



**TECHNISCHE
UNIVERSITÄT
DRESDEN**

Fakultät Informatik & IBM Lehrstuhl Rechnernetze & Centers for Solution Innovation :: Hamburg

TECHNOLOGISCHE ANALYSEN IM UMFELD SOZIALER NETZWERKE

Peter Schnitzler



TECHNISCHE UNIVERSITÄT DRESDEN
Fakultät Informatik
Institut für Systemarchitektur
Professur Rechnernetze

Prof. Dr. rer. nat. habil. Dr. h. c. Alexander Schill



IBM DEUTSCHLAND
Global Business Services
Centers for Solution Innovation
Hamburg

IBM Interactive

DIPLOMARBEIT

Technologische Analysen im Umfeld Sozialer Netzwerke

zum
Erlangen des akademischen Grades

Diplom-Medieninformatiker

Eingereicht von:	Peter Schnitzler peter@schnitzlers.de http://www.peterschnitzler.de
Verantwortlicher Hochschullehrer:	Prof. Dr. rer. nat. habil. Dr. h. c. Alexander Schill schill@rn.inf.tu-dresden.de http://rn.inf.tu-dresden.de
Betreuer TU Dresden:	Dipl.-Inf. Marius Feldmann feldmann@rn.inf.tu-dresden.de http://rn.inf.tu-dresden.de
Betreuer IBM:	Dipl.-Ing. Silvan Thiele, Dipl.-Designer Tristan Reckhaus { silvan.thiele , tristan.reckhaus }@de.ibm.com http://www.ibm.com/de
Eingereicht am:	31. Oktober 2008



Aufgabenstellung für die Diplomarbeit

Name, Vorname: Schnitzler, Peter

Studiengang: Medieninformatik

Matr.Nr.: | 3 | 0 | 1 | 9 | 0 | 2 | 4 |

Thema: „Technologische Analysen im Umfeld Sozialer Netzwerke“

Beschreibung:

In den vergangenen Jahren ist im World Wide Web eine Vielzahl unterschiedlicher Softwaresysteme entstanden, die der Kommunikation und Zusammenarbeit dienen. Diese Sozialen Netzwerke differenzieren sich unter anderem in Möglichkeiten, die sie zum Zugriff auf intern verwaltete Datenbestände bieten. Zentrales Ziel dieser Arbeit ist es, nach einer Evaluierung geeigneter Technologien für die Realisierung von sozialen Netzwerken, ein prototypisches System für ein Soziales Netzwerk zu schaffen, das Datenbestände aus verschiedenen Quellsystemen zusammenführt und in sinnvoller Weise für eine Verwendung aufbereitet.

In einem ersten Schritt sind dazu verbreitete Soziale Netzwerke auf ihren Funktionsumfang und angebotene Schnittstellen hin zu analysieren. Zusätzlich soll dabei überprüft werden, auf welche Weise Datenbestände, die nicht über eine offene Schnittstelle verfügbar sind, verwendet werden können. Resultat dieses Schrittes ist zum einen eine Aufstellung von Möglichkeiten des Datenimports ausgehend von den betrachteten Systemen, der sowohl temporärer und dynamischer als auch persistenter Natur sein kann und zum anderen eine Zusammenstellung von Funktionalitäten, die das zu schaffende Soziale Netzwerk aufsetzend auf den zusammengetragenen Datenbeständen anbieten soll. Bei der Auswahl der Funktionalitäten ist sowohl aus einer Perspektive technischer Machbarkeit als auch von möglichen Nutzerbedürfnissen auszugehen. Nach der Analyse, wie Informationsquellen gebündelt werden können, ist zu untersuchen, wie diese aggregierten Informationen in strukturierter Weise nach außen über eine zu definierende Schnittstelle zugänglich gemacht werden können. Dazu ist die Erstellung einer konkreten API in Erwägung zu ziehen. In einem weiteren Schritt soll eine geeignete Technologie zur Implementierung eines Sozialen Netzwerkes ausgewählt werden. Hierbei sind zunächst Optionen aufzuführen, die in die Betrachtung einbezogen werden. Diese sind anhand einer aufgestellten Metrik zu evaluieren. Bei dieser Metrik ist von vorliegenden Bewertungsmatrizen auszugehen, die in geeigneter Weise zu adaptieren sind. Ausgehend von den gewonnenen Erkenntnissen ist eine Auswahl von Technologien zu treffen, die zur Umsetzung des Prototypen verwendet werden soll. Daraufhin ist die Software für ein Soziales Netzwerk, das die in der Analyse zusammengetragenen Möglichkeiten bietet, prototypisch zu implementieren. Am Ende der Arbeit ist das Resultat in geeigneter Weise zu bewerten und ist ein mögliches Verbesserungskonzept darzustellen.

Zentrale Ergebnisse:

- Überblick zum Funktionsumfang Sozialer Netzwerke und Möglichkeiten der Datenintegration aus diesen Systemen
- Evaluierung geeigneter Technologien zur Umsetzung eines sozialen Netzwerkes anhand einer Bewertungsmatrix
- Prototypische Implementierung eines Sozialen Netzwerkes, das eine Aggregation unterschiedlicher Datenquellen umsetzt
- Evaluierung der Lösung samt Angabe von Möglichkeiten zur Weiterentwicklung

Betreuer: Dipl.-Inf. Marius Feldmann

**Verantwortlicher
Hochschullehrer:** Prof. Dr. rer. nat. habil. Dr. h. c. Alexander Schill

Institut: Systemarchitektur

Beginn am: 01. Mai 2008

Einzureichen am: 31. Oktober 2008

Unterschrift des verantwortlichen Hochschullehrers

SELBSTÄNDIGKEITSERKLÄRUNG

Hiermit erkläre ich, dass ich die von mir am heutigen Tage dem Prüfungsausschuss der Fakultät Informatik eingereichte Diplomarbeit zum Thema

Technologische Analysen im Umfeld sozialer Netzwerke

vollkommen selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie Zitate kenntlich gemacht habe.

Dresden, den 31.10.2008

Unterschrift

DANKSAGUNG

Viele Menschen haben mittel- und unmittelbar zum Gelingen meines Studiums beigetragen. Für die Hilfe und Unterstützung bei der Studiumswahl, zwei Auslandsaufenthalten, Beschäftigungen als studentische Hilfskraft, ehrenamtlichen Tätigkeiten und zahlreichen Prüfungen, möchte ich mich sehr herzlich bedanken. Auch das Gespräch beim guten Glas Wein war oft sehr hilfreich.

Für die Unterstützung bei dieser Diplomarbeit möchte ich mich besonders bei

- dem Lehrstuhl Rechnernetze, insbesondere meinem Betreuer Marius Feldmann für viele konstruktive Gespräche und Ratschläge
- IBM, insbesondere meinen Betreuern Tristan Reckhaus und Silvan Thiele für die Möglichkeit, diese Arbeit im Industrieumfeld zu schreiben
- meiner Familie für die langjährige Unterstützung in allen Lebenslagen
- meiner Freundin Julia Grauvogel für lange Gespräche und Gelassenheit bei zahlreichen Planänderungen

bedanken.

In Gedenken an meine Mutter Rita Schnitzler.

ZUSAMMENFASSUNG

Die vorliegende Arbeit analysiert die Möglichkeiten einer Daten- und Kontakttaggregation im Umfeld Sozialer Netzwerke. Zunächst wird eine Kategorisierung der wichtigsten Netzwerke und Frameworks vorgenommen. Die Funktionalitäten von acht Sozialen Netzwerken und fünf Frameworks werden anhand einer zuvor entwickelten Evaluierungsmatrix detailliert untersucht. Dabei stehen insbesondere die Funktionalitäten der APIs im Vordergrund. Aufbauend auf den Ergebnissen der Analyse wird ein Prototyp für eine Daten- und Kontakttaggregation konzipiert, implementiert und evaluiert. Abschließend werden Empfehlungen zu den verwendeten Technologien und für die Konzipierungen von zukünftigen Daten- und Kontakttaggregationen im Umfeld Sozialer Netzwerke gegeben.

Stichworte: Soziale Netzwerke, Datenaggregation, Kontakttaggregation, Frameworks, Evaluierung, Matrix

ABSTRACT

This thesis analyses the potential of a data- and contactaggregation in the context of social networks. It provides an overview and categorization of the most important networks and frameworks. The functions of eight social networks and five frameworks are analyzed on the basis of a previously developed matrix of evaluation. Special attention is paid to the features of the APIs. Using the results from the evaluation a prototype is planned, coded and evaluated. Finally, regards for future aggregations between social networks are elaborated.

Keywords: social networks, dataaggregation, contactaggregation, frameworks, evaluation, white label social networks, matrix of evaluation

INHALTSVERZEICHNIS

1	Einleitung	1
1.1	Motivation	2
1.2	Zielstellung und Gliederung	2
2	Grundlagen	5
2.1	Definition des Begriffs „Soziales Netzwerk“	6
2.2	Entstehung von Sozialen Netzwerken	7
2.3	Stand der Forschung	7
2.4	Technologien im Umfeld Sozialer Netzwerke	8
2.4.1	Representational State Transfer (REST)	9
2.4.2	hCard (Mikroformate)	10
2.4.3	XFN	11
2.4.4	Friend of a Friend (FOAF)	12
2.4.5	OpenID	14
2.5	Überblick und Kategorisierung von Sozialen Netzwerken	15
3	Bewertung der Funktionalitäten von Sozialen Netzwerken	19
3.1	Einführung in den Funktionsüberblick	20
3.2	Analyse der Funktionalitäten	20
3.2.1	Accountfunktionalitäten	21

3.2.2	Applikationen und APIs	21
3.2.3	Blog	22
3.2.4	Datenfreigabe	22
3.2.5	Fotos	23
3.2.6	Gästebuch	24
3.2.7	Gruppen	24
3.2.8	Nachrichten	25
3.2.9	Kalender	25
3.2.10	Profil	25
3.2.11	Suche	26
3.2.12	Tagging	26
3.2.13	Videos	27
3.3	Fazit des Funktionsvergleichs	27
4	APIs in Sozialen Netzwerken	29
4.1	Kategorien von APIs	30
4.2	APIs für Applikationen in Sozialen Netzwerken	30
4.2.1	Facebook	31
4.2.2	MySpace	33
4.2.3	Orkut	33
4.2.4	Bebo	34
4.2.5	Hi5	35
4.2.6	OpenSocial	35
4.2.7	Fazit des API-Vergleichs	36
4.3	APIs für die Verknüpfung von Nutzerdaten mit externen Webseiten	37
4.3.1	Google Friend Connect	37
4.3.2	Facebook Connect	37
4.3.3	Data Availability von MySpace	38
4.3.4	Windows Live	38
4.3.5	Yahoo! Address Book API	38
4.3.6	Zusammenfassung	39

5	Frameworks für Soziale Netzwerke	41
5.1	Kategorien von Frameworks	42
5.2	Kriterien für die Auswahl der Frameworks	42
5.3	Auswahl der Frameworks für die Analyse	43
5.4	Kriterien der Analyse	45
5.5	Evaluierung der Frameworks	47
5.5.1	Peopleaggregator	48
5.5.2	Clearspace Community	50
5.5.3	Ning	53
5.5.4	Community Server	57
5.5.5	Lotus Connections	61
5.6	Fazit des Frameworksvergleichs	65
6	Untersuchung und Bewertung der Datenaggregation	69
6.1	Aktueller Stand der Datenaggregation nach Durchführung der Analyse	70
6.2	Aktueller Stand der Visualisierung von Daten nach Durchführung der Analyse	71
6.3	Problemkategorien einer Datenaggregation	72
6.3.1	Definition der Problembereiche und Bewertungsaspekte	72
6.4	Anforderungsanalyse	73
6.5	Konzeption der Funktionalitäten	73
6.5.1	Import	73
6.5.2	Export	74
6.5.3	Definition der zu exportierenden Daten	74
6.5.4	Visualisierung	74
6.6	Implementierung	75
6.6.1	Funktionalitäten für die Problemidentifikation bei einer Datenaggregation	75
6.6.2	Integration in Clearspace	76
6.7	Aggregationsprobleme bei der Implementierung	81
6.8	Allgemeine Aggregationsprobleme und Empfehlungen für die Realisierung von Aggregationen	85
6.9	Zusammenfassung	89

