte, zu überprüfen, werden im Folgenden drei Szenarien analysiert:

- Föderation,
- Inter-Föderation und
- Virtuelle Föderation.

Die Testumgebung ist wie in Abbildung 6.10 aufgesetzt [GHMP15]. Sie besteht aus zwei Föderationen, *RAINbow* und *Federation Blue*, sowie mehreren Entitäten:

- IdP Orange University, IdP Violet University und SP Green Hopper sind Mitglieder der Föderation *RAINbow*.
- IdP Blue University und SP Aqua Service der Föderation *Blue*.
- SP Grey Services und IdP Yellow University, die keiner Föderation angehören.

Nachdem Kooperationen nicht nur innerhalb von festen Föderationen entstehen, können zudem virtuelle Föderationen über die TTP gebildet werden. Hierfür müssen alle beteiligten IdPs die Metadaten der TTP hinzufügen und die Erweiterung herunter laden sowie installie-

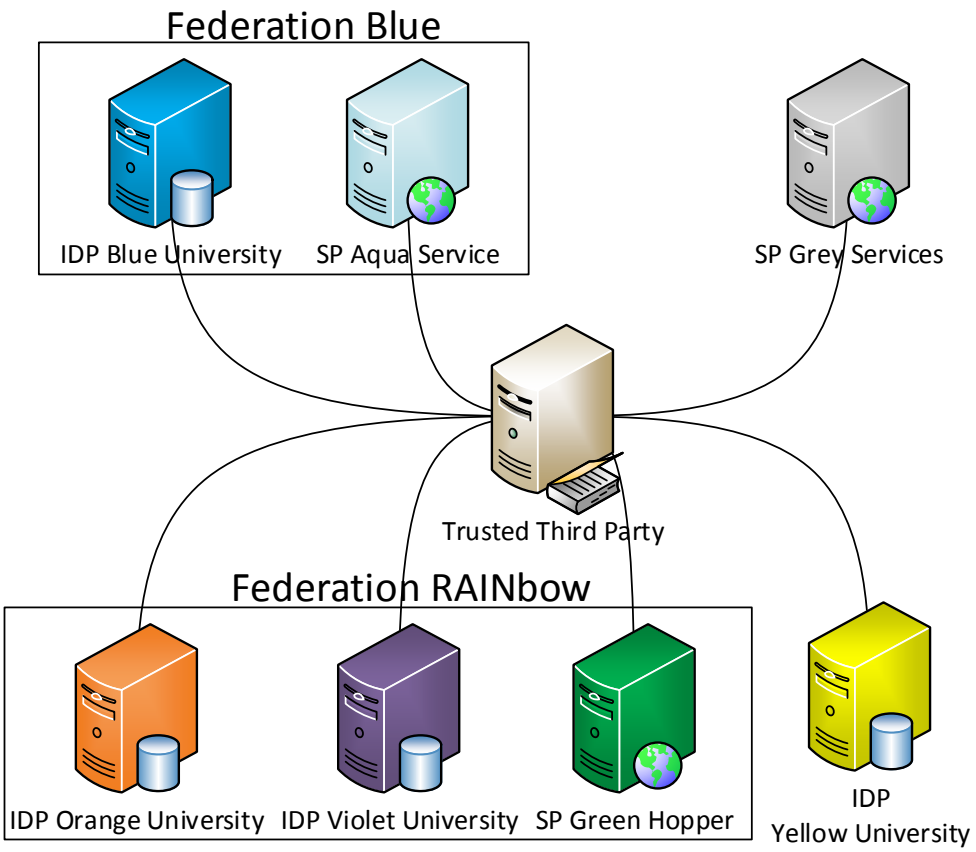

Abbildung 6.10.: Überblick über die Testumgebung [GHMP15]

ren und konfigurieren. Die Metadaten der TTP sind dadurch relevant, da die TTP dem IdP gegenüber als SP erscheint. SPs müssen den MetadataProvider für die TTP konfigurieren. Im Gegensatz dazu werden bei herkömmlichen Föderationen vorab aggregierte Metadaten ausgetauscht. Anstelle von virtuellen Föderationen müssen Inter-Föderationen oder neue Föderationen gegründet werden, bei denen ebenfalls aggregierte Metadaten ausgetauscht werden. Diese Unterschiede werden bei den nachfolgenden Szenarien hervorgehoben.

Hierbei wird die Konfiguration der Erweiterungen nicht mit einbezogen. Die Szenarien finden in der kleinen Testumgebung statt, jedoch können bereits hier Unterschiede zwischen statischem und dynamischem FIM festgestellt werden.

6.2.1. Szenarien und Vorgehensweise

Innerhalb einer Föderation kann die TTP verwendet werden, um Metadaten zwischen einzelnen Mitgliedern auszutauschen. Dieser Ansatz verbessert die Skalierbarkeit in großen Föderationen, nachdem nur ein kleiner Teil der möglichen Vertrauensverbindungen überhaupt verwendet wird. Dieses Szenario wird am Beispiel der Föderation *Blue* in der Testumgebung überprüft.

Der Metadatensatz einer Inter-Föderation, wie beispielsweise eduGAIN, enthält noch mehr Entitäten als der Satz einer Föderation, nachdem die Metadaten der Mitglieder mehrerer Föderationen aggregiert werden. In diesem Szenario bilden die Föderationen *Blue* und *RAINbow* die Inter-Föderation *BlueRAINbow*. Um effizient die Inter-Föderation aufzusetzen, sollen die Administratoren der Föderationen ihre Metadatensätze in die TTP importieren und die Metadaten der TTP in ihren eigenen Satz einfügen. Dies ist unabhängig davon, ob die TTP oder ein Metadata Aggregator eingesetzt wird.

Zusätzlich zu den bekannten Strukturen Föderation und Inter-Föderation soll die Bildung einer virtuellen Föderation überprüft werden. Das dynamische Verhalten wird hierbei nicht analysiert, sondern neben der Zeit für den Metadatenaustausch vor allem die benötigten Schritte. Um eine virtuelle Föderation zu bilden, schließen sich die Entitäten

- SP Grey Services,
- IdP Yellow University und
- IdP Blue University

zusammen.

6.2.2. Ergebnisse zur Skalierbarkeit

In der Föderation ohne TTP müssen n Entitäten sich zunächst bei der Föderation registrieren. Anschließend müssen die Föderationsmetadaten in die eigene Konfiguration eingefügt werden, was n weitere Schritte verursacht. Dies ergibt insgesamt $2n$ Schritte ohne die Verwendung einer TTP.
Beim Einsatz einer TTP ergeben sich ebenfalls n Schritte für die Registrierung. Jedoch müssen nur IdPs die Metadaten der TTP registrieren. Wenn innerhalb der Föderation n_{idp} Identity Provider sind, ergeben sich daraus $n + n_{idp}$ Schritte, wobei n_{idp} kleiner ist als n.

Im Fall der Föderation *Blue* ergibt sich eine Anzahl von je einem integrierten Metadatensatz der gegenüberliegenden Entität plus die Metadaten der TTP auf Seiten des IdPs. Bei einer Anzahl von n Entitäten werden bei SPs maximal n_{idp} und bei IdPs maximal $1 + n_{sp}$ Metadaten integriert. Folglich ist sogar bei einer sehr kleinen Föderation von 2 Entitäten die Größe des Metadatensatzes kleiner als bei herkömmlichen Föderationen.

Beim Szenario Föderation ergeben sich folgende Werte:

- Schritte: 3 (4)
- Anzahl: 1 bzw. 2 (4)

Wenn die Föderationen *Blue* und *RAINbow* die Inter-Föderation *BlueRAINbow* gründen, ergeben sich ohne TTP $2i$ Schritte bei i Föderationen.
Ohne Bulk-Import müssen sich $\sum_{x=1}^{i} n_x$ Entitäten bei der TTP registrieren und $\sum_{x=1}^{i} n_{idp_x}$ IdPs die Metadaten der TTP integrieren. Daraus ergeben sich für das Szenario 8 Schritte.

Ein aggregierter Metadatensatz umfasst im Szenario Inter-Föderation $n = 8$ Metadaten. Mit der TTP umfasst der Metadatensatz für IdPs maximal 3 Metadaten und für IdPs maximal ebenfalls 3 Metadaten, da auch die Metadaten der TTP integriert werden müssen.

- Schritte: 8 (8)
- Anzahl: Maximal 3 (5)

Ohne einer TTP müssen in diesem Szenario alle Entitäten die Metadaten der anderen integrieren, was zu $2n(n-1)$ Schritten führt.
Bei einer Verwendung der TTP reduziert sich die Anzahl auf $n + n_{idp}$.

Die Größe des integrierten Metadatensatzes beträgt bei dem SP maximal 2, während die IdPs jeweils 2 Metadaten benötigen. In der Übersicht sieht das wie folgt aus:

- Schritte: 5 (12)
- Anzahl: Maximal 2 bzw. 2 (3)

Tabelle 6.1.: Vergleich der Schritte ohne und mit TTP [GHMP15]

	Manuell	TTP
Föderation	$2n$	$n + n_{idp}$
Inter-Föderation	$2i$	$\sum_{x=1}^{i} n_x + \sum_{x=1}^{i} n_{idp_x}$
Virtuelle Föderation	$2n(n-1)$	$n + n_{idp}$

Tabelle 6.2.: Vergleich der integrierten Metadatensätze ohne und mit TTP

	Manuell	TTP
Föderation	n	maximal n_{idp} bzw $1 + n_{sp}$
Inter-Föderation	n	maximal n_{idp} bzw $1 + n_{sp}$
Virtuelle Föderation	n	maximal n_{idp} bzw $1 + n_{sp}$

Nachdem mit der TTP durch den dynamischen Metadatenaustausch feste Strukturen beseitigt und virtuelle Föderationen ermöglicht werden, sind alle möglichen Kombinationen von Szenarien möglich, die oben präsentiert wurden. Die beschriebenen Szenarien sind bezüglich der Schritte und Anzahl der integrierten Metadaten mindestens genauso gut, wenn nicht besser als statisches FIM.
Die Tabelle 6.1 zeigt die benötigten Schritte auf, um sowohl manuell als auch dynamisch Metadaten auszutauschen. Dabei sind:

- n: Anzahl der Entitäten
- n_{idp}: Anzahl der IdPs
- i: Anzahl der Föderationen

Dies zeigt, dass bis auf dem Szenario der Inter-Föderation mit dynamischen Metadatenaustausch weniger Schritte nötig sind. Dies ist selbst bei einer kleinen Anzahl an Entitäten der Fall. Wenn die prototypische Implementierung um einen Bulk-Import erweitert wird, verbessert sich auch dieser Wert. Somit ist die technische Lösung TTP genauso gut beziehungsweise bei mehr Entitäten besser bezüglich der nötigen Operationen wie aktuelle Föderationen und Inter-Föderationen.

Bezüglich der Anzahl integrierter Metadaten gibt es keinen Unterschied zwischen Föderationen, Inter-Föderationen und virtuellen Föderationen. Bei Verwendung des dynamischen Metadatenaustausches mit der TTP existiert jedoch eine Differenz zwischen IdPs und SPs. Nachdem IdPs auch die Metadaten der TTP integrieren müssen, erhöht sich die maximale Anzahl der integrierten Metadaten um 1. Da Identity Provider neben den Metadaten der TTP nur die Metadaten von maximal allen SPs integrieren und umgekehrt, werden weniger Metadaten insgesamt ausgetauscht und integriert wie mit klassischem FIM. Dies ist insbesondere bei einer großen Anzahl an Teilnehmer entscheidend.

Dies zeigt, dass die Skalierbarkeit bei dynamischen Metadatenaustausch mit einer TTP

besser ist als bei FIM mit aggregierten Metadatensätzen.

6.3. Zusammenfassung und Aspekte des praktischen Einsatzes

In diesem Kapitel wurde die Implementierung der Managementplattform MdFIM mit den Basisanwendungen Registrierung und Einloggen und dem Metadaten Management dargelegt. Der für diese Arbeit grundlegende dynamische Metadatenaustausch wurde ebenso dokumentiert wie die Kommunikation zwischen den Entitäten und MdFIM. Zudem wurde die grundlegende Implementierung der beiden Werkzeuge Trust Management und Conversion Rule Management vorgestellt. Die Implementierung dieser Bausteine hat gezeigt, dass für einen vollständigen Funktionsumfang tiefgreifende Änderungen in die Komponenten von Shibboleth notwendig werden, so dass dieser zugunsten einer realisierbaren Implementierung reduziert wurde.

Die Implementierungen stellen sowohl Kernideen dieser Arbeit als auch einen praktischen Mehrwert für Föderationen und Entitäten dar. Die Szenarien haben bewiesen, dass die Skalierbarkeit bezüglich der Größe der integrierten Metadatensätze und benötigten Schritte verbessert wurde.

Die an den Shibboleth Komponenten durchgeführten Änderungen wurden öffentlich gemacht und mit einem breiten Publikum sowie den Entwicklern von Shibboleth diskutiert. Obwohl die Idee von dynamischen Metadatenaustausch partiell positiv aufgenommen wurde, besteht aktuell keine Möglichkeit, die Erweiterungen in den bereits in Arbeit befindlichen nächsten Versionen zu berücksichtigen. Durch die Dauer der bereits geplanten Releasezyklen ist eine Integration bei darauf folgenden Versionen grundsätzlich möglich. Hierfür soll Kontakt zum Shibboleth-Konsortium aufgenommen werden. Parallel dazu wird versucht Verbesserungen durch MdFIM über das Projekt GÉANT möglichst vielen Entitäten und Föderationen bereitzustellen.

Bezüglich des praktischen Einsatzes muss jedoch die Schnittstelle der Föderationsverwaltung integriert werden, um die Qualität der Konvertierungsregeln zu gewährleisten. Die Schnittstelle wird zudem zur Erstellung, Überprüfung und Pflege der Konvertierungsregeln benötigt. In Hinblick auf die zunehmende Verbreitung von ADFS soll zudem eine Möglichkeit zur automatisierten Einbindung von Konvertierungsregeln gefunden werden. Die zusätzliche Speicherung der Implementierungs-spezifischen Konvertierungsregeln bei den IdPs kann hinsichtlich der Performanz überprüft werden. Hierfür müssen die Schlüsselwörter jedoch bei dem jeweiligen IdP eingefügt werden. Zudem muss eine Möglichkeit geschaffen werden auch mit Hilfe einer kleinen Testumgebung die Performanz zu testen. Hierbei sind jedoch größere Änderungen notwendig. Um die Verwaltung der Workflows bei der MdFIM zu optimieren, soll, wie bei Shibboleth Identity Provider Version 3, das Spring Framework und damit eine Workflow Engine eingesetzt werden.

Kapitel 7

Prototypische Anwendung

Inhalt dieses Kapitels

In diesem Kapitel werden die erarbeitete Architektur und ihre Werkzeuge anhand eines konkreten, realistischen Beispiels angewandt. Anstelle einzelner Szenarien, wie im Kapitel 2, wird hier ein komplexes Beispiel verwendet, welches mehrere Szenarien abdeckt. Das *Bayerische Archiv für Sprachsignale (BAS)* ist ein SP bei CLARIN. Das Archiv ist der LMU zugeordnet, deren IdP durch das LRZ betrieben wird. Folglich wird davon ausgegangen, dass der IdP der Mitarbeiter von BAS zur LMU gehört. Neben der Föderation DFN-AAI und der Inter-Föderation eduGAIN ist der SP Teilnehmer in den Föderationen ACOnet (Österreich), Belnet (Belgien), eduID.cz (Tschechien), Haka (Finnland), SWAMID (Schweden) und UK Federation. Nachdem CLARIN mit weiteren Föderationen Verträge aushandelt, wandelt sich die Struktur. Dies bietet in dieser Arbeit die Möglichkeit die Gesamtarchitektur anhand eines Beispiels zu demonstrieren. Zudem werden die Möglichkeiten, die bisher mit FIM möglich waren, mit den neu eingeführten und zum Teil implementierten FIM-Komponenten gegenüber gestellt. Die Einführung von MdFIM basiert auf der Einführung von FIM [Hom07].