7 Zusammenfassung der Ergebnisse und Ausblick	91
7.1 Fazit	92
7.2 Ausblick	94
A Anhang	97
A.1 Inhalt der DVD	98
A.2 Aktuelle Entwicklungen im Verlauf der Arbeit	99
A.3 Grundlagen	100
A.4 Übersicht über Soziale Netzwerke ohne inhaltlichen Schwerpunkt	100
A.5 Übersicht über Soziale Netzwerke mit inhaltlichem Schwerpunkt	102
A.6 Funktionalitäten von Sozialen Netzwerken	103
A.6.1 Accountfunktionalitäten	103
A.6.2 Applikationen und APIs	106
A.6.3 Blog	107
A.6.4 Datenfreigabe	108
A.6.5 Fotos	109
A.6.6 Gästebuch	110
A.6.7 Gruppen	111
A.6.8 Kalender	113
A.6.9 Nachrichten	114
A.6.10 Profil	115
A.6.11 Suche	116
A.6.12 Tagging	117
A.6.13 Videos	118
A.7 Frameworks für soziale Netzwerke	119
A.7.1 Übersicht über Frameworks	119
A.8 Vergleich und Bewertung der Funktionalitäten	122
A.8.1 Accountfunktionalitäten	122
A.8.2 Applikationen und APIs	124

A.8.3	Blog	126
A.8.4	Datenfreigabe	127
A.8.5	Gruppen (Communities)	128
A.8.6	Implementierung	129
A.8.7	Inhalte (Dokumente, Fotos, Videos)	131
A.8.8	Kalender & Aktionen	132
A.8.9	Kollaboration	133
A.8.10	Nachrichten	134
A.8.11	Profil	135
A.8.12	Suche	136
A.8.13	Tagging	137
A.9	Untersuchung und Bewertung der Datenaggregation	138
A.9.1	Kommunikationsstruktur des Plugins	138
	Literatur- und Quellenverzeichnis	141
	Abkürzungsverzeichnis	153
	Abbildungsverzeichnis	155
	Tabellenverzeichnis	157
	Quellcodeverzeichnis	159

1 EINLEITUNG

In diesem Kapitel wird die Ausgangslage für die Diplomarbeit und die daraus resultierenden Probleme behandelt. Aufbauend darauf wird die Zielstellung dieser Arbeit definiert und ein Überblick über die Gliederung gegeben.

1.1 MOTIVATION

In Folge der Verbreitung von schnellen Internetzugängen, den daraus resultierenden vereinfachten Möglichkeiten für den Nutzer Inhalte beizusteuern und dem Wunsch der Anwender Netzwerke auch online abzubilden hat sich eine große Anzahl an Sozialen Netzwerken entwickelt. In dem Bereich der Sozialen Netzwerke finden sich Anbieter wie Facebook, die unter einem Dach eine sehr breit gefächerte Funktionspalette anbieten, aber auch Dienste wie del.icio.us, die sich auf ein einziges Werkzeug, in diesem Fall das gemeinschaftliche Abspeichern von Lesezeichen, konzentrieren. Berücksichtigt man nur die größeren Netzwerke, existieren bereits über 30 Soziale Netzwerke. Viele Netzwerke haben einen bestimmten Fokus. So ist StudiVZ auf Studenten ausgerichtet, während Facebook insbesondere die Verbreitung über Ländergrenzen hinweg zum Ziel hat. Hinzu kommen Unternehmen und Vereine, die entweder als Intranet oder für bestimmte Zielgruppen eigene Soziale Netzwerke gründen. All dies führt dazu, dass viele Nutzer in mehreren Netzwerken Mitglied sind.

Für die vollständige partizipative¹ Nutzung der Netzwerke ist fast immer eine Registrierung notwendig. Aber auch für die konsum-orientierte² Nutzung, also den Zugriff auf Informationen, die andere Nutzer angelegt haben, ist oft eine Registrierung notwendig. Bei jeder Registrierung müssen zahlreiche Daten über die eigene Person angegeben werden. Damit ein Netzwerk für den Betreiber und die Anwender einen Nutzen hat, müssen die Daten aktuell gehalten werden. Dies bedeutet, dass der Nutzer in zahlreichen Netzwerken über unterschiedliche Eingabemasken die Daten verwalten muss.

Die Möglichkeit, Daten zu im- und exportieren, würde diesen Prozess erheblich vereinfachen. Allerdings werden weder Import noch Export von den Betreibern der Sozialen Netzwerke angeboten. Selbst der Import, von dem die Betreiber und Nutzer gleichermaßen profitieren würden, wird nicht angeboten. Einzig der Import von Kontakten anhand der Emailadresse ist bei vielen Netzwerken möglich. Für Nutzer bedeutet dies zum einen, dass sie keine Möglichkeit haben, auf ihre Daten standardisiert zuzugreifen und zum anderen, dass keinerlei Möglichkeiten bestehen, diese komfortabel zwischen den Netzwerken zu transferieren. Soziale Netzwerke verhalten sich aus Sicht der Anwender äußerst verschlossen und entsprechen damit dem Bild einer abgeschoteten Insel.

1.2 ZIELSTELLUNG UND GLIEDERUNG

Zentrales Ziel dieser Arbeit ist es, aufbauend auf einer Analyse im Umfeld Sozialer Netzwerke eine prototypische Datenaggregation zu realisieren und zu evaluieren.

Ausgangspunkt ist die Entwicklung einer geeigneten Evaluierungsmatrix für die Funktionalitäten von Sozialen Netzwerken. Ziel der Evaluierung ist insbesondere, die Möglichkeiten für den Datenaustausch zu eruieren. In einem weiteren Schritt sollen Softwarelösungen bzw. Frameworks betrachtet werden, die es ermöglichen, ein eigenes Soziales Netzwerke zu betreiben. Dafür soll

¹Hierunter ist eine Nutzung zu verstehen, bei der vom Nutzer Inhalte eingestellt werden.

²Hiermit ist die klassische Nutzung des Internets gemeint, bei der nur Informationen abgerufen werden.

die erstellte Evaluierungsmatrix verwendet werden. Hierbei steht zum einen die Eignung zum Betreiben eines Sozialen Netzwerks und zum anderen die Möglichkeiten für den Datenaustausch im Mittelpunkt. Mit Hilfe dieser Erkenntnisse soll der Prototyp konzipiert, implementiert und evaluiert werden. Hierbei sollen insbesondere Erkenntnisse darüber gewonnen werden, welche Probleme bei einer Datenaggregation auftreten können. Abschließend sollen Empfehlungen für weitere Datenaggregationen im Umfeld Sozialer Netzwerke entwickelt werden.

Um diese Ziele zu erreichen und die Ergebnisse auf eine fundierte Grundlage zu stellen, wurde die Arbeit in die folgenden Punkte gegliedert:

Im Kapitel zwei werden die grundlegenden Technologien im Umfeld Sozialer Netzwerke betrachtet und deren Vor- und Nachteile diskutiert. Nach einer Einführung in die Geschichte von Sozialen Netzwerken und der Betrachtung von anderen Arbeiten auf diesem Gebiet schließt das Kapitel mit einer Kategorisierung von bedeuteten Netzwerken und ermöglicht so einen Überblick über die Bandbreite im Umfeld Sozialer Netzwerke. In Kapitel drei wird eine Evaluierungsmatrix für Funktionalitäten von Sozialen Netzwerken entwickelt und zur Untersuchung bestehender sozialer Netzwerke verwendet. Dabei werden die Funktionen von acht Sozialen Netzwerken verglichen. Kapitel vier untersucht die Möglichkeiten für einen Datenaustausch mit diesen Netzwerken über die angebotenen APIs. In Kapitel fünf werden Frameworks zur Erstellung von Sozialen Netzwerken untersucht und bewertet. Aufbauend auf den Erkenntnissen der Analyse werden in Kapitel sechs die Problembereiche bei einer Datenaggregation eingehend beleuchtet und Lösungsmöglichkeiten für diese aufgezeigt. Um die theoretischen Erkenntnisse zu evaluieren, wurde eine prototypische Implementation vorgenommen, sowie eine Bewertungsmatrix für die Implementierung entwickelt.

2 GRUNDLAGEN

Im folgenden Kapitel werden die Grundlagen für die Arbeit gelegt. Dies umfasst die Definition des Begriffs „Soziales Netzwerk“, eine Zusammenfassung der Geschichte Sozialer Netzwerke sowie einen Überblick über die wichtigsten Technologien im Umfeld Sozialer Netzwerke. Abschließend werden bestehende Soziale Netzwerke kategorisiert und auf Grundlage der vorgenommenen Kategorisierung acht Netzwerke für die weitere Analyse ausgewählt.

2.1 DEFINITION DES BEGRIFFS „SOZIALES NETZWERK“

Der Begriff „Soziales Netzwerk“ wird in unterschiedlichsten Kontexten verwendet. Er kann die Bezeichnung für eine Foren-Community sein, aber auch für ein öffentliches Lesezeichenverwaltungstool verwendet werden. Im allgemeinen Sprachgebrauch bezeichnet er zumeist Webseiten wie studiVZ und Facebook, die viele verschiedene Dienste unter einer Oberfläche anbieten. Um eine einheitliche Begrifflichkeit in dieser Arbeit verwenden zu können, soll in diesem Abschnitt eine Definition für Soziale Netzwerke gefunden werden.

Boyd und Ellison definieren Soziale Netzwerke folgendermaßen:

We define social network sites as web-based services that allow individuals to (1) construct a public or semi-public profile within a bounded system, (2) articulate a list of other users with whom they share a connection, and (3) view and traverse their list of connections and those made by others within the system. The nature and nomenclature of these connections may vary from site to site. (vgl. [16])

Sie legen dabei Ihren Fokus auf die Möglichkeit, im Rahmen eines Sozialen Netzwerks eine Verbindung zu anderen Nutzern aufbauen zu können. Dies ist sicherlich ein wichtiges Merkmal von Sozialen Netzwerken. Die Möglichkeit, Inhalte mit anderen Nutzern zu teilen, darf jedoch nicht außer Acht gelassen werden.

Golbeck ergänzt als wichtigen Aspekt, dass ein Soziales Netzwerk ohne zusätzliche Software erreichbar sein muss (vgl. [58]). Des Weiteren führt sie an, dass Verbindungen nicht nur existieren, sondern außerdem von anderen Nutzern einsehbar sein müssen. Richter und Koch hingegen greifen eine sehr breite Definition von Sixtus auf:

Social Software ist eine Umschreibung für „Softwaresysteme, welche menschliche Kommunikation, Interaktion und Zusammenarbeit unterstützen“ (vgl. [92])

Für die vorliegende Arbeit werden Soziale Netzwerke folgendermaßen definiert:

Als Soziale Netzwerke werden Softwarelösungen definiert, die das Anlegen von Nutzerprofilen, die Gruppierung bzw. das Verbinden von Nutzern und den Zugriff auf gemeinsame Ressourcen browserbasiert ermöglichen.

Im Unterschied zur Definition von Boyd und Ellison umfasst diese Definition auch Soziale Netzwerke, die alternative Verbindungsmöglichkeiten wie Netzwerke und Gruppen anbieten.