Zunächst werden *Planungsaspekte und Vorbedingungen* in Abschnitt 7.1 bestimmt, wobei organisatorische Regelungen und technische Vorbereitungen betrachtet werden. Basierend auf der Referenzarchitektur wird in Abschnitt 7.2 die *Zielarchitektur* spezifiziert. Diese wird

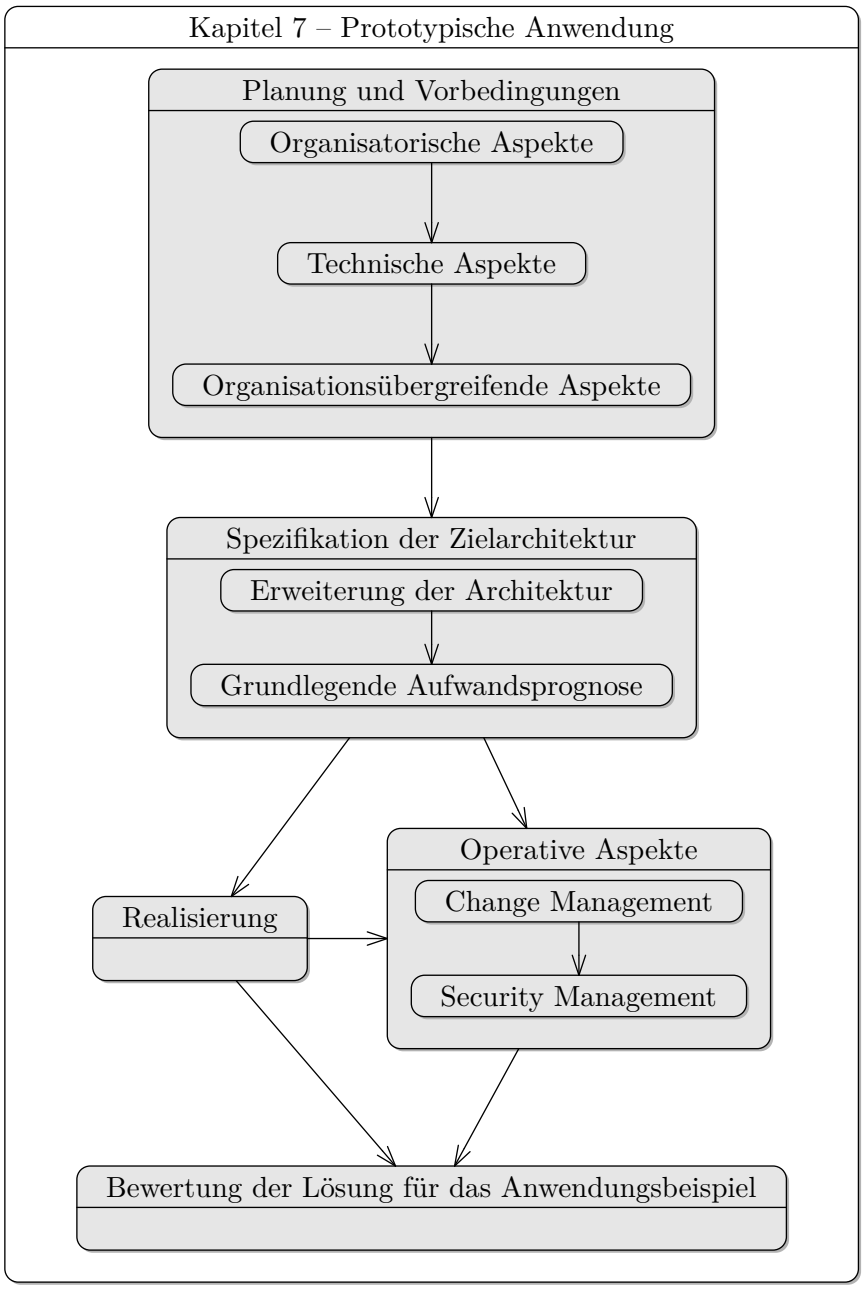

Abbildung 7.1.: Vorgehensmodell in diesem Kapitel

im nächsten Schritt realisiert. Bei der *Realisierung* stellt die Migration einen Schwerpunkt dar, die in Abschnitt 7.3 aufgezeigt wird. Die Untersuchung verschiedener *operativer Aspekte* in Abschnitt 7.4 betrachtet die Konfiguration und Veränderungen im Change und Security Management. Die Lösung wird außerdem für das Anwendungsbeispiel bewertet. Diese Vorgehensweise ist auch in Abbildung 7.1 dargestellt.

7.1. Planungsaspekte und Vorbedingungen

Für die Umsetzung des in Kapitel 4 erläuterten Konzepts auf das BAS mit seinem Service Provider für CLARIN und einem Identity Provider der LMU am LRZ müssen bestimmte *Randbedingungen* und *Ziele* betrachtet werden:

- Die organisatorischen und technischen Voraussetzungen für den Einsatz von dynamischen FIM am BAS und am LRZ bzw. an der LMU sowie für weitere beteiligte Organisationen sind zu untersuchen.
- Die Erweiterungen der bestehenden FIM-Komponenten und deren nahtlose Integration sind zu spezifizieren. Dabei ist die Integration in die vorhandene IT-Sicherheitsinfrastruktur zu berücksichtigen.
- Die resultierende Architektur ist im Hinblick auf entstehende Kosten zu bewerten. Hierbei sind Realisierungsaufwand, Investitions- und Betriebskosten sowohl für LMU bzw. LRZ als auch für das BAS entscheidend.
- Eine vorgegebene Migrations- und Integrationsmethodik ist zu berücksichtigen.
- Die Selektion der verwendeten Komponenten und Werkzeuge mündet in der Skizzierung ihrer Konfiguration.
- Die Auswirkungen auf das IT Service Management sind darzustellen. Hierbei wird wie in Kapitel 4 eine Einschränkung auf Security Management und Change Management am LRZ durchgeführt.

Zwar sind im Gegensatz zur Umstellung von I&AM zu FIM weniger neue Komponenten einzubinden, jedoch muss die *Automatisierung* möglichst sicher sein. Das deckt sich mit der Risikoabschätzung des eigenen Dienstes sowie der entsprechenden Konfiguration. Diese Phase hat insbesondere Auswirkungen auf das *Change Management* im Vergleich zum späteren Betrieb und nachfolgenden Anpassungen. Um alle Planungsaspekte zu berücksichtigen, wird nachfolgend die Planung zur Migration aufgeteilt in *organisatorische*, *technische* und *organisationsübergreifende Aspekte* betrachtet.

7.1.1. Organisatorische Aspekte

Nach der Entscheidung zur Migration auf dynamisches FIM ist innerhalb des BAS und LMU/LRZ eine Projektgruppe zu gründen, die aus den folgenden Bereichen gebildet wird:

- Am LRZ wird bereits FIM in Form der SAML-Implementierung Shibboleth von den für FIM zuständigen Mitarbeitern eingesetzt. Diese Mitarbeiter haben das technische Wissen über die bisherige Implementierung und daher bilden sie den Kern der Pro-

jektgruppe.

- Vertreter des BAS und weiterer über FIM angebotenen Dienste bringen ihr Wissen in die Projektgruppe. Gleichzeitig stellen sie spezifische Anforderungen. Die Projektgruppe kann zudem genutzt werden, um eine Inventarisierung aller möglicher FIM-Dienste zu machen. Nachdem viele Dienste noch nicht über FIM angebunden sind, kann dieser Schritt nach der Migration der bereits angebundenen Dienste nachgeholt werden.

- Ein oder mehrere Spezialisten im Bereich System- und Netzwerksicherheit begleiten die grundlegende Planung und das Design. Ferner sind die Spezialisten für die Umsetzung von Schutzmaßnahmen bei der Inbetriebnahme zuständig.

- Ein Mitarbeiter ist ferner für die Koordination mit dem LRZ- und LMU-weiten IT Service Management zuständig. Dazu gehören auch die Planung und Dokumentation der Prozesse.

Die Projektgruppe ist mit den ersten organisatorischen Aufgaben betraut, die wie folgt lauten:

- Zum einen müssen die bisher nicht mit FIM betrauten Mitarbeiter eine Schulung über FIM und dynamisches FIM erhalten. Dies gilt insbesondere für die Vertreter von Diensten, die in Zukunft über FIM angebunden werden sollen. Zum anderen sollen Mitarbeitern, die sich bereits mit FIM beschäftigt haben, die Neuerungen durch dynamisches FIM näher gebracht werden. Dies kann in einer Zweiteilung der zweiten Schulung realisiert werden, wo die allgemeinen Änderungen für alle Teilnehmer der Projektgruppe präsentiert werden, während ein zweiter Teil der Schulung auf die technischen Details eingeht, die vor allem für den Kern der Projektgruppe relevant sind.

- Die Projektziele müssen priorisiert werden. Hierbei kann eine Reihenfolge der zu migrierenden Dienste entstehen. Bei der Anbindung über FIM muss bei bisher nicht angebundenen Diensten zudem überprüft werden, ob Anpassungen benötigt werden.

- Eine weitere Aufgabe ist die Koordination mit Externen, d. h. anderen Hochschulen und Organisationen, wie Föderationen und der Community CLARIN. Diese Koordination soll von Mitarbeitern der Projektgruppe gemacht werden, die fachlich am nächsten sind. Für CLARIN bedeutet das, dass ein Vertreter vom BAS die Koordination mit anderen Vertretern von CLARIN übernehmen soll. Bei der Koordination mit der Föderation DFN-AAI wird ein Mitarbeiter der mit FIM betrauten Aufgaben gewählt.

- Mit dem lokalen Datenschutzbeauftragten sollte möglichst früh Kontakt aufgenommen werden, um Verfahrensbeschreibungen vorzubereiten. Nachdem durch das dynamische FIM automatisiert Metadaten und Informationen über Nutzer ausgetauscht werden, müssen Unklarheiten möglichst frühzeitig beseitigt und eine entsprechende Konfiguration der Software vorgenommen werden. Dies ist im Beispiel BAS für die LMU vorzunehmen.

- Grundlegende Finanzierungsaspekte sind zu klären. Dies betrifft in erster Linie den Personalaufwand für die Migration. Während die genauen Kosten für die Migration von der Zielarchitektur und von der Anzahl der anzubindenden Dienste abhängt, kann die Größenordnung der Investitionskosten bezogen auf den Personalaufwand der Migration und der Hardware- und Lizenzkosten bereits in dieser Phase ermittelt werden. Dies wird in Abschnitt 7.2 genauer erläutert.

- Das Change Management ist für die Migration vorzubereiten. Zuständige Mitarbeiter für das CAB sind zu ernennen und die neuen Use Cases sind zu beschreiben und zu dokumentieren.

- Die Änderungen durch dynamisches FIM und die damit modifizierte Software sind bezüglich des Security Managements zu betrachten. Nachdem das Trust Management ein neuer Bestandteil des FIMs ist, müssen sowohl der eigene Trust-Level als auch die Mindestanforderungen an SPs bzw. IdPs überdacht werden.

- Um bei Lokalisierungsdienst der MdFIM angezeigt zu werden, müssen sich teilnehmende IdPs und SPs dort registrieren.

Auf weitere Aspekte, die nicht FIM-spezifisch sind, wird an dieser Stelle nicht weiter eingegangen.

7.1.2. Technische Aspekte

Neben den organisatorischen Aspekten müssen intern auch die technischen Aspekte betrachtet werden. Nachdem bereits FIM eingesetzt wird, muss die Erweiterung installiert und konfiguriert werden. Für den Identity Provider sieht das wie folgt aus:

- Installation und allgemeine Konfiguration der Erweiterungen: Die Erweiterung muss zunächst heruntergeladen, entpackt und in das richtige Verzeichnis kopiert werden. Nach der Installation und Konfiguration werden durch einen Neustart des Webservers die Änderungen sichtbar.

- Die Automatisierung des Conversion Rule Managements muss konfiguriert werden. Hierbei ist entscheidend, welchen Konvertierungsregeln vertraut wird und welchen nicht und welcher Grad der Automatisierung wünschenswert ist.

- Für die Konfiguration des Vertrauens ist es wichtig, den eigenen Wert `Advanced` sowie gegebenenfalls das Äquivalent in Maturity Level in die Metadaten einzufügen und die Mindestanforderungen für SPs festzulegen. Diese kann ebenso in den Metadaten erscheinen.

Diese Schritte müssen nicht nur für die Installation der LMU, sondern auch für die TUM und das LRZ durchgeführt werden. Die technischen Aspekte sind für die Erweiterung um

dynamisches FIM für den Service Provider ähnlich.

- Die Erweiterung muss zunächst heruntergeladen, entpackt und in das richtige Verzeichnis kopiert werden. Im nächsten Schritt wird das Installationsskript angestoßen. Anschließend müssen die benötigten Libraries (`ttp-sync`) installiert werden. In der Konfigurationsdatei müssen `OutOfProcess` und `InProcess` Tags hinzugefügt werden. Zusätzlich wird ein neuer Handler im Sessions Element benötigt, was gegebenenfalls eine Anpassung der ACL verursacht. Zudem sollte der Ordner, in dem Metadaten gespeichert werden, schreibbar sein. Im letzten Schritt muss ein neuer `MetadataProvider` hinzugefügt werden. Nach einem Neustart des Webservers und des `shibd` Prozesses, werden die Änderungen sichtbar.

- Für die Konfiguration des Vertrauens ist es wichtig, die Mindestanforderungen an den IdP als LoA und als Äquivalent in Maturity Level in die Metadaten einzufügen. Zudem soll der eigene Level of Trust analysiert und ebenfalls in die Metadaten geschrieben werden.

Diese beiden Schritte sind für alle SPs relevant. Die Änderungen für IdP und SP sollen zunächst in einem Testsystem getestet und gegebenenfalls angepasst werden, bevor auf das Produktivsystem umgestellt wird.

7.1.3. Organisationsübergreifende Aspekte

Neben den genannten internen organisatorischen und technischen Aspekten müssen auch organisationsübergreifende Aspekte betrachtet werden. Nachdem die LMU in der Föderation DFN-AAI und in der Inter-Föderation eduGAIN integriert ist und das BAS ein SP von CLARIN ist, sind Absprachen mit den Beteiligten essentiell. Hierbei ist die Koordination bei der Umstellung besonders wichtig. Die DFN-AAI sowie CLARIN sollen zudem die Föderationsverwaltung der MdFIM anpassen und die eigene Workflows einpflegen. Dabei können bereits vorhandene Mitglieder importiert werden, was den Aufwand der Registrierung reduziert.

7.2. Spezifikation der Zielarchitektur

In diesem Abschnitt wird diskutiert, wie die vorhandene FIM-Architektur für Identity Provider und Service Provider erweitert werden muss, um dynamisches FIM zu realisieren. Hierfür soll die Zielarchitektur und Zielkonfiguration beschrieben werden. Der darauf anschließende Schwerpunkt liegt bei der Kostenabschätzung. Diese basiert auf der Auswahl und Konkretisierung der Werkzeuge, die im darauf folgenden Abschnitt realisiert werden.

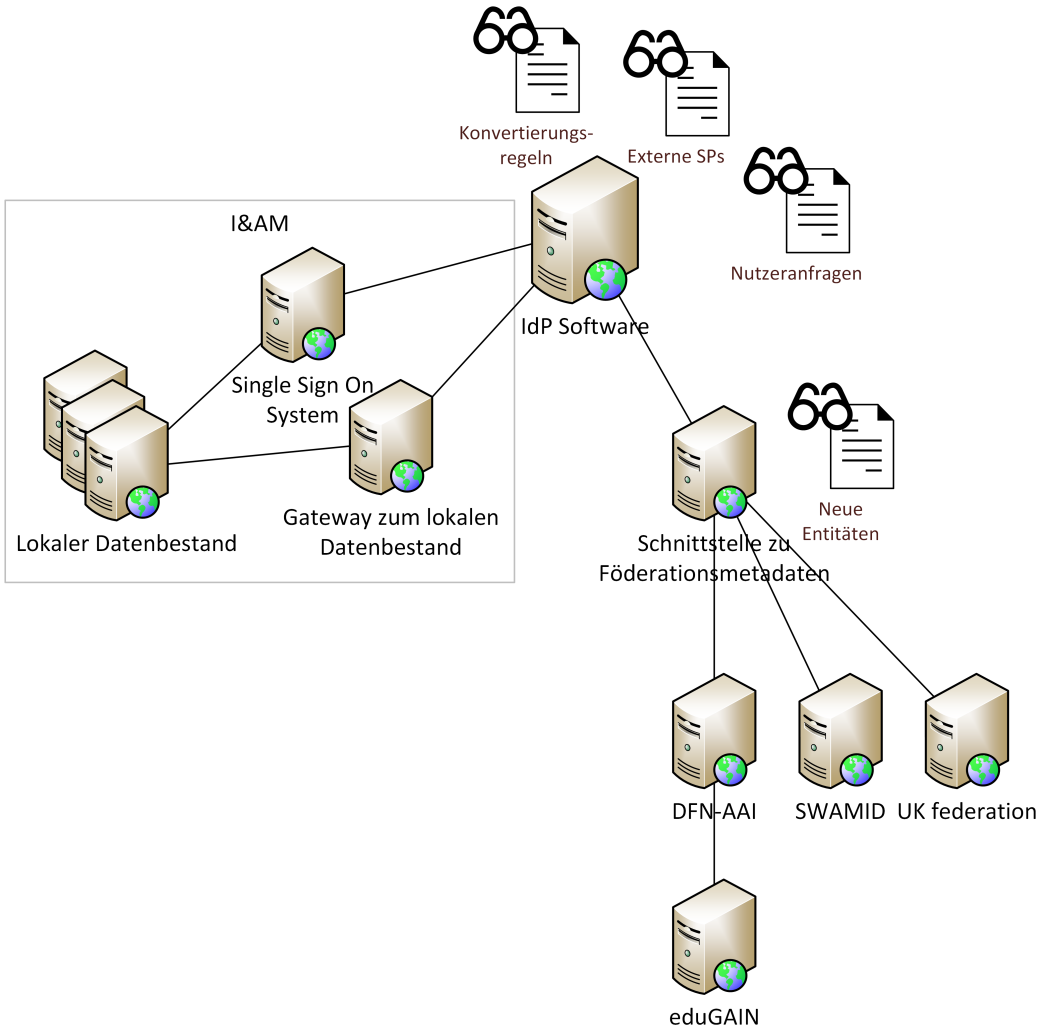

Abbildung 7.2.: FIM-Architektur des IdP LMU mit traditionellem FIM

7.2.1. Erweiterung der Architektur

FIM wird beim IdP LMU und beim SP BAS bereits eingesetzt. Die aktuelle Architektur ist in den Abbildungen 7.2 und 7.4 dargestellt. Um die Vorzüge von MdFIM nutzen zu können, müssen Änderungen durchgeführt werden. Diese sehen sowohl für IdP als auch SP wie folgt aus. Die geänderte Architektur ist in den Abbildungen 7.3 und 7.5 abgebildet.

- Die Erweiterung um dynamisches FIM ist die zentrale Änderung und somit auch eine wichtige Komponente, die benötigt wird. Die Erweiterung (Nr. 1 in den beiden Abbildungen 7.3 und 7.5) ist dafür zuständig, dass Metadaten dynamisch ausgetauscht werden. Nachdem die LMU unterschiedliche Nutzer bedient, die verschiedene Diens-

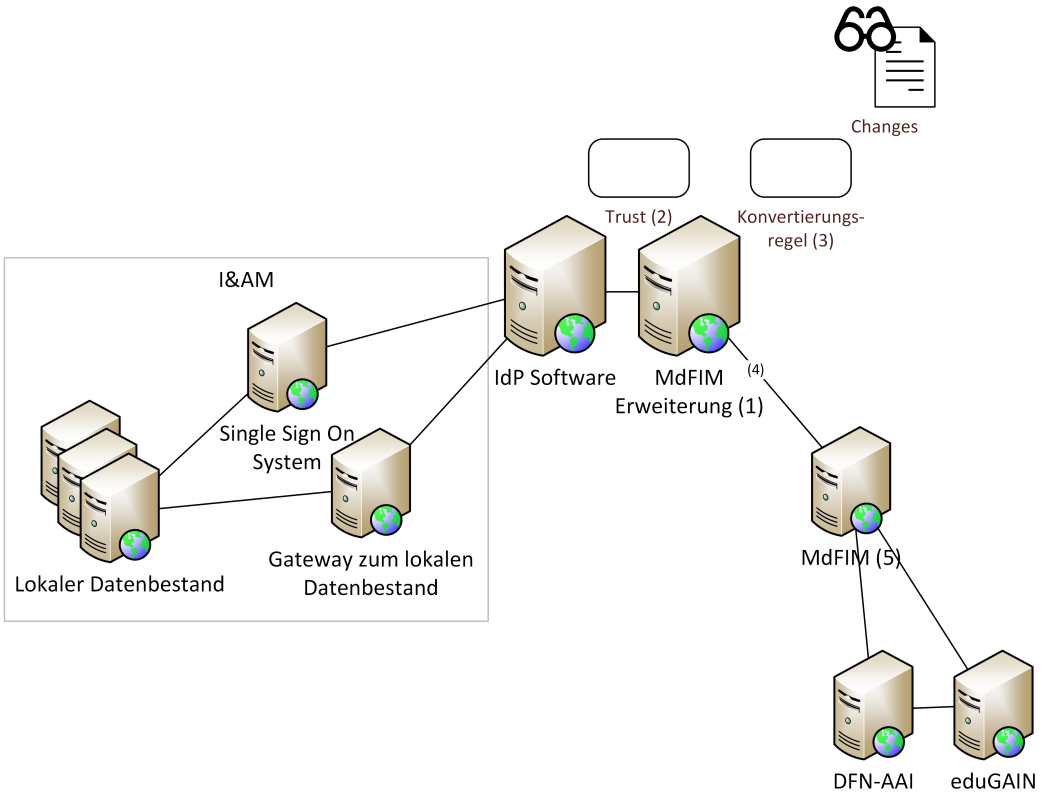

Abbildung 7.3.: FIM-Architektur des IdP LMU mit dynamischen FIM

te in Anspruch nehmen, stellt dynamisches FIM eine automatisierte Lösung dar. Zur Realisierung muss die Erweiterung heruntergeladen, installiert und die Konfiguration angepasst werden.

- Administratoren können durch eine Erweiterung direkt über die Kommandokonsole mit der MdFIM zu kommunizieren (Nr. 4 in Abbildung 7.3 und Nr. 3 in Abbildung 7.5), ohne das Webinterface benutzen zu müssen. Diese Kommunikationsschnittstelle ist für die meisten Aktionen möglich und ist Administratoren von IdPs durch ihre tägliche Arbeit geläufig.

- Zudem wird innerhalb der Gesamtarchitektur eine MdFIM (Nr. 5 in Abbildung 7.3 und Nr. 4 in Abbildung 7.5) benötigt. Bei dieser MdFIM muss sich der IdP der LMU und der SP von BAS registrieren.

Spezielle Änderungen für den IdP ergeben sich aus den folgenden Aspekten:

- Um die Konvertierungsregeln (Nr. 3 in Abbildung 7.3) zu automatisieren, wird Conversion Rule Management aktiviert und die Konfiguration entsprechend angepasst.

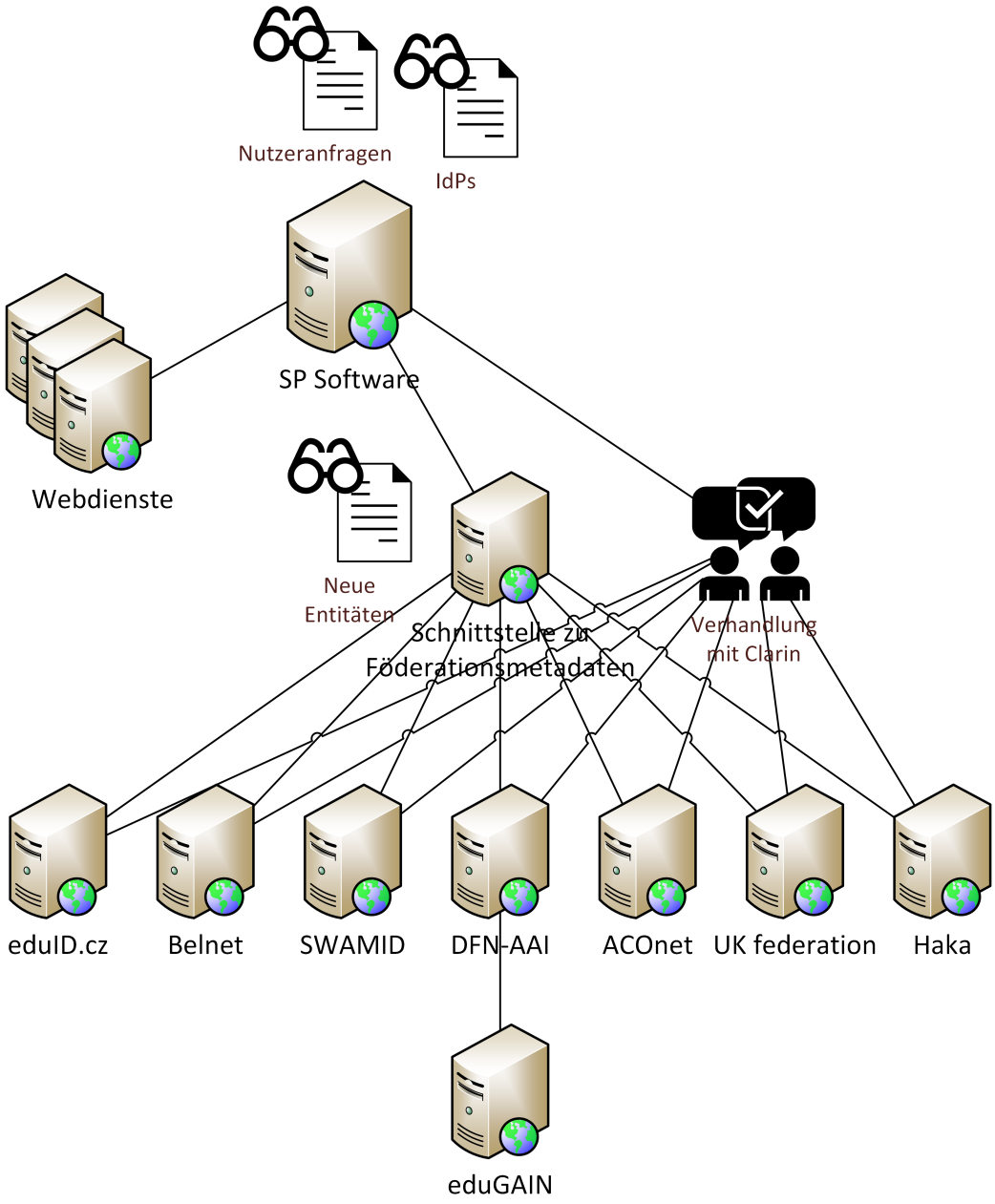

Abbildung 7.4.: FIM-Architektur des SP BAS mit traditionellem FIM

Konvertierungsregeln der Föderation DFN-AAI und der Inter-Föderation eduGAIN können grundsätzlich vertraut und somit automatisiert geladen werden. Andere Konvertierungsregeln werden zwar heruntergeladen, aber erst durch den Administrator autorisiert.

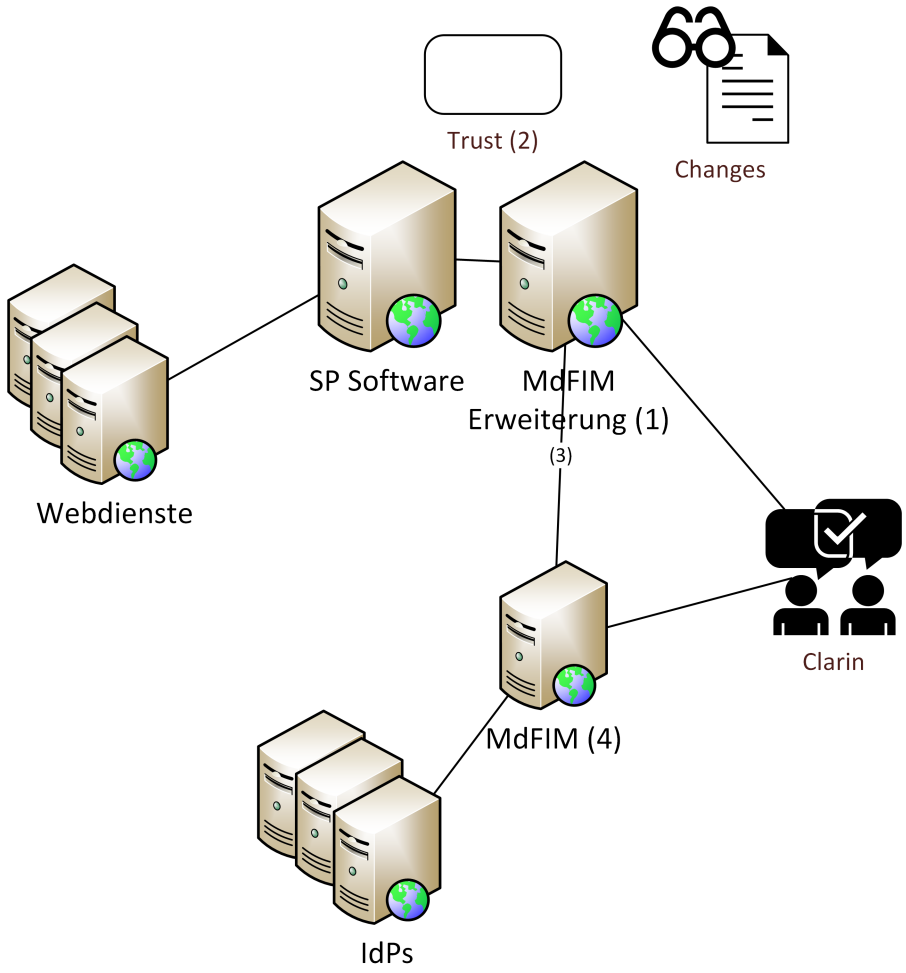

Abbildung 7.5.: FIM-Architektur des SP BAS mit dynamischen FIM

- Für das Trust Management (Nr. 2 in Abbildung 7.3) muss ebenfalls die Konfiguration angepasst werden. Neben dem Eintrag des LoAs Advanced und gegebenenfalls des äquivalenten Maturity Levels in den Metadaten muss basierend auf einer systematischen Risikoanalyse eine Mindestanforderung an SPs festgelegt werden. Nachdem diese Mindestanforderung bisher nicht explizit vorhanden war, müssen interne Verfahren und Entschlüsse auf das Schema des LoT angewandt werden. Diese Angabe wird ebenfalls in den Metadaten eingetragen.

Die Komponenten für LRZ und TUM sind identisch, wodurch die Migration durch die gesammelten Erfahrungen leichter fallen sollte. Die Konfiguration muss entsprechend der Wünsche angepasst werden. Dabei müssen unterschiedliche Ausrichtungen der Datenschutzbeauftragte insbesondere bei der Konfiguration des Trust Managements beachtet werden.

Die Anpassungen für die SPs sind ähnlich. Während keine Konvertierungsregeln benötigt werden, müssen Dienste, die bisher nicht an FIM angebunden sind, angebunden werden:

- Im Trust Management (Nr. 2 in Abbildung 7.5) müssen die Mindestanforderungen an IdPs in den Metadaten spezifiziert werden. Durch die Umstellung von mehreren Metadatenstreams auf dynamischen Metadatenaustausch mit Trust Management können hierbei bestehende Angaben auf ihre Aktualität, basierend auf das Risikomanagement, überprüft werden. Neben dieser Angabe ist es zudem notwendig, den eigenen Level of Trust zu definieren. Hierfür müssen die einzelnen Aspekte des Trusts betrachtet und die eigene Einordnung in den Metadaten deutlich gemacht werden.

- Um SPs, die bisher kein FIM verwenden, anzubinden, kann die Integrationslogik von Wolfgang Hommels Dissertation [Hom07] (Kapitel 7) angewandt und um die Änderungen durch die Erweiterung ergänzt werden.

Diese Anpassungen sind ebenfalls bei anderen SPs der TUM, LMU und des LRZs durchzuführen.

Möglichst gleichzeitig dazu muss von einer zentralen Stelle, wie der DFN-AAI, CLARIN oder GÉANT eine MdFIM aufgesetzt werden. Dies soll parallel zur aktuellen Infrastruktur geschehen. Zunächst ist hier die Managementplattform zu installieren und anschließend hinsichtlich Workflows zu konfigurieren. Um die Registrierung der einzelnen Entitäten zu erleichtern, sollen die eduGAIN-Metadaten als Bulk-Import in die MdFIM eingetragen werden. Die Entitäten müssen anschließend ihre Benutzerkonten bestätigen, jedoch sich nicht ursprünglich registrieren. Vorhandene Konvertierungsregeln in den Föderationen sowie deren Workflows sollen ebenso importiert werden.

7.2.2. Grundlegende Aufwandsprognose

Um für die Einführung von dynamischen FIM anfallende Kosten grob abschätzen zu können, wird im Folgenden eine grundlegende Aufwandsprognose durchgeführt. Dabei wird kurz auf die szenarienspezifischen Hardware- und Softwarekosten eingegangen. Zudem wird grob der Personalaufwand für die Migration aufgeführt.

Der Hardwareaufwand ergibt sich, wie bereits in [Hom07] aufgeschlüsselt, aus der Notwendigkeit, die Hochverfügbarkeit sicherzustellen. Im Gegensatz zum Stand bei [Hom07] wird am LRZ häufig Virtualisierung eingesetzt. Nachdem bereits mehrere VMs für Test- und Produktivumgebung eingesetzt werden, können für die Migration VMs aus der Testumgebung eingesetzt werden. Wenn die Installation auf der Testumgebung getestet und optimiert wurde, kann sie auf die Produktivumgebung migriert werden. Durch die bereits existierende Infrastruktur fallen keine neuen Hardwarekosten an. Dies gilt für den IdP der LMU. Eine ähnliche Infrastruktur wird für den SP mit dem Archiv, das ebenfalls am LRZ gehostet wird, vorausgesetzt.

Nachdem sowohl Shibboleth als auch die Erweiterung Open Source Produkte sind, fallen keine Lizenzkosten an. Bei Diensten, die zukünftig mit FIM betrieben werden sollen, können jedoch Anpassungskosten entstehen. Dies ist insbesondere bei kommerziellen Produkten der Fall, aber auch wenn Anpassungen an die SP-Software oder an die eigentliche Software notwendig sind. Die Entwicklungskosten hängen hierbei am Interesse für eine Änderung und die Entwicklungsdauer ab und müssen für jeden Dienst einzeln betrachtet werden. Zudem fällt eine Wartezeit bis zur Einsatzfähigkeit an, da die Anpassung zunächst entwickelt und getestet werden muss. Hierbei sind gegebenenfalls Softwarereleasezyklen und geeignete Programmierer zu beachten.

Der Personalaufwand hängt von verschiedenen Faktoren wie Kenntnisstand der Mitarbeiter und Auslastung, aber auch Koordination zwischen den unterschiedlichen Organisationen ab. Hierbei wird von Mitarbeitern ausgegangen, die über weitreichende Kenntnisse bei FIM, aber keine bis wenige Kenntnisse bei dynamischem FIM verfügen. Unterbrechungen wie Urlaub, Krankheit und Ausgleichstage, werden in der Abschätzung nicht berücksichtigt. Während bei der rein technischen Umstellung von 4,5 Wochen pro Entität ausgegangen wird, kann für die Absprachen mit anderen Einrichtungen und den internen Prozessen von einer Umstellungsdauer von etwa einem Jahr gerechnet. Dies kommt dadurch zustande, dass bei einer Community wie CLARIN alle Service Provider und Identity Provider möglichst strukturiert umstellen und Workflows festgelegt werden sollten. Im Gegensatz dazu stehen aktuell mehrere Verträge, verschiedene Tools und Schnittstellen für die Metadatenverwaltung sowie immer neu hinzukommende Konvertierungsregeln, die von Hand eingepflegt und die Konfigurationen entsprechend angepasst werden müssen.