2.2 ENTSTEHUNG VON SOZIALEN NETZWERKEN

Zu den bekanntesten Sozialen Netzwerken gehören MySpace, Facebook und im deutschsprachigen Raum studiVZ. Als erstes Soziales Netzwerk gilt das 1997 gegründete Six Degrees (vgl. [16]), welches allerdings 2000 wieder geschlossen wurde, da kein tragfähiges Businessmodell entwickelt wurde. 1999 wurde LiveJournal gegründet und führte als neue Funktion eine Übersicht über die Verbindungen zu anderen Nutzern ein. Als erstes Netzwerk mit dem Fokus auf Geschäftskunden entstand 2001 Ryze.com. 2002 wird Friendster gegründet, welches sich hauptsächlich als Datingplattform verstand. Mit zunehmenden Nutzerzahlen ergaben sich sowohl technische Probleme als auch solche, die das Nutzerverhalten betreffen (z. B. Fake-Accounts). Viele Friendster Nutzer wanderten auf Grund dieser Probleme 2003 zu MySpace ab. Insbesondere die Implementierung von Funktionswünschen machte MySpace 2004 sehr populär. Ins Jahr 2004 fiel außerdem eine Gründungswelle von Sozialen Netzwerken, die auf eine bestimmte Nutzergruppe fokussiert waren (Dogster, Catster). Facebook - im ersten Jahr nur für Harvardstudenten zugänglich - folgt 2004 und der populäre Mikrobloggingdienst „Twitter“ wurde 2006 gegründet. Aus dem Zeitstrahl in Abbildung 2.1 lässt sich erkennen, dass es in den letzten Jahren eine starke Zunahme bei der Gründung von Sozialen Netzwerken gegeben hat.

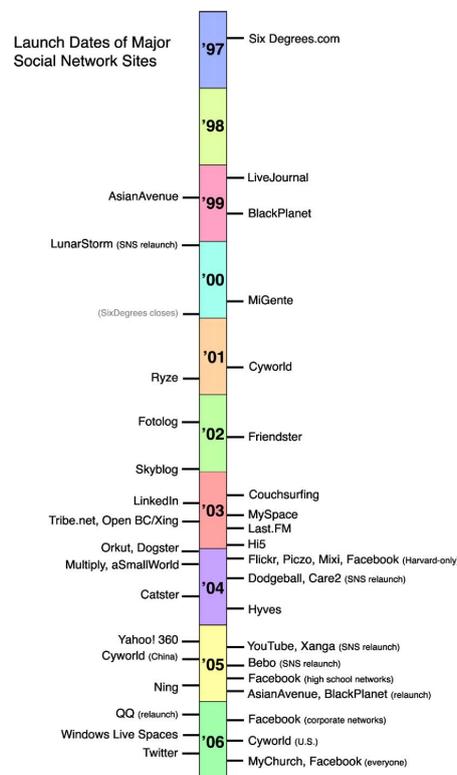


Abb. 2.1: Gründung von Sozialen Netzwerken (vgl. [16])

2.3 STAND DER FORSCHUNG

Im Bereich Sozialer Netzwerke existiert nur wenig Literatur. Insbesondere in den klassischen Publikationsformaten (Bücher und Zeitschriften) ist diese Thematik wenig vertreten. Eine Vielzahl von Informationen findet sich hingegen in Blogbeiträgen und angegliederten Diskussionen. Problematisch ist die Einschätzung der Qualität dieser Informationen. Sie muss für jeden Eintrag individuell erfolgen, da klassische Maßstäbe - wie die Bekanntheit des Journals - nicht angewendet

werden können. Ein bekanntes Instrument der Qualitätseinschätzung - die Häufigkeit, mit der ein Autor zitiert wird - lässt sich jedoch insbesondere auf Blogbeiträge gut übertragen. Im Verlauf der Arbeit wurde deshalb mit einer Mischung aus klassischen Publikationsformen und Informationen aus Blogs (Weblog) und Wikis gearbeitet.

Einen allgemeinen Überblick über das Thema Social Computing und die verschiedenen Ausgestaltungen bietet Manoj Parameswaran (vgl. [129]). Neben einer Abhandlung von typischen Tools im Umfeld von Sozialen Netzwerken liegt der Fokus auf den veränderten Arbeitsabläufen durch Social Computing und den Vor- und Nachteilen.

Alexander Richter und Michael Koch beschäftigen sich in ihrer Veröffentlichung mit den verschiedenen Funktionsgruppen von Sozialen Netzwerken (vgl. [92]). Sie nehmen nur eine sehr grobe Unterteilung in vier Bereiche (Weblogs, Wikis, Social Tagging und Social Networking) vor. Zwar nennen Richter und Koch im weiteren Verlauf auch zahlreiche Beispiele, der Schwerpunkt des Papers liegt aber eher auf der Beschreibung der Funktionalitäten. Dies umfasst auch kurze Einführungen in grundlegende Technologien im Umfeld Sozialer Netzwerke wie „Ajax“ (Asynchronous JavaScript and XML) .

Alan Mislove et al. untersuchen in Ihrer Studie die Struktur Sozialer Netzwerke. Sie konzentrieren sich dabei auf vier Netzwerke (Orkut, Flickr, YouTube und LiveJournal) (vgl. [111]) und kommen dabei zu dem Schluss, dass es in Sozialen Netzwerken sehr viele Clusterings von Nutzern nach Orten gäbe. James Vaerlee und Steve Webb beschäftigen ebenfalls mit der Struktur Sozialer Netzwerke, beschränken sich dabei jedoch auf das Netzwerk MySpace und untersuchen, wie Nutzer sich selber beschreiben, ihre Interessen kommunizieren und im Netzwerk agieren (vgl. [26]).

Ein weiteres Problem im Umfeld Sozialer Netzwerke (Datenschutz und Privatsphäre) betrachten Catherine Dwyer und Starr Roxanne Hiltz (vgl. [36]). Dwyer und Hiltz kommen bei Ihrem Vergleich zu dem Schluss, dass Facebook und MySpace Nutzer gleichermaßen Wert auf ihre Privatsphäre legen, allerdings vertrauten Facebook-Nutzer der Plattform und den Mitgliedern mehr. Dies zeige sich beispielsweise darin, dass Facebook-Nutzer mehr Informationen in Ihren Profilen freigeben.

Jennifer Golbeck führte eine Studie über den Verlauf von Nutzerzahlen bei 13 Netzwerken durch (vgl. [58]) und fand heraus, dass die Mitgliederzahlen linear wachsen, kaum Profile von den Nutzern gelöscht werden und kontinuierlich Verbindungen zwischen Nutzern hinzugefügt werden. Die Anzahl von Nutzern, die „sozial isoliert“ sind, schwankt dabei sehr von Netzwerk zu Netzwerk.

Reisberger et al. entwickeln ein Klassifikationsschema für Soziale Netzwerke (vgl. [135]). Als Grundlage für die Klassifikation dienen sieben Eigenschaften (Ausnutzung der Vernetzung, Zugang zu Kollektivwissen, Entstehung neuer Kontakte, Qualität der Netzwerkbeziehung, Unterstützung der Kommunikation, Anzahl der Beziehungen und Zugangsmöglichkeit). Jede Eigenschaft wird mit einem Funktionsmerkmal und einer Ausprägung versehen. Beispielhaft untersuchen sie mit diesen Funktionsmerkmalen fünf Netzwerke (MySpace, Facebook, Xing, StudiVZ und Flickr). Abschließend entwickeln sie daraus drei Zieldimensionen für Soziale Netzwerke: Kommunikation, Selbstdarstellung und Beziehungen.

2.4 TECHNOLOGIEN IM UMFELD SOZIALER NETZWERKE

Im Umfeld Sozialer Netzwerke spielen eine Reihe von Technologien eine wichtige Rolle. Die wichtigsten Technologien im Umfeld Sozialer Netzwerke werden in diesem Kapitel knapp erläutert, um spätere Entscheidungen für eine Technologie im Rahmen der Implementierung besser nachvollziehen zu können. Untersucht wurden:

- Representational State Transfer (REST)

- HTML vCard (hCard)
- XHTML Friends Network (XFN)
- Friend of a Friend (FOAF)
- OpenID

2.4.1 Representational State Transfer (REST)

Das Representational State Transfer (REST) Prinzip wurde von Roy Thomas Fielding (vgl. [48]) im Jahr 2000 im Rahmen seiner Dissertation entwickelt. REST stellt neben XML-RPC und SOAP eine weitere Möglichkeit dar, um Webservices zu realisieren und findet hauptsächlich in verteilten Netzwerkanwendungen Anwendung. Eine beispielhafte Verwendung von REST könnte folgendermaßen aussehen:

Ein Web Browser fordert eine Seite, oder allgemeiner eine Ressource über eine URL an. Ein HTML Dokument, welches eine Repräsentation der Ressource darstellt, wird vom Server zum Client übertragen. Das HTML Dokument kann Links enthalten, die auf weitere Ressourcen im Web verweisen. Navigiert der Client zu einer neuen Seite, so verändert er seinen Zustand, er wechselt oder macht einen Transfer zu einem neuen Zustand durch. Über Repräsentationen wird ein Transfer von einem Status in einen anderen Status durchgeführt. (vgl. [7])

Entscheidend für die Struktur von REST sind dabei die folgenden Komponenten:

Ressourcen Jedes Objekt (Bilder, Texte usw.) stellt eine Ressource da. Jede Ressource wird dabei mit einer eindeutigen ID versehen. Eine Ressource wird nicht direkt manipuliert, sondern über seine URI mit den im Folgenden genannten Methoden angesprochen.

Repräsentation Jede Ressource hat eine oder mehrere Repräsentationen. Eine Repräsentation selber kann wiederum auf weitere Repräsentationen verweisen.

Methoden Ressourcen und Repräsentationen werden mit Methoden modifiziert. In der Praxis werden meistens die folgenden vier verwendet: „Get“ erfragt die Repräsentation einer Ressource. „Post“ fügt einer bestehenden Ressource etwas weiteres hinzu. „Put“ fügt eine neue Ressource hinzu oder ersetzt eine bestehende komplett. „Delete“ entfernt eine Ressource.

Status und Sessions Implementierungen nach dem REST Architekturstil sind grundsätzlich zustandslos, d.h. eine Anfrage an den Server muss immer alle benötigten Informationen beinhalten und es kann keine Kommunikation über mehrere Nachrichten hinweg aufrechterhalten werden.

Eine weitergehende Beschreibung der verschiedenen Implementierungsmöglichkeiten des REST Architekturstils findet sich bei Richard N. Taylor (vgl. [49]). Auf den Unterschied zwischen der strikten theoretischen Definition und der eigentlich Verwendung geht Stefan Tilkov (vgl. [151]) ein, der insbesondere betont, dass oft kein Unterschied zwischen REST und HTTP gemacht wird. In einem weiteren Dokument führt er acht häufige Fehler bei der Implementierungen von REST auf (vgl. [152]). Dabei verweist er u.a. auf die Verwendung von „GET“, „POST“ Methoden und das

Ignorieren von Cachemöglichkeiten und Status Codes. Grundsätzlich muss zwischen der theoretischen Definition von Fielding und der praktischen Verwendung von REST unterschieden werden. Systeme, die angeben RESTful zu sein, bezeichnen damit in der Regel nicht die vollständige Konformität mit Fieldings Definition. Oft bedeutet dies nur, dass sie die Auslieferung von XML über das HTTP Protokoll ermöglichen.

Neben REST spielt auch das SOAP im Bereich Sozialer Netzwerke eine bedeutende Rolle. Die Unterschiede zeigt Cesare Pautasso auf (vgl. [131]). Auf der Konzeptebene versteht REST das Web als öffentliche Plattform für Dokumente, während SOAP das Web als Medium zum Nachrichtenaustausch verwendet. Konzeptionell ist REST ein auf Ressourcen basierendes Protokoll, wohingegen SOAP auf Operationen basiert. Stand beim Design von REST die Skalierbarkeit und Performance im Vordergrund, geht es bei SOAP um das Entwickeln von integrierten verteilten Anwendungen. SOAP wird für die Abbildung von Geschäftsprozessen verwendet, wohingegen REST eher für die Verknüpfung von Mashups in Betracht gezogen wird. Gegensätzlich ist auch die Implementierung von Zuständen. REST setzt ein zustandloses Protokoll voraus, wohingegen bei SOAP Konversationen mit dem Server über mehrere Nachrichten hinweg aufrechterhalten werden können.

2.4.2 hCard (Mikroformate)

Die hCard gehört zu den Mikroformaten (Microformats) und dient dazu, Daten über den Nutzer als digitale Visitenkarte zu speichern. Die Daten sollen sowohl für Menschen als auch Maschinen verständlich bzw. gut erfassbar sein.

Mikroformat ist ein Markup-Format zur semantischen Annotation von HTML oder XHTML. (vgl. [166])

Technisch bauen die Mikroformate auf dem XHTML bzw. XML Standard auf. Die hCard ist sicherlich eine der Ausprägungen der Mikroformate, die schon eine große Verbreitung gefunden hat (vgl. [108]). Aber auch Formate wie hReview werden zunehmend angewandt, beispielsweise im Apple Store für die Bewertungen (vgl. [132]). Für eine umfassende Nutzung von Mikroformaten argumentieren Ryan King (vgl. [90]) und Rohit Khare (vgl. [89]). King argumentiert, dass Mikroformate die Mischung aus CSS und XML sind und somit die Trennung von Inhalt und Gestaltung (CSS) und gleichzeitig die Maschinen- und Menschenlesbarkeit (XML) zusammenbringen. King verweist insbesondere auf den Nachteil bei XML, dass dort beliebige Knotennamen gewählt werden können, wohingegen bei den Mikroformaten durch Wiederverwendung von bereits bestehenden Definitionen eine konsistente Grundlage geschaffen wird.

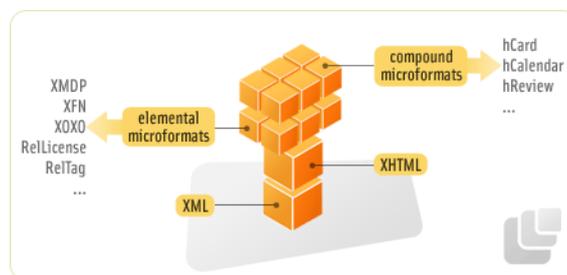


Abb. 2.2: Überblick über Mikroformate (vgl. [109])

Wie in Abbildung 2.2 zu sehen ist, gehört die hCard zu den so genannten gemischten (compound) Formaten. Während die HTML Darstellung, wie im Listing 2.1 gezeigt, keinerlei Informationen über den Inhalt eines Tags enthält, kann sowohl der Mensch als auch der Computer aus dem Listing 2.2 ablesen, um welche Informationen es sich handelt.

```

1 <div>
2 <div>Max Mustermann</div>
3 <div>Musterfirma</div>
4 <div>01234/56789</div>
5 <a href="http://example.com/">http://example.com/</a>
6 </div>

```

Quellcode 2.1: Kontaktinformationen im HTML Format

```

1 <div class="vcard">
2 <div class="fn">Max Mustermann</div>
3 <div class="org">Musterfirma</div>
4 <div class="tel">01234/56789</div>
5 <a class="url" href="http://example.com/">http://example.com/</a>
6 </div>

```

Quellcode 2.2: Kontaktinformationen im hCard Format

Durch die XML Struktur lassen sich die digitalen Visitenkarten in beliebiger Form präsentieren, während die Maschinenlesbarkeit weiterhin erhalten bleibt. Allerdings gibt es keine Möglichkeit mit Hilfe der hCard eine Beziehung zu anderen Personen auszudrücken.

2.4.3 XFN

Das Internet besteht zu einem wichtigen Teil aus Verlinkungen zwischen Objekten. Es gibt aber kaum Informationen, in welchen Beziehungen Webseiten miteinander stehen. Diese Information lässt sich meistens nur aus dem Kontext um den Link herum und nicht aus dem Link selbst extrahieren. Es wird also die Beziehung zwischen zwei Seiten vernachlässigt, die jedoch - wie auch Berners-Lee hervorhebt (vgl. [57]) - genauso interessant ist wie der Inhalt. Nicht der Inhalt, sondern die Struktur steht damit an erster Stelle.

The web is more a social creation than a technical one. I designed it for a social effect — to help people work together — and not as a technical toy. The ultimate goal of the Web is to support and improve our weblike existence in the world. We clump into families, associations, and companies. We develop trust across the miles and distrust around the corner. (vgl. [57])

XFN bietet eben diese Möglichkeit, Links mit Beziehungsinformationen zu annotieren. So lässt sich beispielsweise ein Link mit der Information annotieren, dass es sich um einen Freund handelt, wie in Listing 2.3 gezeigt.

```

1 <a href="http://josh.example.com/" rel="friend">Josh</a>

```

Quellcode 2.3: XFN Codebeispiel[57]

Allerdings lassen sich mit Hilfe von XFN nur einseitige Beziehung aufstellen, da sich aus dem Listing 2.3 nicht ablesen lässt, ob Josh den Autor des Links ebenfalls als Freund ansieht. Dies kritisiert auch Dodds (vgl. [33]), hebt aber gleichzeitig in seinem Vergleich zwischen FOAF und XFN hervor, dass XFN bei vielen Systemen schon vorhanden und daher einfacher zu integrieren sei.

Technisch betrachtet handelt es sich bei XFN ebenfalls um ein Mikroformat, wie in Abbildung 2.2 auf der vorherigen Seite gezeigt. Allerdings gehört es im Unterschied zu der hCard zu den „elementaren“ Formaten¹.

Während die hCard Beziehungen nicht abbilden kann, fehlt XFN die Möglichkeit, Daten über eine Person abzuspeichern. Sind beide Funktionsbereich gewünscht, ist eine Kombination beider Techniken notwendig (vgl. [107]).

¹Mit „elementar“ werden Mikroformate bezeichnet, wenn diese die minimale Lösung für ein bestimmtes Problem darstellen. Als „compound (verbundene)“ Formate werden Konstrukte aus mehreren elementaren Mikroformaten bezeichnet (vgl. [105, 104]).

2.4.4 Friend of a Friend (FOAF)

FOAF documents are, in essence, machine-readable home pages. (vgl. [52])

So beschreiben die Urheber von FOAF zusammenfassend ihre Technologie. Auf der Anwenderseite bietet FOAF die Möglichkeit, Personen und ihre Beziehung zu Objekten und anderen Personen zu beschreiben.

Strukturell ist das FOAF aus Klassen aufgebaut. Dies können sowohl Personen als auch Objekte wie beispielsweise Bilder umfassen. Die Klassen wiederum haben Eigenschaften, wie z.B. eine E-Mailadresse. Die Besonderheit von FOAF zeigt sich in der Möglichkeit, eben diese Klassen mit Beziehungen untereinander zu versehen.

Der Quellcode 2.4 zeigt ein Beispiel für eine FOAF Datei. Dargestellt ist neben persönlichen Daten über Jimmy Wales auch, dass er eine Person mit dem Namen Angela Beesley kennt. Über Angela Beesley könnten nun ebenfalls weitere Daten angezeigt werden. Eine besondere Rolle nimmt die Emailadresse (mbox) ein. Diese dient als eindeutiges Identifikationsmerkmal. Alternativ könnten auch noch die Hashsumme der Emailadresse, die msn/aol/yahoo/icq-ChatID, die OpenID, die Adresse des Weblogs/Homepage oder die Thematik eines Dokuments als ID verwendet werden. Eine gute Einführung in die Verwendung von FOAF bietet auch Edd Dumbill (vgl. [35]). Neben der Anwendung als einfaches Verzeichnis führt er drei weitere Möglichkeiten für eine Anwendung von FOAF auf. Emails könnten auf Grundlage des Bekanntheitsgrades gefiltert werden und der Zugang zu einer Community kann auf Grund der FOAF Daten gewährt werden. Als dritte Möglichkeit bietet FOAF auch die Möglichkeit, Kontakte mit ähnlichen Interessen zu finden. Abschließend stellt er fest:

By aggregating and merging the FOAF files, you can achieve the same effect as operating a centralized directory service, without any of the issues of single points of failure or control. This is a very attractive feature for many communities for which decentralized or devolved control is a requirement, either because of political structure or sheer size.[35]

```
1 <rdf:RDF
2
3   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
4   xmlns:foaf="http://xmlns.com/foaf/0.1/"
5   xmlns:dc="http://purl.org/dc/elements/1.1/"
6   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
7
8   <foaf:Person xml:lang="en">
9     <foaf:name>Jimmy Wales</foaf:name>
10    <foaf:mbox rdf:resource="mailto:jwales@bomis.com"/>
11    <foaf:homepage rdf:resource="http://www.jimmywales.com/" />
12    <foaf:nick>Jimbo</foaf:nick>
13    <foaf:depiction rdf:resource="http://example.org/France.jpg" />
14    <foaf:interest>
15      <rdf:Description rdf:about="http://www.wikimedia.org/"
16        <dc:title>Wikipedia, The Free Encyclopedia</dc:title> </rdf:Description>
17    </foaf:interest>
18    <foaf:knows>
19      <foaf:Person>
20        <foaf:name>Angela Beesley</foaf:name>
21      </foaf:Person>
22    </foaf:knows> </foaf:Person>
23
24 </rdf:RDF>
```

Quellcode 2.4: Beispiel für eine FOAF Datei[163]

Eine präzise technische Einordnung von FOAF gestaltet sich schwierig. Die Betreiber selber schreiben in Ihrer Definition:

This document presents FOAF as a Semantic Web vocabulary or Ontology. (vgl. [18])

form non-blog		from livejournal.com	
foaf:mbox sha1sum	0.84	foaf:mbox sha1sum	1.0
foaf:homepage	0.66	dc:description	1.0
foaf:name	0.64	dc:title	1.0
foaf:nick	0.61	foaf:nick	1.0
foaf:weblog	0.60	foaf:page	1.0
foaf:knows	0.44	foaf:weblog	0.99
foaf:mbox	0.38	rdfs:seeAlso	0.85
foaf:img	0.38	foaf:knows	0.85
bio:olb	0.35	foaf:dateOfBirth	0.71
rdfs:seeAlso	0.34	foaf:interest	0.67

Tab. 2.1: Häufigkeit der Verwendung von FOAF Eigenschaften pro FOAF Datei (vgl. [32])

Dies spiegelt sich auch in den Wikipediadefinitionen wieder. Auf der englischen Wikipediaseite wird es als Ontologie bezeichnet (vgl. [163]), während es auf der deutschen Seite als RDF-Schema (vgl. [163]) eingeordnet wird. Grundlage für die Verwendung in dieser Arbeit ist die Definition von FOAF als semantisches Webvokabular. Auch wenn sich das FOAF eng an W3C Standards anlehnt, ist es selber weder ein ISO noch ein W3C Standard. In einer Studie haben Li Ding et al. (vgl. [32]) sich mit dem FOAF Format beschäftigt. Ihre Definition lautet:

Web homepages for people, groups, companies and other kinds of thing and it is written in XML syntax, and adopts the conventions of the Resource Description Framework (RDF). (vgl. [32])

Da es für FOAF Dokumente keine strikte Syntax gibt, sondern diese an das OpenSource Prinzip angelehnt verändert werden kann, führen Ding et al. eine Unterscheidung in „strikte“ und „generelle“ FOAF Dokumente ein. „Strikte“ FOAF Dokumente zeichnen sich insbesondere dadurch aus, dass sie personenzentriert sind. Dies steht im Gegensatz zu Dokumenten, welche zwar eine Personenbeschreibung beinhalten, aber hauptsächlich der Beschreibung eines Dokuments dienen. In einer zweiten Kategorisierung unterscheiden die Autoren zwischen FOAF Dokumenten, die automatisiert von Blogseiten erstellt wurden und manuell erstellten FOAF Dokumenten. Insbesondere die Verteilung der Attribute wie in Abbildung 2.1 dargestellt war hier von Interesse.