In Hinblick auf den dauerhaften Betrieb sind mindestens zwei Administratoren zu schulen. Während im Regelbetrieb keine Eingriffe nötig sind, müssen Anpassungen an die Konfiguration und Updates der Software durchgeführt werden. Um das Vertreterprinzip möglich zu machen, werden daher mindestens zwei Administratoren benötigt. Diese sollten, nachdem bereits FIM bei SP und IdP eingesetzt werden, bereits vorhanden sein.

7.3. Realisierung

Die nachfolgende Beschreibung der Realisierung orientiert sich an den in Abschnitt 7.1 beschriebenen Schritten. Dabei wird grob auf den erforderlichen Aufwand eingegangen. Die Reihenfolge der Schritte orientiert sich an den zeitlichen Abhängigkeiten, während die Installation und Konfiguration zunächst auf einem Testserver getätigt wird. IdP und SP können gleichzeitig umgestellt werden.

1. Im ersten Schritt werden die Installation und die allgemeine Konfiguration der Erweiterungen getätigt.

    a) Die Erweiterung des IdP muss zunächst heruntergeladen, entpackt und in das richtige Verzeichnis kopiert werden. Um diesen Schritt zu vereinfachen, wird das

gitlab-Repository von GNTB eingesetzt. Die Datei `shib-idp-dame-extension-*.jar` muss hierfür in das Verzeichnis `edit-webapp/WEB-INF/lib` eingefügt werden. Die Konfigurationsdatei muss in das Konfigurationsverzeichnis, genauso muss die Flow-Datei in das Flow-Verzeichnis. Der Ordner, in dem zukünftig die Metadaten gespeichert werden, muss für den Nutzer des Webservers schreibbar sein. Dies lässt sich mit `/opt/shibboleth-idp# chown tomcat8:tomcat8 metadata/ttp` überprüfen. Zudem wird ein Eintrag mit dem Schlüssel `DameAccessByIPAddress` in der Datei `conf/access-control.xml` benötigt. Nach dem Durchlauf des Build-Skriptes und einem Neustart des Webservers werden die Änderungen sichtbar. Als nächstes kann die Automatisierung des Metadatenaustausches konfiguriert werden. Hierbei wird im Fall der LMU eine automatisierte Variante eingestellt, solange der SP minimale Kriterien erfüllt. Diese erstmalige Installation und Konfiguration nimmt kaum Zeit in Anspruch und sollte innerhalb eines Tages für Mitarbeiter, die dynamisches FIM noch nicht kennen, möglich sein. Die Installation und Konfiguration kann mit dem Testsystem von GNTB getestet werden.

b) Die Erweiterung des SPs muss zunächst heruntergeladen, entpackt und in das richtige Verzeichnis kopiert werden. Im nächsten Schritt wird das Installationsskript angestoßen. Anschließend müssen die benötigten Libraries (`ttp-sync`) installiert werden. In der Konfigurationsdatei müssen `OutOfProcess` und `InProcess` Tags hinzugefügt werden. Zusätzlich wird ein neuer Handler im Sessions Element benötigt, was gegebenenfalls eine Anpassung der ACL verursacht. Zudem sollte der Ordner, in dem Metadaten gespeichert werden, schreibbar sein. Im letzten Schritt muss ein neuer `MetadataProvider` hinzugefügt werden. Nach einem Neustart des Webservers und des `shibd` Prozesses, werden die Änderungen sichtbar. Ebenso wie beim IdP kann die Automatisierung des Metadatenaustausches konfiguriert werden. Für die Installation und Konfiguration wird ebenfalls 1 Tag eingerechnet. Der Test erfolgt ebenso mit dem Testsystem von GNTB.

2. Nach einem erfolgreichen Test kann die Automatisierung des Conversion Rule Managements konfiguriert werden. Hierbei muss festgelegt werden, welchen Quellen vertraut wird und welchen nicht und welcher Grad der Automatisierung wünschenswert ist. Diese Konfiguration nimmt keinen erheblichen Aufwand in Anspruch, jedoch kann eine Absprache der Konfiguration mit dem Datenschutzbeauftragten nötig sein. Der IdP der LMU vertraut automatisch allen Regeln der Föderation DFN-AAI und der Inter-Föderation eduGAIN, während andere Regeln nur heruntergeladen, aber nicht automatisch integriert werden. Bei der Dauer für die Festlegung diese Regelung wird von einer Dauer von 1 Woche ausgegangen.

3. Um das Trust Management zu konfigurieren, müssen zwei Vorbedingungen erfüllt werden: Das eigene Level kennen und eine Mindestanforderung an die gegenüberliegende Entität haben.

   a) Um das eigene LoA zu kennen, muss entweder der durch die Föderation vergebene LoA Advanced verwendet werden oder das Trust Assessment der zentralen

MdFIM befragt werden. Der eigene Trust Level wird entsprechend in den Metadaten eingefügt. Sind mehrere Levels in der Organisation vorhanden, müssen alle in den Metadaten stehen. Zudem muss der jeweilige Trust Level des Benutzers als Attribut gespeichert werden. Ist dies noch nicht möglich, muss das LDAP-Schema bzw. die entsprechende Datenbank um den Trust Level erweitert werden. Ferner muss in diesem Fall der Attribute Filter angepasst werden, um das Attribut an SPs zu senden. Nachdem SPs von CLARIN bei manchen Sprachdateien Zwei-Faktor-Authentifizierung verlangen, ist diese Änderung für den IdP der LMU notwendig. Dieses Verfahren ist ähnlich zur Ermittlung des eigenen LoT.

b) Die Mindestanforderungen für SPs basieren auf der eigenen Risikoanalyse und ihre Festlegung erfolgt gegebenenfalls in Absprache mit dem Datenschutzbeauftragten. Um den LoT zu bestimmen, können die einzelnen Aspekte und ihre Levels herangezogen werden. Der oder die Trust Levels sind ebenfalls in den Metadaten einzutragen. Diese Mindestanforderungen sind in diesem Beispiel sehr gering.

c) Die Mindestanforderungen sind ebenfalls für IdPs zu ermitteln. Nachdem für die Webanwendungen, die BAS über die Webseite anbietet, keine Authentifizierung verlangt wird, sind keine Anforderungen vorhanden. Für mögliche weitere Webanwendung müssen die Anforderungen entsprechend der Maturity Level bestimmt werden.

Während die Konfiguration an sich kaum einen Aufwand darstellt, kann die Ermittlung des LoA und des LoT theoretisch mehr Zeit in Anspruch nehmen; in dem Beispiel dieses Kapitels ist dies jedoch nicht der Fall. Die Erweiterung des LDAP-Schemas ist ebenfalls mit Zeit verbunden, nachdem hier die Werte der Benutzer eingetragen werden müssen und eine Anpassung des Attribute Filters folgt. Für den IdP wird hier mit einer Dauer von 1-2 Wochen gerechnet, während die Anpassung für den SP innerhalb eines Tages möglich ist.

4. Im nächsten Schritt muss die Mitgliedschaft in der Föderation geändert und die Registrierung bei MdFIM getätigt werden. Die Änderung bei der Föderation ist bedingt durch die geänderten Metadaten. Für diesen Schritt wird eine Zeitdauer von 1 Tag eingeplant.

5. Nach Abschluss aller Tests können die Erweiterungen in den Produktivbetrieb überführt werden. Hierbei sind die Mitarbeiter über die Änderungen zu informieren, beispielsweise in Form einer Präsentation vor allen interessierten Mitarbeitern. Dieser Schritt ist im Rahmen des Change Managements insbesondere mit dem Service Desk zu koordinieren, da anfangs eine erhöhte Anzahl von Suppportanfragen durch gestiegene Nutzung zu erwarten ist. Um möglichst viele Fragen beim First Level Support abzufangen, ist dieser vorab zu schulen. Zudem sollen für diesen Zeitraum im Second Level Support entsprechende Ressourcen bereitstehen, um die Anfragen mit Incidents und Problemen zu beantworten. Dieser Schritt wird mit grob 2 Wochen berechnet.

6. Entsprechend des Vorgehens beim IdP des LMU sind die IdPs der TUM und des LRZs umzustellen und zu konfigurieren. Diese Umstellung ist erst dann sinnvoll, nachdem ein erster IdP in Betrieb genommen und erste Erfahrungen gesammelt wurden. Die Umstellung wird mit je 2 Wochen eingeplant. Zudem müssen alle SPs umgestellt werden, was eine ähnliche Zeitspanne pro Dienst benötigt.

7. Nachdem alle bisher über FIM angebundenen Dienste auf dynamisches FIM umgestellt wurden, können neue Dienste und Dienste, die zusätzliche Komponenten wie Bridges benötigen, an dynamisches FIM angebunden werden. Falls seit Anfang des Projektes Ressourcen zur Entwicklung der Anpassungen verfügbar sind, sollen diese genutzt werden. Hierbei sind unter Umständen Absprachen mit externen Organisationen notwendig.

Allein für die Umstellung des IdPs der LMU wird dadurch von einer Mindestdauer von 4,5 Wochen gerechnet. Nachdem durch die Zusammenarbeit von mehreren Personen Terminkonflikte entstehen, Fehlersituationen bei Migrationen auftreten können und Absprachen notwendig sind, wird von mindestens 8 Wochen ausgegangen. Bei Betrachtung des Münchner Wissenschaftsnetz (MWN) in Zusammenspiel mit der DFN-AAI und der Community CLARIN kann hierdurch die Migration realistisch 1 Jahr benötigen. Diese Zeit ist vor allem durch Absprachen geprägt. Mit der Überführung der Erweiterung in die Benutzerphase fängt der Produktivbetrieb an, der im nächsten Abschnitt näher betrachtet wird. In dieser Zeit sollte auch die Umstellung von MDS auf MdFIM von Seiten GÉANTs möglich sein.

7.4. Operative Aspekte

In der Betriebsphase sind Eingriffe in die Architektur nur bei Fehlersituationen und im Rahmen des Change Managements möglich. Zusätzlich zu den Aspekten, die bereits von Wolfgang Hommel [Hom07] (Abschnitt 7.5) aufgeschlüsselt wurden, wird hier insbesondere auf die Änderungen der Architektur Wert gelegt. Diese Änderungen zeigen sich beim Change und Security Management, wie nachfolgend zu sehen.

7.4.1. Change Management

Allgemein sollen durch das Change Management Änderungen nur geplant und koordiniert durchgeführt werden, wobei die Risiken zu beachten sind. Dies wurde bereits in Abschnitt 4.8 allgemein betrachtet. Änderungen gibt es auf verschiedenen Ebenen:

- Einführung des dynamischen FIM inklusive Konfiguration,

- dynamischer Metadatenaustausch,

- automatisierte Prozesse,
- geänderte Metadaten,
- Trust Level,
- Konvertierungsregeln,
- geänderte benötigte Attribute und
- Teilnahme an föderationsähnlichen Strukturen.

Kleine Änderungen, wie die folgt dargestellten, gelten als pre-authorized Changes:

- Verschwinden von Entitäten,
- Hinzufügen neuer Entitäten innerhalb der akzeptierten Trust Levels,
- etablierte Prozesse, die zuvor manuell getätigt wurden,
- geänderte Metadaten,
- Änderungen an Trust Levels und
- neue Konvertierungsregeln von vertrauenswürdigen Quellen.

Service Provider unterhalb des benötigten Mindestvertrauens können genauso wie kommerzielle Dienste sollten nach einem 4-Augen-Prinzip angebunden werden, was als ein Change Model beschrieben wird.

Änderungen von SPs hinsichtlich der geforderten Attribute werden zunächst als Changes gehandhabt und genehmigt. Werden deutlich mehr Attribute angefordert, beispielsweise 10 statt ursprünglich 2, sollte dies genauer betrachtet werden, um den Verlust von Benutzerinformationen zu vermeiden. Was deutlich mehr Attribute bedeutet, kann nach einem Jahr genauer bestimmt und anschließend bei einem jährlichen Review angepasst werden. Als erstmaliger Wert werden 3 zusätzliche Attribute angegeben. Wird dieser Wert häufiger überschritten für Dienste, die dies berechtigt oder nachvollziehbar anfordern, kann der Wert erhöht werden.

Geänderte Teilnahmen an föderationsähnlichen Strukturen haben ebenfalls komplexe Änderungen, insbesondere wenn Policies akzeptiert werden müssen, so dass diese Änderungen ebenso genehmigt werden sollen. Virtuelle Föderationen ohne Policies können dahingegen als kleinere Änderungen angesehen werden, die beispielsweise bei wechselnden Projekten relativ häufig anfallen können. Die Teilnahme an virtuellen Föderationen ist daher ein pre-authorized Change. Beim SP BAS von CLARIN hängt das stark davon ab, ob weiterhin Verträge mit den Föderationen abgeschlossen oder Vertrauensbeziehungen mit den benö-

tigten IdPs aufgebaut werden. Da letzteres den Aufwand deutlich reduziert, während die Reichweite trotzdem erhalten bzw. erweitert wird, wird von dieser Variante ausgegangen. Folglich wird zwar eine offizielle Föderation CLARIN innerhalb der MdFIM gebildet, aber alle weiteren Vertrauensbeziehungen basieren auf dem Metadatenaustausch zwischen zwei Entitäten, was wiederum einen vorautorisierten Change darstellt.

Durch die Erfahrungen bei der Migration von mehreren IdPs und der mit den Universitäten umgestellten SPs, können die Münchner Universitäten bei der Erstellung einer Leitlinie bezüglich der Migration mitwirken und anderen Universitäten helfen.

7.4.2. Security Management

Bei der Einführung des dynamischen FIM ergeben sich für alle beteiligten Organisationen Änderungen. Insbesondere die Automatisierung ist im Rahmen des Security Managements zu betrachten. Da einmal vollständig abgerufene Benutzerdaten nicht wieder durch den IdP gelöscht werden können, ist dies besonders kritisch. Das Ansehen des IdPs könnte durch einen solchen Datenmissbrauch geschädigt werden; im schlimmsten Fall droht eine Geldstrafe. Daher sind eine sichere Konfiguration und eine fehlerfreie Funktion besonders wichtig. Folglich muss bei der ersten Konfiguration das Security Management und damit einhergehend der Verlust von Daten betrachtet werden. Zusätzlich zu den von Wolfgang Hommel [Hom07] in Abschnitt 7.5.3 beschriebenen Maßnahmen muss die Auditierbarkeit auch die Automatisierung der Konfiguration berücksichtigen. Ein Zusammenspiel zwischen Change Management und Release Management muss sicherstellen, dass keine Fehlkonfiguration der FIM-Komponenten in der Produktivumgebung auftritt. Parallel zu dieser technischen Maßnahme muss die Sensibilisierung der Mitarbeiter bezüglich des Umgangs mit Personendaten einhergehen. Auch hier zeigt sich, dass technische Maßnahmen alleine keinen erfolgreichen Dauerbetrieb gewährleisten können.

7.5. Bewertung der Lösung für das Anwendungsbeispiel

Das ausgewählte Anwendungsbeispiel weist durch die Kombination mehrerer in Kapitel 2 vorgestellter Szenarien eine relativ hohe Komplexität auf. Bei der Anwendung der erarbeiteten Architektur hat sich gezeigt, dass die spezifizierte Architektur aus Kapitel 4 sowie die Werkzeuge in Kapitel 5 ohne Anpassungen auf das Beispiel angewandt werden konnten. Die Integration der Erweiterungen in die FIM-Infrastruktur bereitete keine Schwierigkeiten. Durch die Erweiterung werden bisher manuelle Arbeitsschritte automatisiert und die Skalierbarkeit der Gesamtarchitektur verbessert. Dies wird insbesondere am SP BAS und dessen Mitarbeiter am IdP LMU deutlich.

Dadurch, dass CLARIN mit weiteren Föderationen Verträge abschließt, werden aktuell mehr Verträge benötigt. SPs müssen sich an alle Eigenheiten der einzelnen Föderationen,

wie unterschiedliche Metadaten-Tools und Policies, richten. Zudem fallen hier häufiger manuell eingetragene Konvertierungsregeln und Rücksprache mit Administratoren wegen nicht versendeten Attributen an. Somit ist manuelle Arbeit insbesondere bei Verträgen, Konfiguration und der Anpassung an Föderationen notwendig.
Durch den Einsatz von MdFIM kann zum einen der organisatorische Aufwand durch den Wegfall von Verträgen reduziert werden. Zum anderen werden Metadatenaustausch und die Einbindung neuer Konvertierungsregeln weitgehend automatisiert, wodurch hier weniger Arbeit anfällt. Durch die einmalige Bestimmung und Konfiguration des Vertrauens werden ganz unsichere Entitäten trotzdem ausgeschlossen, so dass eine gewisse Sicherheit, entsprechend der Risikoanalyse, besteht. Dies stellt eine Verbesserung zu den in Kapitel 3 vorgestellten Ansätzen und der Ist-Situation aus Kapitel 2 dar. Neben der untersuchten Skalierbarkeit kann durch die Anwendung auf das Beispiel BAS von der praktischen Anwendbarkeit des Konzepts und der Werkzeuge ausgegangen werden.
Bei der prototypischen Anwendung des Konzepts auf BAS hat sich gezeigt, dass organisationsübergreifende Abstimmungen speziell bei der Einführung notwendig sind. Dieser Aufwand wird durch die Automatisierung und Vereinfachung der Strukturen insbesondere für Communities schnell aufgewogen.