Die aktuelle Spezifikation stammt aus dem Jahr 2007, wobei es FOAF schon seit 2000 gibt. Trotz seiner langen Existenz hat es noch keine Verbreitung im Massenmarkt gefunden². Es existieren einige Werkzeuge³, welche die Erstellung von FOAF Dateien ermöglichen oder eine Visualisierung erstellen, allerdings hat die Technologie in die großen Sozialen Netzwerke noch keinen Eingang gefunden (vgl. [163]). Implementierungen gibt es bisher nur für:

²Dies kann u.a. mit der in Abschnitt 6.7 auf Seite 82 thematisierten Missbrauchsproblematik zusammenhängen.

³<http://www.foaf-project.org/world/index.html>

- LiveJournal
- Peopleaggregator
- Tribe.Net
- Videntity
- Hitflip
- sendia
- Blogscout.de

Für Facebook existieren zwar zwei FOAF-Erweiterungen, diese sind jedoch kein Angebot des Betreibers. Während Matthew Rowe (vgl. [139]) ein Tool anbietet, mit dem die eigenen Daten exportiert werden können, bietet die Applikation von Dmitry Yakhnov (vgl. [177]) die Möglichkeit, alle Freunde der eigenen Kontakte gleichzeitig zu sehen.

Neben der reinen Abbildung von Daten und Kontakten, sind im Umfeld von FOAF einige interessante Anwendungsansätze entstanden. So stellen Jing Mei et al. einen Ansatz vor mit dem sich „Experten“ aus einem Netzwerke von FOAF Dokumenten extrahieren lassen (vgl. [94]). In Anlehnung an OpenID diskutieren Adam Gzella et al. die Möglichkeit, FOAF Daten auf Basis des P2P (Peer-to-Peer) Protokolls für eine verteilte Authentifizierung verwendet werden können (vgl. [66]). Beide Ansätze sollen im Rahmen dieser Arbeit nicht weiter verfolgt werden, zeigen aber auf, dass das Vorliegen von strukturierten Kontakt- und Beziehungsdaten diverse Anwendungsmöglichkeiten bietet.

Dumbill zeigt einige Probleme auf, die aus der Struktur von FOAF resultieren (vgl. [35]). So ist zum Beispiel keine Unterscheidung zwischen privaten und öffentlichen Informationen möglich, jedwede Information verfügt über die gleiche Öffentlichkeitsstufe. Dies ist ein gravierendes Problem, welches aber bei der Verwendung innerhalb von geschlossenen Netzwerken und für den eigenen Export und Import nur eine geringe Relevanz hat. Dumbill wirft auch die Frage auf, wie sich das FOAF in bereits existierende Netzwerke wie Weblogs, RSS, Wikis und IRC integrieren lässt. Außerdem stellt die Veröffentlichung von persönlichen Informationen durch andere Nutzer ein Problem da, da hier nur sehr schwer korrigierend eingegriffen werden kann.

Alternativ zum FOAF Format kann man für die Datenspeicherung die hCard (vgl. Abschnitt 2.4.2 auf Seite 10) und für die Verbindung von Kontakten XFN verwenden (vgl. Abschnitt 2.4.3 auf Seite 11).

Meyer und Doods diskutieren in kurzen Beiträgen die Unterschiede von XFN und FOAF. Hauptkritikpunkt von Meyer ist die fehlende Möglichkeit bei FOAF, unterschiedliche Kontaktgrade auszudrücken (vgl. [103]), wohingegen dies mit XFN möglich ist. Doods hingegen argumentiert, dass auch für FOAF Beziehungsmodelle entwickelt wurden (vgl. [33]). Am XFN Format hingegen kritisiert Doods, dass es kein „ich“ gäbe. In diesem Punkt stimmt er mit Meyer überein. Meyer führt an, dass mit FOAF Informationen über eine Person publiziert werden könnten, die nicht der Wahrheit entsprechen. Beim XFN sei dies zwar auch möglich, aber hier lässt sich überprüfen, ob die Verbindung von beiden Seiten besteht und somit der Wahrheitsgehalt der Information verifizieren.

Zusammenfassend bietet das FOAF einen einheitlichen Ansatz, um persönliche Informationen und Informationen über die eigenen Kontakte in einem maschinen- und menschenlesbaren Format abzubilden.

2.4.5 OpenID

Wegen der Vielzahl von Sozialen Netzwerken und anderen Diensten für die ein Login benötigt wird, ist ein Service, der eine zentrale ID bereitstellt, sinnvoll. Diese Funktionalität erfüllt OpenID. Im Unterschied zu der Microsoft LiveID ist sie dezentral aufgebaut und kann auf jedem beliebigen Server implementiert werden (vgl. [110]).

Die OpenID selbst ist eine URI, die aus dem Provider- und dem Benutzernamen zusammengesetzt ist, wie in Listing 2.5 auf der nächsten Seite gezeigt.

Quellcode 2.5: OpenID Aufbau

Die grundsätzliche Funktionsweise ist in Abbildung 2.3 dargestellt. Zu Beginn des Prozesses möchte sich der Benutzer auf einer beliebigen Seite einloggen, die eine OpenID akzeptiert. Er wählt im Schritt eins die OpenID Authentifizierung aus und gibt seine OpenID ein. Daraufhin wird er in Schritt zwei zu seinem OpenID Anbieter weitergeleitet und im Schritt drei aufgefordert, sein Passwort einzugeben. Im vierten Schritt gibt der Nutzer seine OpenID für die Webseite frei und im fünften Schritt sendet der OpenID Anbieter den Token an die Webseite zurück, die daraufhin den Nutzer einloggt.

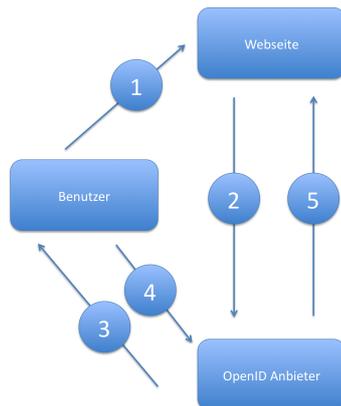


Abb. 2.3: Funktionsweise OpenID

Eine OpenID kann von jedem angeboten werden, der einen Server betreibt. Fertige Implementierungen stehen für die gängigen Programmiersprachen zur Verfügung (vgl. [122]). Die vollständige Spezifikation ist öffentlich einsehbar und ermöglicht somit auch eigene Implementierungen (vgl. [123]).

Zusammenfassend bietet OpenID als offener Standard die Möglichkeit sich mit einer ID bei verschiedenen Diensten zu authentifizieren.

2.5 ÜBERBLICK UND KATEGORISIERUNG VON SOZIALEN NETZWERKEN

In diesem Abschnitt soll nun eine Kategorisierung der verschiedenen Netzwerke vorgenommen werden. Die Übersicht von Sozialen Netzwerken basiert auf der Auflistung von Boyd und Ellison (vgl. [16]) und wurde um bedeutende deutsche bzw. zum Zeitpunkt der Veröffentlichung des Artikels noch nicht existierende Netzwerke ergänzt. Die Aufzählung erhebt keinen Anspruch auf Vollständigkeit⁴. Ein weiterer Überblick über Soziale Netzwerke findet sich in einer Übersichtsseite der Wikipedia (vgl. [165]).

Zur Beschreibung der Sozialen Netzwerke wurde eine Kombination aus statistischen Daten (Gründungsdatum, Mitgliederzahl, URI) und Einschätzungen auf Basis der Funktionalität verwendet. Bei dem Gründungsdatum und der Mitgliederzahl⁵ wurde vorwiegend auf die Angaben der Betreiber

⁴Zu beachten ist, dass bei der Aufzählung die vielen geschlossenen Netzwerke von Firmen und Institutionen fehlen, die als Intranet fungieren.

⁵Die Angaben bei der Mitgliederzahl sind sehr uneinheitlich. Je nach Betreiber werden aktive Nutzer angegeben oder aber auch die Gesamtzahl der registrierten Benutzer. Auch sind die Zahlen, bedingt durch das rasante Wachstum mancher Netzwerke nur als Anhaltspunkt und nicht als exakte Größenbestimmung zu betrachten.

zurückgegriffen. Wenn diese keine Zahlen zur Verfügung stellten, wurden Presseartikel oder als letzte Möglichkeit die Angaben bei Wikipedia als Quelle verwendet.

Die Angaben zu Zielgruppe, inhaltlicher Schwerpunktart und Werkzeugart beruhen auf eigener Recherche. Bei der Zielgruppe wurde zwischen Netzwerken unterschieden, die sich an die General Public (GP) und solchen, die sich nur an eine spezielle Zielgruppe (SZ) wenden. Des Weiteren wurde erfasst, ob eine inhaltliche Schwerpunktsetzung vorliegt. Ein Beispiel hierfür ist Xing, welches seinen inhaltlichen Schwerpunkt auf das Knüpfen von Geschäftskontakten legt. Eine grundlegende Beschreibung der einzelnen Zielgruppen findet sich bei Ina O'Murchu et al. (vgl. [121]). Im letzten Schritt wurde jedem Netzwerk eine Werkzeugart zugewiesen. Hierbei wurde eine grobe Einteilung in die Folgenden drei Bereiche vorgenommen:

- All-in-One Lösung (A)
 - diese bieten viele unterschiedliche Funktionalitäten an
- Create your own Social Network (CSN)
 - bietet dem Nutzer die Möglichkeit, sich auf einfache Art und Weise ein eigenes Soziales Netzwerk zu erstellen
- Tools⁶ (T)
 - Hier wird jeweils nur ein bestimmtes Tool angeboten. Zur genaueren Spezifizierung wurde eine Unterteilung in die jeweiligen Tools vorgenommen
 - * Blog (T-B)
 - * Fotos (T-F)
 - * Video (T-V)
 - * Musik (T-M)
 - * Reise/Unterkunft (T-R)
 - * Lesezeichen (T-L)
 - * Treffen/Meeting (T-T)

Die Ergebnisse der Recherche wurden in zwei Tabellen nach Soziale Netzwerken mit inhaltlichem (Tabelle A.2 auf Seite 102) und ohne inhaltlichem (Tabelle A.1 auf Seite 101) Schwerpunkt gegliedert.

Der begrenzte Rahmen dieser Arbeit lässt keine vollständige Untersuchung aller bestehenden Netzwerke zu. Um aber dennoch ein repräsentatives Bild des Umfelds zu bekommen, wurden relevante Netzwerke nach den folgenden Kriterien ausgewählt:

Größe Das Kriterium bietet sich als grundlegendste Auswahlmöglichkeit an. Auch wenn es inhaltlich zunächst keine Aussage über das Netzwerk ermöglicht, sichert es jedoch eine gewisse Relevanz, da das Netzwerk von zahlreichen Nutzern frequentiert wird.

Regionaler Einfluss Bei diesem Kriterium wurde darauf geachtet, dass in der Untersuchung Netzwerke vertreten sind, die in verschiedenen Regionen der Welt weit verbreitet sind. Hierdurch konnte sichergestellt werden, dass regionale Eigenheiten bei den Funktionalitäten und dem Aufbau berücksichtigt sind.

⁶Hiermit sind einzelne Soziale Funktionalitäten wie beispielsweise das Führen eines Blogs gemeint.