Kapitel 8

Fazit

Inhalt dieses Kapitels

Durch die zunehmende Vernetzung der Forschung, aber auch in anderen Sektoren, und die gestiegene Mobilität werden skalierbare Architekturen für das dem IT-Sicherheit zugeordnete Federated Identity Management relevant. Als FIM wird die aktuelle Form des organisationsübergreifenden Identity Managements mit ihrer Software und Architektur bezeichnet. Um als Nutzer nicht mehrere Benutzerkonten zu pflegen und die Redundanz der Daten zu reduzieren sowie deren Qualität zu erhöhen, wurde FIM eingeführt. Nachdem die aktuelle Architektur mit ihren vorab ausgetauschten, aggregierten Metadaten durch die Vielzahl an Teilnehmern an ihre Grenzen stößt, müssen skalierbare, dynamische Lösungen entwickelt werden. Durch einen dynamischen Metadatenaustausch werden sowohl die Größe des ausgetauschten Metadatensatzes reduziert als auch die Reichweite erhöht und somit technische Barrieren abgebaut. Hierfür wurde in dieser Arbeit ein skalierbarer Ansatz über eine TTP entwickelt und durch eine Managementplattform um relevante Funktionen erweitert.

Im folgenden Abschnitt 8.1 werden die wichtigsten Aspekte dieser Arbeit zusammengefasst. Eine Diskussion der Ergebnisse und deren Weiterverwendung folgen in Abschnitt 8.2. Anschließend wird ein Ausblick auf offenen Punkte unterteilt in FIM-spezifische Arbeitsbereiche und verwandte offene Forschungsfragestellungen in Abschnitt 8.3 gegeben. Das Vorgehen ist in Abbildung 8.1 dargestellt.

8.1. Zusammenfassung dieser Arbeit

Im *Kapitel 2* wurden zunächst die *Grundlagen* von I&AM und FIM erklärt, bevor verstärkt auf Inter-FIM eingegangen wurde. Nachdem Inter-FIM noch relativ neu ist und unterschiedliche Begriffe verwendet werden, wurden somit Grundlagen für die Arbeit geschaffen. Zudem wurde das MNM-*Dienstmodell* mit dem Basismodell und der Dienstsicht auf FIM ange-

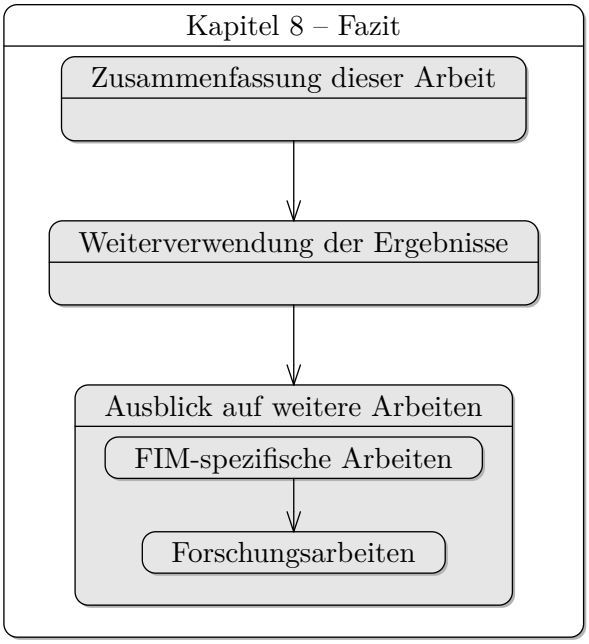

Abbildung 8.1.: Vorgehensmodell in diesem Kapitel

wandt, um es anschließend auf Inter-FIM und die darauf folgenden Szenarien anzupassen. Die realistischen Szenarien boten einen breiten und zugleich umfassenden Überblick über FIM und Inter-FIM:

- Szenario 1: Inter-FIM innerhalb der Inter-Föderation eduGAIN am Beispiel des LRZs.
- Szenario 2: Die SP-Föderation der Community CLARIN.
- Szenario 3: Die Community Grid mit Beteiligung des LRZs.
- Szenario 4: Ein fiktives Wirtschaftszenario mit Beteiligten aus der Automobilbranche, R&E sowie einem Startup.
- Szenario 5: Die benutzerbezogene Technologie UMA, welche einen Teilbereich von UCIM darstellt.

Durch die eben genannten realistischen Szenarien wurde ein breites Spektrum an Anwendungsfällen, verwendeten Technologien und Anforderungen abgedeckt. In jedem Szenario wurden funktionale, nicht-funktionale, organisatorische, sicherheitsrelevante und datenschutzspezifische Anforderungen erhoben. Die über 70 erhobenen Anforderungen wurden in einem gewichteten Anforderungskatalog zusammengefasst, der zur Bewertung bisheriger Ansätze und des in dieser Arbeit erstellten Konzepts verwendet wurde.

Im darauf folgenden *Kapitel 3* wurden die Grundlagen der eingesetzten Technologien beschrieben und der aktuelle Stand der Forschung durch

FIM-Standards: SAML und OAuth bzw. OpenID Connect.

SAML Implementierungen: Shibboleth, SimpleSAMLphp, PySAML2 und ADFS.

Technisches Vertrauen durch Metadaten: Resource Registry, IdP-Proxy, Metadata Distribution Service und Metadata Query Protocol mit PEER.

Forschungsansätze zu Vertrauen in Föderationen: DIMDS, FAMTN, IdMRep, Dynamic Identity Federation und TSP.

Forschungsansätze zur Interoperabilität von Attributen: Ontologische Ansätze, CCS und FSCS.

Level of Assurance: In Föderationen und Normen.

aufgezeigt. Die beschriebenen Ansätze wurden jeweils mit dem im vorherigen Kapitel definierten Anforderungskatalog verglichen. Zunächst wurden die Protokolle SAML und OAuth bzw. das darauf basierende OpenID Connect dargestellt. Nachdem SAML ein Protokoll ist, wurden zudem die Open Source Implementierungen Shibboleth, SimpleSAMLphp und PySAML2 sowie die kommerzielle Implementierung ADFS miteinander verglichen. Ein wichtiges Element in aktuellen Föderationen ist das technische Vertrauen durch Metadaten, ohne das kein FIM möglich ist. Hier wurden die Ansätze Resource Registry, Metadata Distribution Service, IdP Proxy, Metadata Query Protocol und PEER vorgestellt und auf ihre Einsatzfähigkeit überprüft. Zusätzlich wurden Forschungsansätze zum Vertrauen in Föderationen und zur Interoperabilität von Attributen betrachtet. Die Interoperabilität von Attributen ist essentiell, damit Service Provider benötigte Benutzerinformationen in der richtigen Syntax und Semantik erhalten. Um diese Informationen zu senden, muss ein Vertrauensverhältnis zwischen IdP und SP bestehen. Neben den ausgetauschten Metadaten (technisches Vertrauen) ist das Vertrauen in die gegenüberliegende Entität relevant. Dieses kann durch Level of Assurance sichtbar gemacht werden. Daher wurden anschließend Normen und Ansätze in den Föderationen zum Level of Assurance analysiert und gegenüber gestellt. Abschließend wurden die vorhandenen Protokolle und SAML-Implementierungen miteinander verglichen und die Defizite im Vergleich mit den in Kapitel 2 aufgestellten Anforderungen herausgestellt.

Kapitel 4 konzipierte die Architektur für dynamisches FIM. Zunächst wurde das Konzept von dynamischen virtuellen Föderationen und Inter-Föderationen sowie Föderationsverwaltungen beschrieben, um eine möglichst große Anzahl an Kooperationen zu unterstützen. Basierend darauf wurde eine Managementarchitektur für dynamisches Federated Identity Management anhand der Teilmodelle

- Informationsmodell,

- Organisationsmodell,
- Kommunikationsmodell und
- Funktionsmodell

etabliert. Das Informationsmodell beschrieb die wichtigen Informationsbausteine basierend auf Domänen, die für dynamisches FIM notwendig sind, während das Organisationsmodell unterschiedliche Domänen und ihre Rollen definierte. Darauf aufbauend wurde die Kommunikation zwischen Rollen und Domänen aufgezeigt. Abschließend wurden durch das Funktionsmodell zahlreiche Funktionen analysiert, die für dynamisches FIM benötigt werden. Als Beispiele hierfür gelten Conversion Rule Management, Metadata Management, Member Management, Service Management, Configuration Management und Trust Management. Das Ziel und der Umfang basieren dabei auf dem in Kapitel 2 definierten Anforderungskatalog. Die wesentlichen Neuerungen zur aktuellen Situation sind die folgenden Elemente:

- Dynamischer Metadatenaustausch über eine TTP.
- Managementplattform verbunden mit einer TTP zur Verwaltung von Entitäten und Föderationen.

Anschließend wurde die Integration in die bestehende Umgebung demonstriert, bevor das IT Service Management mit dem Fokus auf Security Management und Change Management auf das Konzept angewandt wurde.

Darauf aufbauend wurden im *Kapitel 5* drei Werkzeuge für diese Architektur entwickelt:

- Managementplattform MdFIM mit der technischen TTP,
- Conversion Rule Management und
- Trust Management.

Als zentrale Komponente wurde die Managementplattform MdFIM genauer definiert. Sie bildet mit ihrer Datenbank, der Dateiablage, API und Verwaltungsfunktionen die Basis für das anschließend beschriebene Conversion Rule Management und Trust Management. Durch die verbundene TTP wird der dynamische Metadatenaustausch orchestriert. Zudem helfen Funktionen wie Configuration Management, Policy Management und Member Management Entitäten und Föderationen das dynamische FIM zu unterstützen.
Durch das Conversion Rule Management können generische Konvertierungsregeln für Benutzerattribute gespeichert und wiederverwendet werden. Um sie an die SAML Implementierung, wenn bei OpenID Connect ein Schema bekannt ist auch daran, anzupassen, müssen diese generischen Regeln in die Form der Implementierung transformiert und anschließend in die lokale Software integriert werden. Das Conversion Rule Management trägt dazu bei, die Verwendung von Diensten zu beschleunigen und bisher manuelle Schritte zu automatisieren,

ohne dabei auf Sicherheit zu verzichten.
Abschließend wurde das Trust Management spezifiziert. Um verschiedene Level of Assurance miteinander zu vergleichen, wurde ein eigenes LoA Schema mit verschiedenen, orthogonalen Maturity Level eingeführt. Diese Maturity Level können neben dem reinen Vergleich beim Metadatenaustausch durch MdFIM zudem als eigenes LoA eingesetzt werden. Es basiert auf einer erweiterten Schnittmenge der in den LoA verwendeten Aspekte und beschreibt den Trust Level als mehrere Aspekte, wie Vectors of Trust. Äquivalent wurde für die Einordnung von Service Providern ein Level of Trust entwickelt. Damit verschiedene Levels miteinander vor dem Metadatenaustausch verglichen werden können, benötigt die Managementplattform MdFIM eine Umrechnungstabelle und Funktionen zum Vergleich, die im Workflow aufgerufen werden. Diese Funktionalität trägt dazu bei, den Metadatenaustausch sicherer zu gestalten und den Verbindungsaufbau zwischen Identity Provider und Service Provider zu vereinfachen.

Aufbauend auf dem bereits beschriebenen Konzept und den Werkzeugen wurde in *Kapitel 6* eine prototypische Implementierung der Architektur als Tragfähigkeitsnachweis gezeigt. Nachdem eine vollfunktionsfähige Managementplattform und die benötigten Erweiterungen den Rahmen dieser Arbeit gesprengt hätten, wurden ausgewählte Komponenten implementiert:

- MdFIM bzw. TTP mit rudimentären Funktionen,
- Trust Management mit Abgleich des Vertrauens und
- Conversion Rule Management mit der Generierung von Konvertierungsregeln für Shibboleth.

Abgerundet wurde dieses Kapitel durch die Untersuchung zur Skalierbarkeit. Dabei zeigte sich, dass die Skalierbarkeit gegenüber statischem FIM hinsichtlich der Schritte und ausgetauschten Metadatensätze erheblich verbessert wurde.

Um die Machbarkeit der Migration zu beweisen, wurde in *Kapitel 7* eine prototypische Anwendung an einem Beispiel-Szenario durchgeführt. Um mehrere Szenarien aus Kapitel 2 abzudecken, wurde das Beispiel Bayerische Archiv für Sprachsignale verwendet, die ein Service Provider von CLARIN sind. Da BAS organisatorisch zur LMU gehört, wurde von einem Identity Provider LMU ausgegangen. An einem exemplarischen Projekt wurde die Einführung von dynamischen FIM mit den Aspekten Integration und organisatorische Aspekte theoretisch durchgeführt. Zudem wurden die Aufwände im Gegensatz zur aktuellen Situation betrachtet und annäherungsweise eine Kostenabschätzung behandelt. Aufbauend auf einer grundlegenden Konfiguration wurden als Abschluss der Beiträge dieser Arbeit die Schritte der Migration und Einführung bedacht.

8.2. Weiterverwendung der Ergebnisse dieser Arbeit

Die Ergebnisse dieser Arbeit können folgendermaßen wiederverwendet werden.

- Der in Kapitel 2 aufgestellte Anforderungskatalog kann für die Neueinführung von FIM und dynamischem FIM als Kriterienkatalog eingesetzt und gegebenenfalls angepasst werden.

- Das Kapitel 3 umfasst die Analyse des aktuellen Standes. Durch die umfassende Betrachtung von aktuellen Ansätzen können Nachfolgearbeiten Synergien nutzen und darauf aufbauen.

- Die plattformunabhängige Managementarchitektur in Kapitel 4 zeigt auf, wo noch weitere Werkzeuge benötigt werden. Die beschriebenen Kommunikationen und Informationen für eine voll funktionsfähige Managementplattform können in weiteren Arbeiten ergänzt und implementiert werden. Die Integrationsmethodik kann für die Migration auf die in dieser Arbeit dargelegte Architektur eingesetzt werden, während das Change Management und das Security Management bei der Migration als Leitfaden dienen.

- Die in Kapitel 5 konzipierten Werkzeuge können in der Architektur eingesetzt werden. Insbesondere das Conversion Rule Management und das Trust Management sind sofort einsatzfähig und können auch bei statischem FIM verwendet werden. Die Konvertierung wird alltäglich bei Administratoren benötigt und auch Trust-Entscheidungen sind bei jedem neu verwendeten Dienst entscheidend. Nichtsdestotrotz wird versucht, Vectors of Trust zu verbessern und mitzubestimmen.

- Die in Kapitel 6 dargestellte Implementierung kann nachfolgend optimiert und produktiv genutzt werden. Zudem ist bereits eine Verwendung mit Shibboleth möglich, auch wenn eine Integration in Shibboleth die noch bessere Lösung wäre. Über das GÉANT Projekt wird versucht, die Implementierung anderen Entitäten, Communities und Föderationen bereitzustellen.

8.3. Ausblick auf weitere Arbeiten

In dieser Arbeit konnten einige für FIM allgemein und dynamisches FIM im speziellen relevante Themen nur oberflächlich behandelt werden. Die nachfolgende Liste gibt einen Überblick über Fragestellungen, die in Nachfolgearbeiten aufgegriffen werden sollen.

- Auch wenn es zur Usability und damit verbunden UMA einige Anforderungen gab, wurden sie im Rahmen dieser Arbeit bewusst ausgeklammert. Nachdem UMA weiter entwickelt wird, kann dieser Ansatz mit Einbeziehung weiterer, für UCIM typischer Gesichtspunkte einen dynamischen FIM-Ansatz mit dem Schwerpunkt auf den Benut-

zer bilden. Hierzu müssen die Protokolle und Workflows von UMA eingebunden und Untersuchungen bezüglich der Usability angestellt werden.

- Obwohl das Konzept und die Werkzeuge generisch konzipiert wurden, soll eine Implementierung auf Basis von OpenID Connect erfolgen. Dies hat den Vorteil, dass neue Konzepte von OpenID Connect schnell integriert und zukünftige Standards, wie Schemata für Benutzerinformationen, mitbestimmt werden können. Zudem erweitert diese Implementierung die Reichweite potentieller Nutzer, da insbesondere in Communities OpenID Connect verstärkt eingebunden wird.

- Die prototypische Implementierung des IdPs für Shibboleth Version 3.1 hat gezeigt, dass eine einfache Workflow Engine in Form von Flows Abläufe vereinfachen kann. Dadurch soll der Einsatz einer Workflow Engine für MdFIM in einer optimierten Implementierung bedacht und stringent umgesetzt werden.

- Um die Parallelität von mehreren TTPs zu vermeiden, soll das Konzept von verteilten TTPs und damit verteiltem Metadatenaustausch [Pöh15] näher betrachtet werden. Diese Topologie soll verhindern, dass Entitäten vielfach ihre Metadaten in parallelen MdFIM verwalten müssen. Nachdem eine weltweite Föderation einer Utopie gleicht, ist eine globale TTP ebenfalls als eine Utopie anzusehen. Um diese Nachteile zu vermeiden, muss ein Ansatz zur Registrierung und Vernetzung von TTPs geschaffen werden.

- Um die Größe und Struktur von dynamischen virtuellen Föderationen zu untersuchen, soll ein Clusterkoeffizient verwendet werden. Diese als Small-World bezeichneten Kooperationen und Verbindungen von Entitäten können durch unterschiedliche Graphen, wie Random Power Law Graphs, beschrieben werden. Unterschiede der Vertrauensbeziehungen zwischen festen Föderationen und diesen losen Verknüpfungen können zudem untersucht werden.

In dieser Arbeit hat sich gezeigt, dass auch Konzepte aus anderen Fachgebieten für dynamisches FIM erweitert werden müssen. So sollen Ontologien für Trust Levels untersucht werden, um den Vergleich einfacher zu gestalten. Ferner soll ein organisationsübergreifendes Security Management für FIM umgesetzt werden. Dies zeigt, dass im Bereich dynamisches FIM immer noch viele Fragestellungen offen sind, die beantwortet werden müssen.