Netzwerk	Begründung für Auswahl
Bebo	Großes internationales Netzwerk mit vielen Nutzern in Irland, Großbritannien, Australien und Neuseeland
Facebook	Großes internationales Netzwerk mit eigener funktionsreicher API
Hi5	Großes internationales Netzwerk mit vielen Nutzern in Mittelamerika und Basis API
LinkedIn	Großes internationales Netzwerk mit dem thematischen Fokus auf Geschäftskontakten
MySpace	Sehr großes soziales Netzwerk mit Fokus auf Musik und Videos
Orkut	Von Google betriebenes Soziales Netzwerk mit großem Anteil in Brasilien und Indien
studivz	Großes Soziales Netzwerke mit Fokus auf der Zielgruppe „Studenten“ und vielen Nutzern in Deutschland
Xing	Größtes deutsches Netzwerk mit dem thematischen Fokus auf Geschäftskontakten

Tab. 2.2: Übersicht über engere Auswahl von Sozialen Netzwerken mit Kriterien

Erweiterungsmöglichkeiten Damit ein Netzwerk auch für weitergehende Einsatzmöglichkeiten interessant ist, sollte es dem Entwickler Möglichkeiten bieten, die Funktionalitäten zu erweitern. Die untersuchten Netzwerke sollten also zumindest eine Basismöglichkeit für Erweiterungen bieten.

Inhaltlicher Fokus Mit der Abdeckung von verschiedenen inhaltlichen Schwerpunkten soll sichergestellt werden, dass mögliche Unterschiede bei der Ausrichtung Berücksichtigung finden.

Personenkreis Hierbei wurde darauf geachtet auch Netzwerke einzubeziehen, die ihren Fokus auf einen bestimmten Personenkreis gelegt haben und somit möglicherweise spezielle Funktionalitäten anbieten.

Betreiber Unter diesen Aspekt fallen Netzwerke, die zwar von ihren sonstigen Kriterien nicht unbedingt eine zentrale Rolle bei der Analyse spielen, bei denen aber auf Grund ihres Betreibers von einer bedeutenden Position im Marktumfeld auszugehen ist.

Tabelle 2.2 zeigt die getroffene Auswahl mit der entsprechenden Begründung.

3 BEWERTUNG DER FUNKTIONALITÄTEN VON SOZIALEN NETZWERKEN

Im vorangegangenen Kapitel wurden die grundlegenden Technologien Sozialer Netzwerke erläutert und diskutiert, die für die Bewertung der Funktionalitäten eine wichtige Grundlage darstellen. Mit Hilfe einer Kategorisierung der Netzwerke wurden acht repräsentative Netzwerke für den Vergleich ausgewählt.

Im folgenden Abschnitt werden die wichtigsten Funktionalitäten der Sozialen Netzwerke zu einer Evaluierungsmatrix zusammengestellt. Insbesondere werden dabei die Möglichkeiten betrachtet, Daten zu schützen und sie zu importieren oder exportieren. Des Weiteren wird eine Evaluierungsmatrix für die Analyse der Funktionalitäten entwickelt, anschließend wird die eigentliche Evaluierung vorgenommen. Diese bietet sowohl in ausformuliertem Text als auch in tabellarischer Form einen Überblick über die einzelnen Funktionen wie auch über die gesamten Netzwerke.

3.1 EINFÜHRUNG IN DEN FUNKTIONSÜBERBLICK

Um eine bessere Vergleichbarkeit der einzelnen Sozialen Netzwerke zu gewährleisten, wurden zunächst die Hauptfunktionalitäten der Netzwerke analysiert. Die Funktionalität wurde jeweils auf einer Skala von eins bis fünf eingeordnet, wobei fünf der höchste erzielbare Wert ist. Null Punkte bedeuten, dass die Funktion nicht verfügbar war. Drei Punkte wurden vergeben, wenn die Funktionalität ohne größere Schwächen implementiert wurde, während Werte über drei ausdrücken, dass die Basisfunktionalität besonders gut oder mit zusätzlichen Funktionen implementiert wurde. Die Sozialen Netzwerke wurden in ihrer Basisfunktionalität ohne mögliche Erweiterungen getestet¹². Für jede untersuchte Funktionalität wurde außerdem der Höchstwert und der Mittelwert ermittelt³, um einen Eindruck über den Stand der Implementierungen der betreffenden Funktionalität zu ermöglichen.

Bis zu einem gewissen Grad unterliegen alle Betrachtungen der subjektiven Wahrnehmung. Durch die Definition der einzelnen Bewertungsaspekte und ihrer möglichen Ausprägungen wurde allerdings versucht, die Analyse so objektiv wie möglich zu gestalten. Aus diesem Grund wurde auch auf anwendungsfallsspezifische Auswertungen verzichtet, da diese den Grad der Subjektivität erhöht hätten.

3.2 ANALYSE DER FUNKTIONALITÄTEN

Bei der Auswahl der Funktionalitäten wurde auf sehr spezialisierte und nur von einem oder zwei Netzwerken unterstützte Möglichkeiten verzichtet. Um die häufigsten Funktionen zu ermitteln, wurden nur Netzwerke herangezogen, die in die Kategorie „All-in-One Lösung“ fallen, wie in Abschnitt 2.5 auf Seite 16 ausgeführt wurde.

Die folgenden Funktionalitäten wurden in den Vergleich einbezogen:

¹Die Einbeziehung von Erweiterungen hätte das Testergebnis erheblich verzerrt, da u.a. nicht für alle Sozialen Netzwerke die Möglichkeit besteht Erweiterungen zu installieren.

²Das Netzwerk Xing wurde als Premiumnutzer evaluiert, da einem unbezahlten Mitglied sehr viel weniger Funktionalitäten bzw. sehr eingeschränkt zur Verfügung stehen.

³Beim Mittelwert wurde die Bewertung mit 0 Punkten herausgenommen, da sie das Ergebnis verfälscht hätte.

- Accountfunktionalitäten
- Applikationen und APIs
- Blog
- Datenfreigabe
- Fotos
- Gästebuch
- Gruppen
- Kalender
- Nachrichten
- Profil
- Suche
- Tagging
- Videos

3.2.1 Accountfunktionalitäten

Bei fast allen Netzwerken ist für die Nutzung eine Anmeldung erforderlich. MySpace stellt eine der wenigen Ausnahmen dar, bei der Seiten auch ohne Registrierung betrachtet werden können. Die Menge an Daten, die für die Registrierung angegeben werden muss, variiert sehr stark. Neben den für die Registrierung notwendigen Informationen wurde erfasst, welche Daten nach einer Registrierung dem Profil hinzugefügt werden können.

Die Tabelle A.3 auf Seite 105⁴ zeigt, dass die meisten Netzwerke den Login über die E-Mailadresse gewählt haben. Bebo wählt einen Zwischenweg, indem der Nutzer sich mit seiner Emailadresse oder mit einem von Bebo gewählten Benutzernamen einloggen kann. Bei den benötigten Informationen für die Registrierung gibt es sehr große Unterschiede. Während Orkut nur Namen und Emailadresse verlangt, verlangt MySpace bereits bei der Registrierung umfangreiche persönliche Informationen (Hobbies, Interessen usw.). Bei den Angaben, die nach der Registrierung gemacht werden können, ähneln sich die Netzwerke hingegen. Vorhandene Variationen haben ihren Ursprung im unterschiedlichen inhaltlichen Schwerpunkt der Netzwerke. So bieten nicht alle Netzwerke die Möglichkeit, politische Ansichten oder den Beziehungsstatus kundzutun. Dafür bieten die auf Geschäftskunden spezialisierten Netzwerke detaillierte Möglichkeiten, Daten zu den eigenen Kompetenzen zu verwalten und eine Beschäftigungshistorie darzustellen.

3.2.2 Applikationen und APIs

Im Bereich der Applikationen und APIs wurde untersucht, welche Möglichkeiten die Betreiber bieten, eigenen Code auf der Plattform auszuführen und welche Möglichkeiten es für den Datenaustausch gibt. Eine detaillierte Analyse findet sich im Kapitel 4 auf Seite 30.

Hinsichtlich der Applikationen wurde analysiert, ob überhaupt die Möglichkeit besteht, eigene Applikationen zu entwickeln und ob der Betreiber Mitglied im OpenSocial Verbund (vgl. 4.2.6 auf Seite 35) ist⁵.

Die Importmöglichkeiten - so sehr diese aus Datenschutz rechtlichen Gründen bedenklich erscheinen mögen⁶ - stellen zu Beginn der Mitgliedschaft eine einfache Möglichkeit dar, auf bereits bestehende Netzwerke des Nutzers zurückzugreifen. Hierbei wurde betrachtet, inwiefern ein Import überhaupt möglich ist und welche Anbieter (Email, IM usw.) unterstützt werden.

⁴Da es sich hierbei mehr um eine Erhebung als um die Bewertung einer Funktionalität handelt, wurden in der tabellarischen Übersicht keine Punkte vergeben.

⁵Hierbei wurde nur die formale Mitgliedschaft in Betracht gezogen und nicht die tatsächlich bereits bestehende Implementierung der API.

⁶Sofort nach der Anmeldung, bevor Orkut Daten zur Verfügung gestellt wurde, schlug Hi5 bei einer Testanmeldung zwei Bekannte vor. Anscheinend hatten diese ihre Kontaktdaten nach Orkut importiert und Hi5 diese auch gespeichert, obwohl die Testanmeldung zu diesem Zeitpunkt noch nicht existierte.

Die Übersicht in der Tabelle A.4 auf Seite 106 zeigt, dass fast alle großen Betreiber inzwischen Mitglied im OpenSocial Verbund sind, der von Google gegründet wurde. Facebook hat sich dem OpenSocial Verbund nicht angeschlossen, sondern setzt mit seiner Marktmacht auf die eigene API. Anzumerken ist, dass z.B. StudiVZ Mitglied im OpenSocial Verbund ist, aber die API noch nicht implementiert hat.

Auch bei den Importmöglichkeiten zeigt sich StudiVZ eher zurückhaltend und ermöglicht diese nicht. Im Gegensatz dazu unterstützt Facebook sehr viele Emailanbieter und auch Textdateien und IM Dienste. Die übrigen Anbieter befinden sich hinsichtlich der Möglichkeit von Applikationen und APIs zwischen den geschilderten Extremen. Eine Sonderrolle nimmt Orkut ein, wo es auch möglich ist, seine Kontakte im CSV Format zu exportieren. Dies steht im besonderen Kontrast zu dem sonstigen Bestreben der Betreiber, möglichst viele Daten zu importieren, aber keine Exportmöglichkeiten zuzulassen.

3.2.3 Blog

Als Pendant zur Pinnwand von StudiVZ bieten einige wenige Netzwerke die Möglichkeit, einen Blog zu führen. Betrachtet wurde, ob diese Möglichkeit besteht und inwiefern es möglich ist, den Blog auch zu abonnieren, so dass andere Nutzer über neue Einträge automatisch informiert werden.

Festzustellen ist, dass bei sehr vielen Netzwerken gar keine Blogginfunktion implementiert ist, wie die Tabelle A.5 auf Seite 107 zeigt. Teilweise wird diese aber durch die Pinnwände einiger sozialer Netzwerke abgedeckt, da die Nutzer dort auch selber schreiben können. Netzwerke, bei denen die Blogfunktionalität vorhanden ist, bieten diese aber zumeist nur mit grundlegenden Funktionen an. Bei Hi5 können zusätzlich Fotos eingebunden werden. Weitere Extras bietet keines der Netzwerke. Nur bei Bebo und MySpace ist eine Abofunktion für den Blog integriert.