Anhang A

Anhang

Inhalt dieses Kapitels

A.1. Dynamischer Metadatenaustausch

Das Protokoll zum dynamischen Metadatenaustausch ist unterteilt in die Authentifizierung des Nutzers (A) und den Metadatenaustausch (B).

Authentifizierung des Nutzers (A)

Schritt 1: Der Nutzer möchte einen Dienst des SPs verwenden. Als Vorbedingung müssen die Pfade zu den lokalen Implementierungen MdFIM bekannt sein. Nach der Auswahl des Dienstes durch den Nutzer, wird der HTTP Request des User Agents, normalerweise des Browsers des Benutzers, an die TTP gesendet.

Schritt 2: Beim Lokalisierungsdienst wählt der Nutzer seinen IdP aus, wodurch ein weiterer HTTP Request an den SP gesendet wird. Dieser Schritt entspricht weitgehend einem standardisierten SAML-Workflow und beeinflusst das Nutzererlebnis erheblich.

Schritt 3: Der Lokalisierungsdienst übermittelt die Auswahl über das Discovery Service Protocol an den SP. Der Authentication Request wird über den User Agent vom SP nicht an den IdP weiter geleitet, sondern an die TTP. Dieser Schritt ist notwendig, da IdPs auf Anfragen von ihnen unbekannten SPs nicht antworten.

Schritt 4: Nachdem der SP die Metadaten des gewählten IdPs nicht in seiner lokalen Konfiguration findet, sendet er einen Authentication Request an den IdP, der bei der Trusted Third Party Lokalisierungsdienst zwischengespeichert wird.

Schritt 5: Die TTP überprüft die Signatur des SPs, um die Authentizität sicher zu stellen.

Schritt 6: Zum Schutz vor Angriffen, muss sichergestellt sein, dass der Nutzer einen gültigen Account beim ausgewählten IdP besitzt. Dafür wird ein neuer Authentication Request an den IdP gesendet.

Schritt 7: Nun authentifiziert sich der Nutzer.

Metadatenaustausch (B)

Schritt 8: Nach erfolgreicher Authentifizierung des Nutzers wird der Metadatenaustausch durch die TTP angestoßen. Dieser kann beispielsweise über das Metadata Query Protocol geschehen. Damit die Metadaten ausgetauscht werden, muss dieser Schritt eingeleitet werden. Dies kann durch einen neuen HTTP Request geschehen, der eine definierte `action`, beispielsweise `action=fetchmetadata` in Zusammenhang mit der `entityID` als Parameter, verwendet. Nach der Abfrage der Metadaten muss die entsprechende Entität einen HTTP Response an die TTP mit dem Status der Integration senden. Da es möglich ist, dass der Server neu gestartet werden muss, soll dies in der Antwort ebenfalls ersichtlich sein. Nach dem Neustart wird hier ein weiterer Response notwendig, der den nun aktuellen Stand der Integration wiedergibt. Wenn IdP oder SP die Integration nicht erfolgreich abgeschlossen haben, soll die Integration zurück gerollt werden.

Schritt 9: Im Anschluss an den erfolgreichen Metadatenaustausch übermittelt die TTP den zuvor gespeicherten Authentication Request des SPs aus Schritt 4 an den IdP.

Schritt 10: Da der Nutzer erst erfolgreich authentifiziert wurde, existiert in der Regel eine gültige Sitzung.

Schritt 11: Darum leitet der IdP seinen Nutzer mit einer Authentication Response an den SP weiter.

Schritt 12: Der SP validiert die Response und der Nutzer kann anschließend den Dienst erfolgreich nutzen.

A.2. Application Programming Interface

User:

- `ttp User_create(username, password, givenName, surname, email, timestamp)`: Erstellung eines Benutzerkontos mit entsprechenden Variablen. Als Antwort kommt entweder ein `200 OK` oder eine Fehlermeldung zurück.

- `ttp User_update(username, password, givenName, surname, email, timestamp)`: Aktualisierung eines Benutzerkontos mit entsprechenden Variablen.
- `ttp User_delete(userID/username)`: Löschung eines Benutzerkontos.
- `ttp User_get(username)`: Hier werden als Antwort die Benutzerinformationen ausgegeben.
- `ttp User_link(userID1, userID2)`: Verknüpfung von zwei Benutzerkonten.

Entitäten:

- `ttp Entity_getMetadata(entityID)`
- `ttp Entity_getTrust(entityID`
- `ttp Entity_register(entityID, name, url, entitytype, aa, timestamp)`
- `ttp Entity_queryRegistration(EntityID)`: Abfrage, ob eine Entität bereits registriert ist.
- `ttp Entity_createOrganization(organizationName, url, timestamp)`: Hinzufügen einer Organisation.
- `ttp Entity_updateOrganization(organizationName, url, timestamp)`: Aktualisierung der Organisation.
- `ttp Entity_deleteOrganization(organizationID)`: Löschen einer Organisation.
- `ttp Entity_update(name, url, entitytype, aa, timestamp)`: Aktualisierung der Entität.
- `ttp Entity_delete(entityID)`: Löschen einer Entität.
- `ttp Entity_getStatus(entityID)`: Abfrage des Status einer Entität.
- `ttp Entity_setStatus(entityID, status)`: Setzen des Status einer Entität.
- `ttp Entity_queryValidation(entityID)`: Abfrage, ob eine Entität valide ist. Die Entität und die Metadaten werden nach der Registrierung validiert.
- `ttp Entity_setNotification(boolean)`: Die Art der Benachrichtigung bei Änderungen, beispielsweise von Metadaten, kann gesetzt und geändert werden.

Metadaten und Trust:

- Der Nutzer stößt über `ttp Metadata_initiate(entityID1, entityID2)` den Metadatenaustausch an.
- Die Methode `ttp Metadata_setAutomation(entityID, sort, rules)` konfiguriert die Automatisierung des Metadatenaustauschens.
- Um das Vertrauen zu berechnen, wurde im Informationsmodell die Methode `ttp Trust_calculateTrust(entityID1, entityID2)` angegeben.
- Metadaten können über die Methode `ttp Metadata_acceptMetadata(entityID, metadataID)` akzeptiert werden.
- Ebenso können Metadaten gesetzt (`ttp Metadata_setMetadata(entityID, metadataName, file/url)`) und
- aktualisiert (`ttp Metadata_updateMetadata(metadataID, metadataName, file/url)`) werden.
- Entsprechend sollen Metadaten gelöscht (`ttp Metadata_deleteMetadata(metadataID)` und
- abgefragt (`ttp Metadata_getMetadata(metadataID/entityID)`) werden können.

Konvertierungsregeln:

- `ttp CR_setConversionRule(conversionID, conversionType, conversionName, from, to, file)` erstellt eine neue Konvertierungsregel.
- `ttp CR_getConversionRule(conversionID)` lädt eine Konvertierungsregel herunter.
- `ttp CR_queryConversionRule(conversionType, conversionName, from, to)` fragt eine Konvertierungsregel ab.
- `ttp CR_approvesCR(conversionID, federationID)` bestätigt eine Konvertierungsregel.

Föderationen und Inter-Föderationen:

- `ttp Federation_applyMembership(federationID, entityID/federationID)` stellt den Antrag auf Aufnahme in einer Föderation.
- `ttp Federation_getCreationDate(federationID/federationName)`, um das Gründungsdatum der Föderation zu erhalten.
- `ttp Federation_getFederation(federationID)`, um die Informationen über eine Föderation zu bekommen.

- `ttp Federation_getFederationID(federationName)`, um die ID der Föderation abzufragen.
- `ttp Federation_getFederationName(federationID)`, um den Namen der Föderation zu erhalten.
- `ttp Federation_getStatus(federationID)`, um den Status der Föderation abzufragen.
- `ttp Federation_setCreationDate(federationID, timestamp)`, um das Gründungsdatum bei der Initiierung der Föderation zu setzen.
- `ttp Federation_setFederationName(federationID, federationName)`, um einen Föderationsnamen zu schreiben.
- `ttp Federation_setStatus(federationID, status)`, um einen Status zu speichern.
- `ttp Federation_getApplicationWorkflow(federationID)` gibt den Aufnahmeprozess wieder.
- `ttp Federation_setApplicationWorkflow(federationID, workflow)` legt den Aufnahmeprozess fest.
- `ttp Federation_setFederation(federationName, founder, foundingEntity, sortOfFederation, timestamp)` erstellt eine neue Föderation.
- `ttp Federation_inform(federationID, userID/entityID)` informiert über Änderungen der Föderation bzw. Inter-Föderation.

Policies:

- `ttp Policy_getPolicy(policyID)` gibt die aktuell gültige Policy der Föderation wieder.
- `ttp Policy_setPolicy(policyName, file, timestamp)` setzt eine Policy für eine Föderation.
- `ttp Policy_changePolicy(policyID, policyName, file, timestamp)`, um eine Policy zu ändern.
- `ttp Policy_deletePolicy(policyID)`, um eine Policy zu löschen.

Interne Zwecke:

- `ttp Internal_handoverFed(federationID, userID[old], userID[new])`, um eine Föderation zu übergeben bzw.

- `ttp Internal_ handoverIFed(federationID, userID[old], userID[new])` um eine Inter-Föderation zu übergeben.
- `ttp Internal_acceptChange(changeID, boolean)`, um Änderungen zu akzeptieren oder abzulehnen.
- `ttp Internal_correlateMonitoringData(timeframe, federation/geolocation/ ...)` um Monitoring-Daten zu korrelieren.
- `ttp Internal_provideMonitoringData(monitoringDataID, email)`, um Monitoring-Daten für andere bereit zu stellen.
- `ttp Meta_add(metaName, validFrom, validUntil)` wird benötigt, um allgemeine Informationen hinzuzufügen.
- `ttp Meta_update(metaID, metaName, validFrom, validUntil)` ändert diese Informationen.
- `ttp Meta_delete(metaID)` löscht allgemeine Informationen.
- `ttp [Entity | Federation]_changeTrustRule(ruleID, spID/category, idpID/-category, boolean, rule)` generische Methode um Vertrauenskonfiguration zu ändern
- `ttp [Entity | Federation]_createTrustRule(spID/category, idpID/category, boolean, rule)` erstellt eine generische Vertrauenskonfiguration
- `ttp [Entity | Federation]_deleteTrustRule(ruleID)` löscht diese Konfiguration

A.3. Vergleich von LoAs anhand von Maturity Levels

Aspect	DFN-AAI		InCommon	
	Basic	Advanced	Bronze	Silver
Identification	2	3	1	3
Data Management	1	3	3	4
Authentication	1	2	2	2
Assertions	1	1	3	3
Accountability	1	1	2	3
Organizational Management	1	1	2	4

Tabelle A.1.: Level of Assurance of DFN-AAI and InCommon described by the aspects

Abbildungsverzeichnis

Abkürzungsverzeichnis

A	Administrator
AA	Attribute Authority
AA-A	AA Administrator
AA-RM	AA Relationship Manager
AA-SD	AA Service Desk
AAI	Authentication and Authorization Infrastructure
ACL	Access Control List
ACOnet	Austrian Academic Computer Network
AD	Active Directory
ADFS	Active Directory Federation Services
AG	Aktiengesellschaft
AL	Assurance Level
AP	Assertion Presentation
API	Application Programming Interface
ARP	Attribute Release Policy
BAS	Bayerische Archiv für Sprachsignale
BMW	Bayerische Motoren Werke
CA	Certificate Authority

CAB Change Advisory Board

CAN Content Addressable Network

CCS Credential Conversion Service

CLARIN Common Language Resources and Technology Infrastructure

CM Change Manager

CM Credential Management

cn commonname

CoCo Code of Conduct

ConM Configuration Manager

Conv Konvertierungsregel

CoT Circle of Trust

CPU Central Processing Unit

CSIRT Computer Security Incident Response Team

CSM Customer Service Management

CSS Cascading Style Sheets

DAME Dynamic Automated Metadata Exchange

DDoS Distributed Denial of Service

DFN-Verein Verein zur Förderung eines Deutschen Forschungsnetzes e. V.

DFN-AAI DFN Authentication and Authorization Infrastructure

DHT Distributed Hash Table

DN Distinguished Name

DNS Domain Name System

DoS Denial of Service

DS Discovery Service

DTL Distributed Trust List

DIMDS Dynamic Identity Management and Discovery System

eCCS eduGAIN Credential Conversion Service

ECMA European Computer Manufacturers Association

ECP Enhanced Client or Proxy

EGCF European Globus Community Forum

EGI European Grid Infrastructure

eID elektronische Identität

eIDAS Electronic identification and trust services

ERIC European Research Infrastructure Consortium

ESGF Earth System Grid Federation

EU Europäische Union

FA Federation Administrator

FAMTN Federated Attribute Management and Trust Negotiation

FCAPS Fault, Configuration, Accounting, Performance, Security Management

Fed-A Federation Administrator

Fed-CM Federation Change Manager

Fed-ConM Federation Configuration Manager

Fed-GM Federation General Manager

Fed-RM Federation Relationship Manager

Fed-SD Federation Service Desk

Fed-TS Federation Technical Support

FIM Federated Identity Management

FM Federation Manager

FSC Forward Schedule of Changes

FSCS Federation Schema Correlation Service

FSD Federation Service Desk

GB Gigabyte

GM General Manager

GmbH Gesellschaft mit beschränkter Haftung

GNTB GÉANT-TrustBroker

GT Globus Toolkit

HTML Hypertext Markup Language

HTTP Hypertext Transfer Protocol

HTTPS Hypertext Transfer Protocol Secure

I-D Internet-Draft

ID Identifier

IdP Identity Provider

IdP-A IdP Administrator

IdPM IdP Manager

IdP-RM IdP Relationship Manager

IdP-SD IdP Service Desk

IDS Intrusion Detection System

IFed-A Inter-Federation Administrator

IFed-SD Inter-Federation Service Desk

IFed-TS Inter-Federation Technical Support

IFed-CM Inter-Federation Change Manager

IFed-ConM Inter-Federation Configuration Manager

IFed-GM Inter-Federation General Manager

IFed-RM Inter-Federation Relationship Manager

IFM Inter-Federation Manager

IFA Inter-Federation Administrator

IFSD Inter-Federation Service Desk

IR Identity Repository

I&AM Identity & Access Management

IANA Internet Assigned Numbers Authority

IAP Identity Assurance Profile

IDS Intrusion Detection System

IEC International Electrotechnical Commission

IETF Internet Engineering Task Force

IGTF International Grid Trust Federation

IMS Identitäts-Management-Systeme

Inter-FIM Inter-Federation Identity Management

IP Internet Protocol

IPV Identity Proofing

ISO International Organization for Standardization

IT Informationstechnologie

ITIL Information Technology Infrastructure Library

ITSM IT Service Management

JSON JavaScript Object Notation

JSP Java Server Pages

JWT JSON Web Token

LDAP	Lightweight Directory Access Protocol
LMU	Ludwig-Maximilians-Universität
LoA	Level of Assurance
LoT	Level of Trust
LRZ	Leibniz-Rechenzentrum
MACE-Dir	Middleware Architecture Committee for Education – Directory
MB	Megabyte
MetaIOP	SAML V2.0 Metadata Interoperability Profile
MdFIM	Management of dynamic Federated Identity Management
MDS	Metadata Distribution Service
MLA	Maturity Level Assertion
MLAuth	Maturity Level Authentication
MLD	Maturity Level Data
MLI	Maturity Level Identification
MLO	Maturity Level Organizational
MLT	Maturity Level Technical
MNM	Munich Network Management
MWN	Münchner Wissenschaftsnetz
NIST	National Institute of Standards and Technology
NISTIR	NIST Internal/Interagency Reports
NREN	National Research and Education Network
o	organizationName
OASIS	Organization for the Advancement of Structured Information Standards
OID	Object Identifier

OMB Federal Office of Management and Budget

OP OpenID Provider

OSI Open Systems Interconnection

PAPE Provider Authentication Policy Extension

PDF Portable Document Format

PEER Public Endpoint Entities Registry

PET Privacy Enhancing Technology

PHP Hypertext Preprocessor

PKI Public-Key-Infrastruktur

pyFF python Federation Feeder

QAA Quality Authentication Assurance

QoS Quality of Service

RA Registration Authority

R&E Research & Education

REFEDS Research and Education Federations

regex regular expression

RFC Request for Comments

PIN Personal Identification Number

RM Relationship Manager

RP Relying Party

RR Resource Registry

SAML Security Assertion Markup Language

SAML2int Interoperable SAML 2.0 Web Browser SSO Deployment Profile

SD Service Desk

SESAM Federated Identity Management Service Standards for Automotive

SIM Secure Identity Management

SSp SimpleSAMLphp

SLA Service Level Agreement

SOAP Simple Object Access Protocol

SP Service Provider

SP-A SP Administrator

SP-Federation Service Provider Federation

SPM SP Manager

SP-RM SP Relationship Manager

SP-SD SP Service Desk

SSL Secure Sockets Layer

SSO Single Sign On

SSTC Security Service Technical Committee

STORK Secure identity across borders linked

SQL Structured Query Language

SWAMID Swedish Academic Identity

TAL Trust Anchor List

TBAC Trust Based Access Control

TCP Transmission Control Protocol

TLS Transport Layer Security

TPM Trusted Platform Module

TS Technical Support

TSP Trust Service Provider

TTP	Trusted Third Party
TUM	Technische Universität München
UCIM	User Centric Identity Management
UMA	User-Managed Access
UML	Unified Modeling Language
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
URN	Uniform Resource Name
UUID	Universally Unique Identifier
VM	Virtuelle Maschine
VO	Virtuelle Organisation
VOMS	Virtual Organisation Management System
VoT	Vectors of Trust
WAYF	Where Are You From
WS	Web Services
WSGI	Web Service Gateway Interface
XML	Extensible Markup Language
XSLT	Extensible Stylesheet Language Transformations
XQuery	XML Query Language
ZF	Zahnradfabrik

Anhang A

Listingsverzeichnis

Tabellenverzeichnis

Index

A

B

C

D

V

W

Anhang A

Literaturverzeichnis

[AAMD09] Florina Almenárez, Patricia Arias, Andrés Marín und Daniel Díaz: *Towards Dynamic Trust Establishment for Identity Federation.* In: *Proceedings of the 2009 Euro American Conference on Telematics and Information Systems: New Opportunities to Increase Digital Citizenship*, EATIS '09, Seiten 25:1–25:4. ACM, 2009, ISBN 978-1-60558-398-3. (Zitiert auf Seite 192.)

[ACAGMM13] Patricia Arias Cabarcos, Florina Almenárez, Félix Gómez Mármol und Andrés Marín: *To Federate or Not To Federate: A Reputation-Based Mechanism to Dynamize Cooperation in Identity Management.* Wireless Personal Communications, Seiten 1–18, 2013, ISSN 0929-6212. `http://dx.doi.org/10.1007/s11277-013-1338-y`. (Zitiert auf Seite 192.)

[AS11] Gail Joon Ahn und Pradeep Sekar: *Ontology-Based Risk Evaluation in User-Centric Identity Management.* In: *Communications (ICC), 2011 IEEE International Conference on*, Seiten 1–5. IEEE Computer Society, Juni 2011. (Zitiert auf Seite 199.)

[BDN+13] William E. Burr, Donna F. Dodson, Elaine M. Newton, Ray A. Perlner, W. Timothy Polk, Sarbari Gupta und Emad A. Nabbus: *NIST Special Publication 800-63-2 – Electronic Authentication Guideline.* Technischer Bericht, Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology, 2013. (Zitiert auf den Seiten 9 und 205.)

[BFS04] Elisa Bertino, Elena Ferrari und Anna Squicciarini: *Trust Negotiations: Concepts, Systems, and Languages.* Computing in Science Engineering, 6(4):27–34, 2004. (Zitiert auf Seite 345.)

[BGS05] Bettina Berendt, Oliver Günther und Sarah Spiekermann: *Privacy in E-Commerce: Stated Preferences vs. Actual Behavior.* Commun. ACM, 48(4):101–106, April 2005, ISSN 0001-0782. (Zitiert auf Seite 102.)

[BIT08] BITKOM: *Schriftreihe Recht & Steuer – Band 2 – Übermittlung personenbezogener Daten – Inland, EU-Länder, Drittländer.* [Online, abgerufen am

13.05,2016], 2008. (Zitiert auf Seite 55.)

[BN15] Sean Brooks und Ellen Nadeau: *NISTIR 8062 (Draft) – Privacy Risk Management for Federal Information Systems.* Technischer Bericht, Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology, 2015. (Zitiert auf Seite 425.)

[Boa06] The Government Information Security Management Board: *Principles and good practices for identity management – VAHTI 9/2006.* Technischer Bericht, Ministry of Finance, Finnland, 2006. (Zitiert auf Seite 383.)

[Bol03] Joshua B. Bolten: *E-Authentication Guidance for Federal Agencies.* Technischer Bericht, Executive Office of the President – Office of Management and Budget, 2003. (Zitiert auf Seite 205.)

[Bou09] Latifa Boursas: *Trust-Based Access Control in Federated Environments.* Dissertation, Technische Universität München, 2009. (Zitiert auf den Seiten 19, 33, 190, 232, 347, 410, 428 und 432.)

[BPM13] Makarand V. Bhonsle, Nayot Poolsappasit und Sanjay Kumar Madria: *ETIS – Efficient Trust and Identity Management System for Federated Service Providers.* In: *Advanced Information Networking and Applications (AINA), 2013 IEEE 27th International Conference on*, Seiten 219–226. IEEE Computer Society, März 2013. (Zitiert auf Seite 195.)

[Bra13] Tim Bray: *Account Chooser API – Draft.* Technischer Bericht, OpenID Foundation, 2013. (Zitiert auf Seite 153.)

[BSSB07] Abhilasha Bhargav-Spantzel, Anna Cinzia Squicciarini und Elisa Bertino: *Trust Negotiation in Identity Management.* Security Privacy, IEEE, 5(2):55–63, 2007, ISSN 1540-7993. `http://dx.doi.org/10.1109/MSP.2007.46`. (Zitiert auf Seite 192.)

[CHK+05] Scott Cantor, Frederick Hirsch, John Kemp, Rob Philpott und Eve Maler: *Bindings for the OASIS Security Assertion Markup Language (SAML) V2.0.* Technischer Bericht, OASIS, 2005. (Zitiert auf den Seiten 139, 142 und 277.)

[CKPM05] Scott Cantor, John Kemp, Rob Philpott und Eve Maler: *Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0.* Technischer Bericht, OASIS, 2005. (Zitiert auf den Seiten 3, 139, 142 und 171.)

[CLA16] CLARIN-D: *Benutzerhandbuch.* `http://de.clarin.eu/de/hilfe/benutzerhandbuch`, 2016. [Online, abgerufen am 13.05.2016]. (Zitiert auf Seite 73.)

[CLGS04] Óscar Cánovas, Gabriel López und Antonio F. Gómez-Skarmeta: *A Credential Conversion Service for SAML-based Scenarios.* In: Sokratis K. Katsikas, Stefanos Gritzalis und Javier Lopez (Herausgeber): *Public Key Infrastructure*, Band 3093 der Reihe *Lecture Notes in Computer Science*, Seiten 297–305. Springer Berlin Heidelberg, 2004. `http://dx.doi.org/10.1007/978-3-540-25980-0_24`. (Zitiert auf Seite 199.)

[CMPM05] Scott Cantor, Jahan Moreh, Rob Philpott und Eve Maler: *Metadata for the OASIS Security Assertion Markup Language (SAML) V2.0.* Technischer Bericht, OASIS, 2005. (Zitiert auf den Seiten 42, 139, 144, 405, 406 und 407.)

[CS12] Scott Cantor und Thomas Scavo: *SAML IdP Proxy.* `https://spaces.internet2.edu/display/GS/SAMLIdPProxy`, 2012. [Online, abgerufen am 13.05.2016]. (Zitiert auf den Seiten 185 und 349.)

[DFN10] DFN-Verein: *Technische und organisatorische Voraussetzungen an das Identity Management.* `https://www.aai.dfn.de/der-dienst/identitymanagement/`, 2010. [Online; abgerufen am 13.05.2016]. (Zitiert auf Seite 29.)

[DFN15a] DFN-AAI: *DFN-AAI: Verlässlichkeitsklassen.* `https://www.aai.dfn.de/der-dienst/verlaesslichkeitsklassen/`, 2015. [Online, abgerufen am 13.05.2016]. (Zitiert auf den Seiten 9 und 58.)

[DFN15b] DFN-Verein: *Attribute in der DFN-AAI.* `https://www.aai.dfn.de/der-dienst/attribute`, 2015. [Online; abgerufen am 13.05.2016]. (Zitiert auf den Seiten 3 und 60.)

[ELBA07] Christian Emig, Kim Langer, Jürgen Biermann und Sebastian Abeck: *Semantic Integration of Identity Data Repositories.* In: Torsten Braun, Georg Carle und Burkhard Stiller (Herausgeber): *Kommunikation in Verteilten Systemen (KiVS)*, Informatik aktuell, Seiten 101–112. Springer Berlin Heidelberg, 2007. `http://dx.doi.org/10.1007/978-3-540-69962-0_9`. (Zitiert auf Seite 199.)

[ER95] Europäisches Parlament und Rat der Europäischen Union: *EUR-Lex – 31995L0046.* Technischer Bericht, EU, 1995. [Online, abgerufen am 13.05.2016]. (Zitiert auf den Seiten 54 und 55.)

[ER14] Europäisches Parlament und Rat der Europäischen Union: *EUR-Lex – 32014R0910.* Technischer Bericht, EU, 2014. [Online, abgerufen am 13.05.2016]. (Zitiert auf Seite 212.)

[FP13] Md.Sadek Ferdous und Ron Poet: *Dynamic Identity Federation Using Security Assertion Markup Language (SAML).* In: Simone Fischer-Hübner, Elisabeth de Leeuw und Chris Mitchell (Herausgeber): *Policies and Rese-*

arch in Identity Management, Band 396 der Reihe *IFIP Advances in Information and Communication Technology*, Seiten 131–146. Springer Berlin Heidelberg, 2013. `http://dx.doi.org/10.1007/978-3-642-37282-7_13`. (Zitiert auf Seite 193.)

[Gak14] GakuNin, `https://meatwiki.nii.ac.jp/confluence/display/GakuNinShibInstall/uApprove+Jet+Pack+2.5.0+user+manual`: *uApprove Jet Pack 2.5.0 User Manual*, 2014. [Online, abgerufen am 13.05.2016]. (Zitiert auf den Seiten 168 und 512.)

[GÉA16a] GÉANT: *About GÉANT*. `http://www.geant.org/About`, 2016. [Online; abgerufen am 13.05.2016]. (Zitiert auf Seite 1.)

[GÉA16b] GÉANT: *eduGAIN technical site*. `https://technical.edugain.org/status.php`, 2016. [Online; abgerufen am 13.05.2016]. (Zitiert auf den Seiten 4 und 6.)

[GÉA16c] GÉANT: *mds.edugain.org*. `http://mds.edugain.org/`, 2016. [Online; abgerufen am 13.05.2016]. (Zitiert auf Seite 5.)

[GHH+01] Markus Garschhammer, Rainer Hauck, Heinz Gerd Hegering, Bernhard Kempter, Michael Langer, Michael Nerb, Igor Radisic, Harald Roelle und Holger Schmidt: *Towards generic Service Management Concepts – A Service Model Based Approach*. In: *Proceedings of the 7th International IFIP/IEEE Symposium on Integrated Management (IM 2001)*, Seiten 719–732. IEEE Computer Society, Mai 2001. (Zitiert auf den Seiten 35, 37, 241 und 347.)

[GHH+02] Markus Garschhammer, Rainer Hauck, Heinz Gerd Hegering, Bernhard Kempter, Igor Radisic, Harald Roelle und Holger Schmidt: *A Case-Driven Methodology for Applying the MNM Service Model*. In: *Proceedings of the 8th International IFIP/IEEE Network Operations and Management Symposium (NOMS 2002)*, IFIP/IEEE, Seiten 697–710. IEEE Computer Society, April 2002. (Zitiert auf Seite 294.)

[GHK+01] Markus Garschhammer, Rainer Hauck, Bernhard Kempter, Igor Radisic, Harald Roelle und Holger Schmidt: *The MNM Service Model – Refined Views on Generic Service Management*. Journal of Communications and Networks, 3(4):297–306, 2001, ISSN 1229-2370. (Zitiert auf den Seiten 35 und 347.)

[GHMP15] Michael Grabatin, Wolfgang Hommel, Stefan Metzger und Daniela Pöhn: *DAME: On-demand Internet-scale SAML Metadata Exchange*. International Journal On Advances in Systems and Measurements, 8:156–167, 2015, ISSN 1942-261X. (Zitiert auf den Seiten 16, 472, 473, 474, 477, 513 und 530.)

[GHMP16] Michael Grabatin, Wolfgang Hommel, Stefan Metzger und Daniela Pöhn: *Improving the Scalability of Identity Federations through Level of Assurance*

Management Automation. In: *9. DFN-Forum Kommunikationstechnologien*. Gesellschaft für Informatik, Bonn, Mai 2016. (Zitiert auf den Seiten 17 und 412.)

[GHN09] James Governor, Dion Hinchcliffe und Duane Nickull: *What entrepreneurs and information architects need to know*. OReilly, 2009. (Zitiert auf Seite 103.)

[Gra14] Michael Grabatin: *Identity Management für dynamische virtuelle Föderationen unter Verwendung einer Trusted Third Party*. Masterarbeit, Ludwig-Maximilians-Universität München, 2014. (Zitiert auf den Seiten 18, 19, 340, 345, 388, 451, 454, 459, 460, 463, 513, 526 und 527.)

[GZAM+12] Eghbal Ghazizadeh, Mazdak Zamani, Jamalul lail Ab Manan, Reza Khaleghparast und Ali Taherian: *A trust based model for federated identity architecture to mitigate identity theft*. In: *Internet Technology And Secured Transactions, 2012 International Conference for*, Seiten 376–381. IEEE Computer Society, Dezember 2012. (Zitiert auf Seite 195.)

[Häm06] Lukas Hämmerle: *SWITCHaai: Shibboleth-based Federated Identity Management in Switzerland*. In: *Proc. CESNET 2006 Conference*, 2006. (Zitiert auf Seite 184.)

[HAN99] Heinz Gerd Hegering, Sebastian Abeck und Bernhard Neumair: *Integriertes Management vernetzter Systeme – Konzepte, Architekturen und deren betrieblicher Einsatz*. dpunkt–Verlag, 1999. (Zitiert auf den Seiten 35, 41, 241, 276, 280, 297 und 347.)

[Har12] Dick Hardt: *The OAuth 2.0 Authorization Framework*. RFC 6749, RFC Editor, Oktober 2012. `http://www.rfc-editor.org/rfc/rfc6749.txt`. (Zitiert auf Seite 150.)

[HCH+05] John Hughes, Scott Cantor, Jeff Hodges, Frederick Hirsch, Prateek Mishra, Rob Philpott und Eve Maler: *Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0*. Technischer Bericht, OASIS, 2005. (Zitiert auf den Seiten 43, 139, 143, 146 und 355.)

[Hed11] Roland Hedberg: *Configuration of pySAML2 entities*. Documentation, Roland Hedberg, 2011. (Zitiert auf den Seiten 176, 178, 179 und 525.)

[HJK08] Patrick Harding, Leif Johannson und Nate Klingenstein (Herausgeber): *Dynamic Security Assertion Markup Language: Simplifying Single Sign-On*, IEEE Security & Privacy, vol. 6, no. 2. IEEE Computer Society, 2008. (Zitiert auf Seite 160.)

[HKP+08] Wolfgang Hommel, Silvia Knittl, Daniel Pluta, Latifa Boursas und

Ralf Ebner (Herausgeber): *Hochschulübergreifend integriertes Identitäts-Management am Beispiel des Münchner Wissenschaftsnetzes*, IIM2008 – Integriertes Informationsmanagement an Hochschulen, Workshop im Rahmen der 38. Jahrestagung der Gesellschaft für Informatik e.V. (GI). Gesellschaft für Informatik e.V. (GI), 2008. (Zitiert auf Seite 25.)

[HLE09] Bob Hulsebosch, Gabriele Lenzini und Henk Eertink: *D2.3 – Quality authenticator scheme.* Deliverable, STORK-eID Consortium, 2009. (Zitiert auf den Seiten 206, 207 und 512.)

[HMP15] Wolfgang Hommel, Stefan Metzger und Daniela Pöhn: *Dynamic virtual federations with GÉANT-TrustBroker – Closing the gap between NREN federations and eduGAIN.* CONNECT – Special Edition, 18:20–21, 2015. `http://www.geant.net/MediaCentreEvents/CONNECT/Documents/CONNECT_Issue_18_Web.pdf`. (Zitiert auf Seite 16.)

[Hom07] Wolfgang Hommel: *Architektur- und Werkzeugkonzepte für föderiertes Identitäts-Management.* Dissertation, Ludwig-Maximilians-Universität München, 2007. (Zitiert auf den Seiten 19, 22, 25, 26, 46, 200, 223, 281, 297, 299, 313, 315, 316, 317, 338, 339, 479, 489, 493 und 495.)

[Hom13] Wolfgang Hommel: *Response to the GN3plus Open Call for selection of additional beneficiaries (topic 13).* [Projektantrag für GN3plus: Open Call for additional beneficiaries], 2013. (Zitiert auf den Seiten 14 und 15.)

[HP16] Wolfgang Hommel und Daniela Pöhn: *Management Architecture for Dynamic Federated Identity Management.* In: *Computer Science & Information Technology – The Sixth International Conference on Computer Science, Engineering and Information Technology (CCSEIT 2016)*, Seiten 211–226. AIRCC Publishing Corporation, Mai 2016. (Zitiert auf den Seiten 17 und 222.)

[HPM05] Frederick Hirsch, Rob Philpott und Eve Maler: *Security and Privacy Considerations for the OASIS Security Assertion Markup Language (SAML) V2.0.* Technischer Bericht, OASIS, 2005. (Zitiert auf den Seiten 297 und 300.)

[InC13] InCommon: *Identity Assurance Profiles Bronze and Silver – Version 1.2.* Technischer Bericht, InCommon, 2013. [Online, abgerufen am 13.05.2016]. (Zitiert auf den Seiten 4 und 201.)

[InC16] InCommon: *InCommon Certified Identity Providers.* `https://incommon.org/federation/info/all-idps-certified.html`, 2016. [Online, abgerufen am 13.05.2016]. (Zitiert auf Seite 202.)