3.2.4 Datenfreigabe

In den Medien hat die Privatsphäre in Sozialen Netzwerken in letzter Zeit zunehmend an Bedeutung gewonnen (vgl. [91, 134, 138]). Zwei große Themenblöcke lassen sich im Bereich Privatsphäre ausmachen: zum einen die Freigabe der persönlichen Daten für andere Nutzer und zum anderen der Zugriff des Betreibers auf die Daten des Nutzers.

Bei dem Zugriff der Betreiber auf die Daten des Netzwerks hat der Nutzer fast keine Möglichkeit der Beschränkung. Möchte er nicht, dass seine Daten verwendet werden, kann dies nur verhindert werden, indem der Nutzer keine Daten angibt. Auch wenn durch Gesetze zum Datenschutz vorgegeben ist, dass Daten nicht ohne ausdrückliches Einverständnis der Nutzer an andere Firmen weitergegeben werden dürfen (vgl. Bundesdatenschutzgesetz §28, §29 und §33 - vgl. [140]), lassen sich die Daten bereits innerhalb des Netzwerks für Werbekunden nutzen (vgl. [170]). Zwar gelten die deutschen Datenschutzbestimmungen im internationalen Vergleich als relativ strikt, nichtsdestotrotz muss der Nutzer beachten, dass die deutschen Datenschutzgesetze nicht greifen, wenn er sich beispielsweise bei einem Netzwerk registriert, das seinen Sitz in den USA hat. Hier greifen nur Selbstverpflichtungserklärungen, wie das Safe Harbor Prinzip (vgl. [50] für weitergehende Ausführungen) oder aber eine starke auf die Privatsphäre bedachte Community (vgl. [169]).

An dieser Stelle wurde nur untersucht, wie der Nutzer die eigenen Daten vor anderen Nutzern des Netzwerks schützen kann. Die Frage, inwiefern der Betreiber Zugriff auf die entsprechenden Daten hat, ist rechtlicher Natur und wird daher an dieser Stelle nicht behandelt.

Bei den Funktionalitäten zur Datenabfrage für andere Mitglieder gibt es zwei Herangehensweisen: Zum einen besteht die Möglichkeit, Einstellungen für einzelne Elemente festzulegen. Hierbei ist von Bedeutung, ob nur für den Bereich „Kontakt“ allgemein eine Einstellung vorgenommen werden kann oder für jedes einzelne Element (Telefonnummer, Postleitzahl usw.). Daran gekoppelt sind die Einstellungsmöglichkeiten in Bezug auf die Personengruppen, die ein Informationselement sehen dürfen. Hierunter fällt insbesondere, ob der Nutzer nur zwischen Freunden und der generellen Öffentlichkeit unterscheiden kann oder ob weitere Abstufungen möglich sind.

Wie die Tabelle A.6 auf Seite 108 zeigt, bietet Facebook, welches in letzter Zeit zunehmend in der Kritik stand bezüglich dem Umgang mit den Nutzerdaten die nuanciertesten Möglichkeiten zur Datenfreigabe an. Bei der Granularität werden die Privacyeinstellungen soweit herunter gebrochen, dass sie für jede einzelne Emailadresse festgelegt werden können. Auch bei den Nutzergruppen bietet Facebook sehr viele Einstellungsmöglichkeiten. So können z.B. bestimmte Daten nur für ein Netzwerk freigeben werden. Dort können aber wiederum bestimmte Nutzer ausgenommen werden. Am anderen Ende der Skala befindet sich Bebo, dass nur Einstellungen bei der Datenfreigabe für das Geburtsdatum ermöglicht. Bei weiteren Daten muss der Nutzer entweder auf die Eingabe verzichten oder akzeptieren, dass er nur zwischen einer Ansicht für Freunde oder Nicht-Freunde unterscheiden kann. Erwähnenswert ist noch die Einstellung bei MySpace, wo neben der Begrenzung auf Freunde noch das Alter (über 18) als Kriterium herangezogen werden kann.

Zusammenfassend lässt sich sagen, dass fast alle Netzwerke die Möglichkeit bieten die Sichtbarkeit nur auf Freunde zu begrenzen, wo hingegen bei der Granularität große Unterschiede bestehen.

3.2.5 Fotos

Die Fotofunktionalität gehört zu den populären Funktionalitäten in Sozialen Netzwerken. Auf Facebook sind über 1.7 Milliarden Fotos gespeichert (vgl. [8]). Für den Bereich Fotos wurden drei Kernfunktionen identifiziert:

1. Komfort des Fotouploads, insbesondere für mehrere Fotos sowie die Möglichkeiten Fotos zu bearbeiten (z.B. drehen)
2. Anreicherung von Fotos mit Metadaten und im speziellen die Markierung von Personen
3. Download von Fotos

Große Unterschiede waren im Bereich des Fotouploads nicht zu finden. Das Hochladen von mehreren Fotos bieten die meisten Sozialen Netzwerke an. Interessant ist, dass Facebook und StudiVZ das gleiche Java Plugin verwenden. Die Sozialen Netzwerke, die sich auf Geschäftskunden konzentriert haben, bieten - außer für das Profilfoto - keine Möglichkeit, Fotos hochzuladen. Die Unterschiede in der Bewertung ergeben sich hauptsächlich aus den Möglichkeiten, die dem Nutzer beim Bearbeiten und Sortieren nach dem Upload geboten werden.

Wie die Tabelle A.7 auf Seite 109 zeigt, erfolgt die Markierung von Personen in allen Sozialen Netzwerken ähnlich. Unterschiede lassen sich hinsichtlich der Tatsache konstatieren, ob Namen von Personen angegeben werden können, die bisher nicht Mitglied im sozialen Netzwerk sind.

Eine Option, mehrere Fotos auf einmal herunter zuladen, bietet keines der sozialen Netzwerke. StudiVZ versucht sogar dem durchschnittlichen PC Benutzer den Download zu erschweren, indem kein einfacher Rechtsklick auf die Fotos möglich ist. Bei Hi5 hingegen können Fotos direkt über einen Link herunter geladen werden.

3.2.6 Gästebuch

Ein weit verbreitete Funktion im Internet sind Gästebücher. Diese Funktion haben viele Netzwerke aufgegriffen, indem sie jedem Nutzer ein eigenes Gästebuch zur Verfügung stellen. Zunächst wurde hierbei bewertet, ob die Funktion überhaupt vorhanden ist und anschließend, inwiefern es möglich ist, auch Medien (Fotos und Videos) in diese Nachrichten einzubinden.

Bei der Grundfunktionalität zeigen sich wenig Unterschiede, wie sich anhand der Tabelle A.8 auf Seite 110 nachvollziehen lässt. Als einziges sticht Bebo positiv heraus, wo auch direkt angefertigte Zeichnungen auf die Pinnwand „geschrieben“ werden können. MySpace bietet als einziges Netzwerk keine Pinnwand an.

Die Einbindung von Fotos oder anderen multimedialen Inhalten erlauben nur Orkut und Facebook. Zusammenfassend sind hier sehr wenige Unterschiede erkennbar und das Feld der betrachteten Sozialen Netzwerke stellt sich homogen dar.

3.2.7 Gruppen

Um die Verbindung mit anderen Kontakten zu ermöglichen und auf bereits bestehende Netzwerke zurückgreifen zu können, bieten alle Sozialen Netzwerke die Funktion an, Gruppen zu bilden. Oft werden diese noch um Netzwerke, die z.B. die Mitarbeiter einer bestimmten Firma abbilden, ergänzt. Eine weitere Art der Gruppierung sind so genannte eventzentrierte Gruppen, die anlässlich eines Events zusammengestellt werden.

Bezüglich der Gruppenbildung wurde untersucht, inwiefern die Möglichkeit besteht, diese Gruppen zu erstellen bzw. Mitglied von bereits existierenden Gruppen zu werden. Im zweiten Schritt wurde bewertet, welche Funktionalitäten den Nutzern im Rahmen dieser Gruppen zur Verfügung stehen.

Die Möglichkeit Gruppen zu bilden, bzw. Mitglied von bestehenden Gruppen zu werden bieten alle untersuchten Netzwerke⁷. Je nach Netzwerk können beliebig Gruppen vom Nutzer ohne Genehmigung gegründet werden oder müssen erst vom Betreiber freigeschaltet werden. Unterschiede finden sich in den Funktionalitäten, die einer Gruppe zur Verfügung stehen. Bei Facebook kann beispielsweise jede Gruppe auf eine Pinnwand und ein Diskussionsforum zugreifen. Außerdem können Videos oder Fotos veröffentlicht werden. Bebo hat hingegen ein Umfragetool implementiert.

Während Gruppen eher informell sind, was sich u.a. in Gruppennamen wie „Ich mag keine Bockwurst“ bei StudiVZ äußerst, haben Netzwerke in der Regel einen formaleren Charakter. Bei Facebook sind Netzwerke oft auch an bestimmte Emailadressen (der jeweiligen Firma oder Universität) gekoppelt. Als zweite Netzwerkart haben sich regionale Netzwerke, wie z.B. bei Facebook „Germany“, herauskristallisiert.

Die Tabelle A.9 auf Seite 112 zeigt, dass die Betreiber bei den eventzentrierten Gruppen unterschiedliche Wege gehen. Bei StudiVZ ist zwar kein Eintragen von Terminen möglich, stattdessen kann sich der User jedoch Vorlesungen seiner Universität zuordnen, woran - außer der Feststellung, wer an der selben Lehrveranstaltung teilnimmt - keine weiteren Funktionalitäten gekoppelt sind. Bei Facebook und Xing ist ebenfalls eine Funktionen für die eventzentrierten Gruppen vorhanden, während die anderen Netzwerke keinerlei direkte Implementierung hierfür anbieten.

⁷Bei LinkedIn ist die Funktion momentan noch nicht für alle freigeschaltet.

3.2.8 Nachrichten

Um die Nutzer verstärkt an die eigene Plattform zu binden, ermöglichen alle Netzwerke inzwischen den Versand von Nachrichten⁸. Gegenstand der Analyse war in diesem Fall, ob diese Möglichkeit besteht, mit wem dies erfolgen kann und welche Komfortfunktionen zur Verfügung stehen. Im nächsten Schritt wurde dann erfasst, inwiefern es möglich ist, die Nachrichten mit multimedialen Inhalten anzureichern.

Wie die Tabelle A.11 auf Seite 114 zeigt, ist der einfache Versand von Nachrichten an Kontakte bei fast allen Netzwerken möglich. Als Komfortfunktionen kommt oft die Möglichkeit hinzu, ganze Listen auf einmal anzuschreiben. Facebook bietet faktisch einen eigenen Emails-service. Hier kann der Nutzer auch Nachrichten mit Leuten austauschen, die nicht Mitglied bei Facebook sind. Bei LinkedIn können Nachrichten direkt an die Emailadresse eines Nutzers geschickt werden, allerdings nur, wenn diese Person bereits ein Kontakt des Nutzers ist.

Bei der Einbindung von multimedialen Inhalten gibt es eine große Bandbreite, von sehr einfachen Möglichkeiten bei Xing, bis zur direkten Einbettung von Audiodateien bei Bebo, welche vom Nutzer über den Browser aufgenommen werden können. Überraschend ist allerdings, dass MySpace, welches seinen Schwerpunkt auf multimediale Inhalte legt, keine Möglichkeiten für die Einbettung von multimedialen Elementen bietet.

3.2.9 Kalender

Um das eigene Soziale Netzwerke zum zentralen Ort der Interaktion zu machen, bieten viele Netzwerke neben den bereits geschilderten Funktionalitäten eine Terminfunktionen an. Hier wurde betrachtet, ob es möglich ist Events einzustellen und wie sich dies in die sonstige Struktur des Netzwerks integriert. Für die Interoperabilität und die Nutzung der Daten sind insbesondere die Möglichkeiten des Im- und Exports wichtig.