[Int16] Internet2: *eduPerson Object Class Specification (201310).* Technischer Bericht, Internet2, 2013), howpublished=`http://software.internet2.edu/`

eduperson/internet2-mace-dir-eduperson-201310.html, note ="[Online, abgerufen am 13.05.2016]",. (Zitiert auf Seite 61.)

[ISO13] ISO/IEC: *ISO/IEC 29115:2013 – Entity authentication assurance framework*. Technischer Bericht, ISO/IEC, 2013. (Zitiert auf den Seiten 207 und 208.)

[JDL+11] Jian Jiang, Haixin Duan, Tao Lin, Fenglin Qin und Hong Zhang: *A federated identity management system with centralized trust and unified Single Sign-On*. In: *Communications and Networking in China (CHINACOM), 2011 6th International ICST Conference on*, Seiten 785–789. IEEE, August 2011. (Zitiert auf Seite 194.)

[JFH+05] Audun Jøsang, John Fabre, Brian Hay, James Dalziel und Simon Pope: *Trust Requirements in Identity Management*. In: *Proceedings of the 2005 Australasian Workshop on Grid Computing and e-Research - Volume 44*, ACSW Frontiers '05, Seiten 99–108. Australian Computer Society, Inc., 2005. (Zitiert auf Seite 190.)

[JH12] Michael B. Jones und Dick Hardt: *The OAuth 2.0 Authorization Framework: Bearer Token Usage*. RFC 6750, RFC Editor, Oktober 2012. http://www.rfc-editor.org/rfc/rfc6750.txt. (Zitiert auf Seite 150.)

[Jis16] Jisc: *UK Access Management Federation | Jisc*. http://www.jisc.ac.uk/uk-federation, 2016. [Online; abgerufen am 13.05.2016]. (Zitiert auf Seite 4.)

[Joh12] Leif Johansson: *An IANA Registry for Level of Assurance (LoA) Profiles*. RFC 6711, RFC Editor, August 2012. (Zitiert auf Seite 214.)

[JSJS13] Paul E. Jones, Gonzalo Salgueiro, Michael B. Jones und Joseph Smarr: *WebFinger*. RFC 7033, RFC Editor, September 2013. (Zitiert auf Seite 156.)

[Kal15] Kalmar2: *Front Page – Kalmar2*. https://www.kalmar2.org/kalmar2web/front_page.html, 2015. [Online; abgerufen am 13.05.2016]. (Zitiert auf Seite 4.)

[Kan15] Kantara Initiative: *Home – WG – User Managed Access*. https://kantarainitiative.org/confluence/display/uma/Home, 2015. [Online, abgerufen am 13.05.2016]. (Zitiert auf den Seiten 103, 104 und 511.)

[KCM+05] John Kemp, Scott Cantor, Prateek Mishra, Rob Philpott und Eve Maler: *Authentication Context for the OASIS Security Assertion Markup Language (SAML) V2.0*. Technischer Bericht, OASIS, 2005. (Zitiert auf den Seiten 139 und 322.)

[KHM+10] Nathan Klingenstein, Thomas Hardjono, RL Bob Morgan, Paul Madsen und Scott Cantor: *SAML V2.0 Identity Assurance Profiles Version 1.0*. Technischer Bericht, OASIS, 2010. (Zitiert auf den Seiten 147 und 420.)

[LCC09a] Hal Lockhart, Brian Campbell und Scott Cantor: *SAML V2.0 Metadata Extension for Entity Attributes – Version 1.0*. Technischer Bericht, OASIS, 2009. (Zitiert auf Seite 436.)

[LCC09b] Hal Lockhart, Brian Campbell und Scott Cantor: *SAML V2.0 Metadata Interoperability Profile – Version 1.0*. Technischer Bericht, OASIS, 2009. (Zitiert auf Seite 160.)

[LCWC08] Hal Lockhart, Brian Campbell, Rod Widdowson und Scott Cantor: *Identity Provider Discovery Service Protocol and Profile*. Technischer Bericht, OASIS, 2008. (Zitiert auf den Seiten 146, 171 und 355.)

[LD09] Konstantinos Lampropoulos und Spyros Denazis: *DIMDS: A Dynamic Identity Management and Discovery System*. In: *Proceedings of the INFOCOMP Workshop 2009, IEEE*, Seiten 1–2. IEEE Computer Society, 2009. ISBN: 978-1-4244-3968-3. (Zitiert auf Seite 191.)

[Lin09] Mikael Linden: *Organisational and Cross-Organisational Identity Management*. Dissertation, Tampere University of Technology, 2009. (Zitiert auf den Seiten 20, 28, 50, 185, 349, 383, 417, 418, 511 und 513.)

[Lin13] Mikael Linden: *GÉANT Data Protection Code of Conduct*. Technischer Bericht, 2013. `http://www.geant.net/uri/dataprotection-code-of-conduct/v1/Pages/default.aspx`, [Online, abgerufen am 13.05.2016]. (Zitiert auf Seite 54.)

[LMCRLGS07] Gabriel López Millán, Óscar Cánovas Reverte, Diego R. López und Antonio F. Gómez-Skarmeta: *Extending the Common Services of eduGAIN with a Credential Conversion Service*. In: *Computer Security – ESORICS 2007*, Band 4734 der Reihe *Lecture Notes in Computer Science*, Seiten 501–514. Springer Berlin Heidelberg, 2007. `http://dx.doi.org/10.1007/978-3-540-74835-9_33`. (Zitiert auf Seite 200.)

[LP09] Farah Layouni und Yann Pollet: *Mobile Agents and Their Ontology Serving a Federated Identity Platform*. In: *Systems, 2009. ICONS '09. Fourth International Conference on*, Seiten 1–6. IEEE Computer Society, März 2009. (Zitiert auf Seite 199.)

[LS15] Mikael Linden und Brook Schofield: *eduGAIN Policy Framework Attribute Profile*. Technischer Bericht, GÉANT, 2015. [Online, abgerufen am 13.05.2016]. (Zitiert auf Seite 60.)

[Mar11] Patricia Marcu: *Architekturkonzepte für interorganisationales Fehlermanagement.* Dissertation, Ludwig-Maximilians-Universität München, 2011. (Zitiert auf Seite 240.)

[MDS95] Roger C. Mayer, James H. Davis und F. David Schoorman (Herausgeber): *An integrative model of organizational trust*, The Academy of Management Review, Vol. 20, 709–734. Academy of Management, 1995. (Zitiert auf Seite 191.)

[Mei15] Johannes Meier: *Evaluation und Konzeption von Verlässlichkeitsklassen für den Einsatz in föderierten Umgebungen.* Bachelorarbeit, Ludwig–Maximilians–Universität München, 2015. (Zitiert auf den Seiten 412 und 416.)

[Mic15] Microsoft, `https://technet.microsoft.com/de-de/windowsserver/dd448613`: *Active Directory-Verbunddienste*, 2015. [Online, abgerufen am 13.05.2016]. (Zitiert auf Seite 181.)

[Mik12] Tomi Mikkonen: *Description of the Identity Management of a Haka Home Organization.* Technischer Bericht, CSC, 2012. `https://confluence.csc.fi/display/HAKA/Joining+and+registrations?preview=/34898755/35094749/Haka-self-assessment-1.0.xlsx`. (Zitiert auf den Seiten 202, 203 und 512.)

[MKT05] Paul Madsen, Yuzo Koga und Kenji Takahashi (Herausgeber): *Federated Identity Management for Protecting Users from ID Theft*, Proceedings of Digital Identity Management '05. ACM, 2005. (Zitiert auf Seite 190.)

[Ode09] Odette: *ODETTE SESAM specification for building up federated Single-Sign-On (SSO) scenarios between companies in the automotive sector – Draft of 15.07.2009.* Technischer Bericht, Odette, 2009. (Zitiert auf Seite 95.)

[OYN+12] Tananun Orawiwattanakul, Kazutsuna Yamaji, Motonori Nakamura, Toshiyuki Kataoka und Noburo Sonehara (Herausgeber): *uApprove.jp - User Consent Acquisition System For Japanese Federation (GakuNin)*, Band 2012 der Reihe *TNC*. TERENA, 2012. (Zitiert auf den Seiten 44 und 169.)

[PGM+15] Daniela Pöhn, Michael Grabatin, Stefan Metzger, David Schmitz und Wolfgang Hommel: *Deliverable OCJ-DS4.1.1 Open Call Deliverable - GÉANT-TrustBroker implementation with documentation.* Technischer Bericht, 2015. (Zitiert auf den Seiten 18 und 451.)

[PH16] Daniela Pöhn und Wolfgang Hommel (Herausgeber): *Automated User Information Conversion to improve Identity Federation Scalability*, 22th congress of the European University Information Systems Organisation (EUNIS 2016), 2016. (Zitiert auf den Seiten 17 und 388.)

[PMH13a] Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *Milestone M.1.1.1: Requirements analysis of Géant-TrustBroker.* Technischer Bericht, 2013. (Zitiert auf Seite 17.)

[PMH13b] Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *Milestone M.1.2.1: Géant-TrustBroker standardisation roadmap.* Technischer Bericht, 2013. (Zitiert auf Seite 17.)

[PMH14a] Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *A SAML Metadata Broker for Dynamic Federations and Inter-Federations.* In: *Proceedings of INFOCOMP 2014, The Fourth International Conference on Advanced Communications and Computation*, Seiten 132–137. IARIA, 2014. ISBN: 978-1-61208-365-0. (Zitiert auf den Seiten 15, 340 und 388.)

[PMH14b] Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *Géant-TrustBroker: Dynamic, Scalable Management of SAML-Based Inter-federation Authentication and Authorization Infrastructures.* In: Nora Cuppens-Boulahia, Frédéric Cuppens, Sushil Jajodia, Anas Abou El Kalam und Thierry Sans (Herausgeber): *ICT Systems Security and Privacy Protection*, Band 428 der Reihe *IFIP Advances in Information and Communication Technology*, Seiten 307–320. Springer Berlin Heidelberg, 2014. `http://dx.doi.org/10.1007/978-3-642-55415-5_25`. (Zitiert auf den Seiten 15, 340, 345 und 388.)

[PMH14c] Daniela Pöhn, Stefan Metzger und Wolfgang Hommel (Herausgeber): *Géant-TrustBroker: Simplifying Identity & Access Management for International Research Projects and Higher Education Communities*, 20th congress of the European University Information Systems Organisation (EUNIS 2014), 2014. (Zitiert auf den Seiten 16, 340, 345 und 388.)

[PMH14d] Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *Milestone Document M.2.1.1: GÉANT-TrustBroker protocol specification written.* Technischer Bericht, 2014. (Zitiert auf Seite 17.)

[PMH14e] Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *Milestone M.4.1.1: TrustBroker service demonstrator.* Technischer Bericht, 2014. (Zitiert auf den Seiten 18, 451, 452 und 513.)

[PMH14f] Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *Open Call Project Deliverable D.2.1.1: Géant-TrustBroker Specification.* Technischer Bericht, 2014. (Zitiert auf Seite 18.)

[PMH14g] Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *Project GÉANT-TrustBroker – dynamic identity management across federation borders.* In: Erik Huizer (Herausgeber): *Networking with the World, The 30th Trans European Research and Education Networking Conference, 19-22 May 2014,*

Selected Papers. TERENA, August 2014, ISBN 978-90-77559-24-6. `http://www.terena.org/publications/tnc2014-proceedings/`. (Zitiert auf den Seiten 16, 340, 370 und 388.)

[PMH15] Daniela Pöhn, Stefan Metzger und Wolfgang Hommel: *Deliverable OCJ DS2.2.1 Open Call Deliverable GÉANT-TrustBroker protocol specification*. Technischer Bericht, 2015. (Zitiert auf Seite 18.)

[Pöh15] Daniela Pöhn: *Topology of Dynamic Metadata Exchange via a Trusted Third Party*. In: *GI-Edition 251 – Open Identity Summit 2015*, Seiten 103–118. Gesellschaft für Informatik, Bonn, November 2015. (Zitiert auf den Seiten 16 und 503.)

[Pöh16a] Daniela Pöhn: *Architecture and Concepts for Federated Identity Management with Federations and Inter-federations*. In: *DCISSP 2016 - Doctoral Consortium*, Seiten 3–9. INSTINCC, Februar 2016. (Zitiert auf Seite 16.)

[Pöh16b] Daniela Pöhn: *Risk Management for Dynamic Metadata Exchange via a Trusted Third Party*. In: *ICISSP 2016 - Proceedings of the 2nd International Conference on Information Systems Security and Privacy*, Seiten 227–234. SciTePress, Februar 2016. (Zitiert auf den Seiten 16 und 297.)

[PvWP16] Remco Poortinga-van Wijnen und Daniela Pöhn: *Deliverable D15.3 Operational GÉANT Trust Broker Pilot Instance*. Technischer Bericht, 2016. (Zitiert auf den Seiten 18 und 451.)

[Rei08] Helmut Reiser: *Ein Framework für föderiertes Sicherheitsmanagement*. Habilitation, Ludwig-Maximilians-Universität München, 2008. (Zitiert auf den Seiten 19, 85 und 328.)

[RHPM08] Nick Ragouzis, John Hughes, Rob Philpott und Eve Maler: *Security Assertion Markup Language (SAML) V2.0 Technical Overview*. Technischer Bericht, OASIS, 2008. (Zitiert auf den Seiten 43 und 146.)

[Ric13] Christian Richter: *Reifegradmodelle für Werkzeuglandschaften zur Unterstützung von ITSM-Prozessen*. Dissertation, 2013. (Zitiert auf Seite 415.)

[RJ15] Justin Richer und Leif Johansson: *Vectors of Trust*. Internet-Draft draft-richer-vectors-of-trust-02, IETF Secretariat, 2015. `http://www.ietf.org/internet-drafts/draft-richer-vectors-of-trust-02.txt`. (Zitiert auf Seite 210.)

[RJB+08] David Recordon, Michael B. Jones, Johnny Bufu, Jonathan Daughtery und Nat Sakimura: *OpenID Provider Authentication Policy Extension 1.0*. Technischer Bericht, OpenID Foundation, 2008. (Zitiert auf Seite 158.)

[Rod02] Gabi Dreo Rodosek: *A Framework for IT Service Management*. Habilitation, Ludwig-Maximilians-Universität München, 2002. (Zitiert auf den Seiten 35 und 37.)

[SBJ+14] Nat Sakimura, John Bradley, Michael B. Jones, Breno de Medeiros und Chuck Mortimore: *OpenID Connect Core 1.0*. Technischer Bericht, OpenID Foundation, 2014. (Zitiert auf Seite 151.)

[SBJJ14] Nat Sakimura, John Bradley, Michael B. Jones und Edmund Jay: *OpenID Connect Discovery 1.0*. OpenID Specification, OpenID Foundation, 2014. (Zitiert auf den Seiten 155, 156, 157 und 525.)

[Sch07] Michael Schiffers: *Management dynamischer Virtueller Organisationen in Grids*. Dissertation, Ludwig-Maximilians-Universität München, 2007. (Zitiert auf den Seiten 20, 30 und 84.)

[SCM+15] Andreas Akre Solberg, Scott Cantor, Eve Maler, Leif Johansson, Jeff Hodges, Ian Young, Nate Klingenstein und Bob Morgan: *Interoperable SAML 2.0 Web Browser SSO Deployment Profile*. `http://saml2int.org/profile/current`, 2015. [Online, abgerufen am 13.05.2016]. (Zitiert auf Seite 160.)

[SG06] Rajarajan Sampath und Deepak Goel: *RATING: Rigorous Assessment of Trust in Identity Management*. In: *Proceedings of the First International Conference on Availability, Reliability and Security*, ARES 06, Seiten 14–23. IEEE Computer Society, 2006. (Zitiert auf Seite 410.)

[Shi15] Shibboleth: *Shibboleth*. `http://shibboleth.net/`, 2015. [Online, abgerufen am 13.05.2016]. (Zitiert auf den Seiten 43 und 161.)

[SWI14] SWITCHaai, `https://www.switch.ch/aai/downloads/uApprove-manual/`: *uApprove Manual*, 2014. [Online, abgerufen am 13.05.2016]. (Zitiert auf den Seiten 44, 64 und 167.)

[tC02] 107th Congress: *Public Law 107–204—July 30, 2002*. Technischer Bericht, Congress, USA, 2002. (Zitiert auf Seite 383.)

[Ter16] Terena: *Metadata Explorer Tool*. `http://met.refeds.org/`, 2016. [Online; abgerufen am 13.05.2016]. (Zitiert auf den Seiten 4, 28 und 49.)

[UNI14] UNINETT: *SimpleSAMLphp*. `http://simplesamlphp.org/`, 2014. [Online; abgerufen am 13.05.2016]. (Zitiert auf den Seiten 43 und 171.)

[U.S16] U.S. Department of Health & Human Services: *National Institutes of Health (NIH)*. `http://www.nih.gov/`, 2016. [Online; abgerufen am 13.05.2016]. (Zitiert auf Seite 4.)

[vL15] Lucas van Lierop. `https://github.com/janus-ssp/janus`, 2015. [Online, abgerufen am 13.05.2016]. (Zitiert auf Seite 185.)

[YLJ09] Ian A. Young und Chad La Joie: *Interfederation and Metadata Exchange: Concepts and Methods*. [Online; abgerufen am 13.05.2016], Mai 2009. (Zitiert auf den Seiten 4 und 188.)

[You15] Ian A. Young: *Metadata Query Protocol*. Internet-Draft draft-young-md-query-05, IETF Secretariat, 2015. `http://www.ietf.org/internet-drafts/draft-young-md-query-05.txt`. (Zitiert auf Seite 187.)