Auch in diesem Bereich ist die Spannweite der Funktionen sehr breit. Während einige Netzwerke wie Facebook und Xing eine sehr umfassende Terminverwaltung, mit Möglichkeiten seine Teilnahme zuzusagen oder direkt eine Telefonkonferenz abzuhalten, anbieten, haben andere Netzwerke keinerlei Funktionalität in diese Richtung entwickelt. Dies verwundert insbesondere beim auf Geschäftskunden orientierten LinkedIn. Xing und Facebook stechen hinsichtlich der Funktionalitäten in diesem Bereich heraus.

Diese Tendenz setzt sich auch bei den Im- und Exportmöglichkeiten fort, wie in der Tabelle A.10 auf Seite 113 zu sehen ist. Ein Import von Termindaten ist bei keinem der Netzwerke möglich, und auch eine Synchronisierung ist nicht möglich. Xing und Facebook bieten zumindest den Export von ICS Dateien an.

3.2.10 Profil

Die Funktionalitäten für das eigene Profil wurden in zwei Kernbereiche unterteilt. Zum einen besteht die Möglichkeit, das eigene Profil zu verändern und nach seinen Wünschen sowohl grafisch als auch funktional anzupassen⁹. Hinsichtlich dieses Aspektes wurde bewertet, inwieweit Anpassungen möglich sind und wie benutzerfreundlich sie gestaltet sind. Eine weitere sehr verbreitete Funktion ist eine Aktivitätsanzeige, welche Inhalte von Kontakten im Netzwerk anzeigt, die in

⁸Beim Netzwerk Xing ist dies ohne Einschränkungen nur als Premiummitglied möglich.

⁹Zu beachten ist, dass an dieser Stelle nur die Modifikationen am eigenen Profil betrachtet wurden, also die Seite, die fremde Nutzer sehen. Außer Acht gelassen haben wir die Modifikationen, die für die eigene Anzeige vorgenommen werden können.

der letzten Zeit eingestellt wurden. Hier wurde der Umfang und die Einstellungsmöglichkeiten bewertet, vor allem vor dem Hintergrund, dass es schnell zu einer Informationsüberflutung kommen kann.

Die Bandbreite bei der Profilmodifikation ist sehr groß. MySpace bietet dem Nutzer fast völlige Freiheit, so kann das Profil sogar mit eigenem HTML Code angereichert werden. Einen Mittelweg geht Bebo, wo vorgefertigte Designs auf das eigene Profil angewendet werden können. Eine weitere Ausgestaltung wählt Facebook, wo die einzelnen Elemente verschoben und minimiert, allerdings keine eigenen Styles angewendet werden können. Immerhin drei Netzwerke erlauben fast keine Modifikationen an dem eigenen Profil - das Hinzufügen von Applikationen ausgenommen.

Wie sich aus der Tabelle A.12 auf Seite 115 ablesen lässt, bietet Facebook die umfangreichsten Benachrichtigungsfunktionen an. Grundsätzlich wird hier erst einmal fast jede Aktivität der anderen Nutzer - solange die es erlaubt haben - gesendet. MySpace wählt einen direkten Abweg, indem der Nutzer sich aktiv dafür entscheiden kann, worüber er informiert werden möchte. Orkut unterstützt zwar im Prinzip Benachrichtigungen, bietet dies aber bisher nur für Kommentare bei Fotos an. StudiVZ verfügt bisher über keine wirkliche Benachrichtigungsfunktion - alternativ informiert es aber darüber, wer zuletzt das eigene Profil besucht hat.

3.2.11 Suche

Gerade zu Beginn einer Mitgliedschaft stellt die Suche eine wichtige Funktion dar, um mit Freunden Kontakt aufzunehmen, die bereits beim betreffenden Sozialen Netzwerk registriert sind. Eine Aufgliederung in einzelne Funktionsbereiche ist nicht sinnvoll, weshalb in der Tabelle die Suchmöglichkeiten hauptsächlich im Bezug auf den Umfang der durchsuchbaren Objekte und die Einstellungsmöglichkeiten bei der Suche bewertet wurden.

Während der Nutzer bei StudiVZ als Suchkriterien Vorlesungen, politische Richtung und vielen anderen Eigenschaften für die Nutzersuche verwenden kann, erlaubt Bebo praktisch keine Einstellungen bei der Suche. Die Tabelle A.13 auf Seite 116 zeigt, dass in allen Netzwerken nach Personen gesucht werden kann. Meistens steht auch eine Suche für Netzwerke und Gruppen zur Verfügung. Interessant ist der Ansatz von Xing, wo es per Klick auch vordefinierte Suchen gibt, z.B. „Direkte Kontakte, deren Firma/Position sich kürzlich geändert hat“. Dies erspart dem Nutzer das Zusammenstellen von komplizierten Abfragen in der Suchmaske.

3.2.12 Tagging

Mit der Möglichkeit für Nutzer, Inhalte sehr einfach selber einzustellen und zu sortieren hat das „Stichwort“ aus Bibliothekszeiten eine Renaissance erlebt. Tagging bedeutet, ein Objekt mit Stichworten zu beschreiben. Durch die Anzeige als Tagcloud¹⁰ kann beispielsweise die Relevanz eines Themas erkannt werden. Für die Funktionalität Tagging wurde untersucht, inwieweit Objekte mit Tags annotiert werden können.

Wie die Auswertung in der Tabelle A.14 auf Seite 117 deutlich zeigt, sind die Taggingmöglichkeiten bei den Sozialen Netzwerken zumeist auf Fotos beschränkt und auch hier steht meistens nur die Option zur Verfügung, Fotos einer oder mehreren Personen zuzuordnen. Die Gelegenheit Personen mit Tags zu versehen bietet nur Xing an. Hier kann der Nutzer vermerken, woher er einen Kontakt kennt. Facebook bietet zwar an, zu notieren, woher man einen Kontakt kennt, dies geschieht jedoch nicht über ein Tag¹¹. Die übrigen Netzwerke machen keinen intensiven Gebrauch von der Taggingfunktionalität.

¹⁰Tags werden dabei in verschiedenen Größen - je nach Häufigkeit - angezeigt.

¹¹Durch das Zusammenfügen von mehreren Informationshappen generiert Facebook einen Satz, der dann die Beziehung von zwei Kontakten anzeigt.

3.2.13 Videos

Spätestens mit dem Erfolg von Videoportalen wie YouTube (vgl. [80]) haben die Videos auch Einzug bei den Sozialen Netzwerken gefunden. Dies führte sogar dazu, dass MySpace zeitweise die Einbindung von YouTube Videos verhinderte, da es einen eigenen Videodienst aufbauen wollte (vgl. [98]). Grundsätzlich bieten sich im Videobereich drei Funktionalitäten zur Untersuchung an. Als erstes wurde überprüft, ob die Möglichkeit besteht, Videos direkt auf die Plattform hochzuladen. Alternativ wurde erfasst, inwieweit es möglich ist externe Videos auf der Plattform einzubinden. Schließlich wurde - analog zur Betrachtung der Fotofunktion - analysiert, ob es möglich ist, die Videos herunterzuladen.

Die Anbieter gehen bei der Möglichkeit, den Upload von Videos anzubieten, sehr verschiedene Wege. Während MySpace eine möglichst komplette Videohosting-Möglichkeit bieten möchte, setzt z.B. Bebo auf den Anbieter „VideoEgg“. Interessanterweise setzte Hi5 bisher auf denselben Anbieter, hat aber die Möglichkeit für einen Upload komplett deaktiviert. Bei den anderen Anbietern ist kein Upload möglich.

Die Tabelle A.15 auf Seite 118 zeigt, dass das Einbinden von Videos mit einem Link bei drei Plattformen möglich ist. Allerdings ist der Nutzer bei Orkut und Bebo auf YouTube und Google Video beschränkt. Das Herunterladen von Videomaterial ist bei keinem der Anbieter möglich¹². Auch die URI eines Videos erfährt der Nutzer nur bei MySpace problemlos.

3.3 FAZIT DES FUNKTIONSVERGLEICHS

Obgleich die Zielsetzung der einzelnen Netzwerke teilweise sehr unterschiedlich ist, greifen sie fast alle auf den gleichen Kern an Funktionen zurück. Die Grundfunktionen für die 13 identifizierten Funktionsgruppen sind bei fast allen Netzwerken vorhanden. Je nach Schwerpunktsetzung sind die angebotenen Interaktionsmöglichkeiten etwas ausgefeilter oder bieten mehr Flexibilität. Einzig wirkliche Ausnahme sind die Video- und Fotofunktionen. Die zwei Netzwerke, welche sich auf den Businessbereich konzentrieren, bieten hier keinerlei Möglichkeiten. Selbst die Blogfunktion ist, wenn auch nicht direkt, oft indirekt über eine Gästebuch realisiert.

Über alle Funktionsgruppen hinweg bietet Facebook eine ausgefeilte und flexible Umsetzung an. Insbesondere bei den Privacyeinstellungen weist Facebook sehr detaillierte Funktionalitäten auf. StudiVZ hingegen konnte sich auf dem deutschen Markt trotz recht eingeschränkter Funktionalitäten gegen die Konkurrenz bisher durchsetzen (vgl. [15]). Vermutlich ist dies darauf zurückzuführen, dass der Nutzer in der Regel seine Entscheidung für ein Netzwerk nicht am Funktionsumfang festmacht, sondern sich dort registriert, wo die meisten Bekannten angemeldet sind.

Im Bereich der APIs zeichnet sich der Trend ab, dass es in Zukunft nur noch zwei große APIs geben wird. Dies ist zum einen Facebook und zum anderen Google mit der OpenSocial Initiative. Eine vertiefende Betrachtung dieser Problematik findet sich im Kapitel 4 auf Seite 30.

Die erstellte Evaluierungsmatrix stellte sich als sehr geeignet für die vorgenommene Analyse heraus. Bei der Anwendung auf die verschiedenen Netzwerke gab es keinerlei Probleme, eine Funktionalität zu evaluieren. Aus diesem Grund ist davon auszugehen, dass die Evaluierungsmatrix auch für die Analyse weiterer Netzwerke ein gutes Tool darstellt.

Die Ergebnisse der Evaluierung sind in der Tabelle 3.1 auf der nächsten Seite zusammengefasst. Zu beachten ist, dass die Summe aller Funktionsbewertungen nur einen groben Überblick über die tatsächliche Qualität des Netzwerks liefert. Dies ist oft sehr viel mehr abhängig vom

¹²Hierbei wurde lediglich evaluiert, ob eine Nutzerfunktion zur Verfügung steht, nicht aber ob es mit anderen Programm möglich ist die Videos doch herunterzuladen.

konkreten Anwendungsfall. Dieser konnte in der vorliegenden Analyse, auf Grund der hohen Variationsmöglichkeiten bei der Anwendung, nicht in Betracht gezogen werden. Allerdings können die vorliegenden Ergebnisse durch eine Gewichtung der einzelnen Funktionsbereiche für zahlreiche Anwendungsfälle aufbereitet werden. Eine pauschale Gewichtung erschien aus Gründen der Subjektivität nicht sinnvoll.

	studiVZ	Facebook	Xing	MySpace	Orkut	LinkedIn	Bebo	Hi5
Account-funktionalitäten	-	-	-	-	-	-	-	-
Applikationen und APIs	1	4	3	4	5	2	4	4
Blog	0	0	0	3	0	0	3	2
Datenfreigabe	4	5	4	2	3	2	2	3
Fotos	2	3	0	3	1	0	3	3
Gästebuch	2	3	2	0	2	0	2	2
Gruppen	4	3	2	1	1	1	2	2
Kalender	0	4	2	2	1	0	0	0
Nachrichten	2	4	2	2	3	2	4	3
Profil	3	5	3	5	1	1	4	3
Suche	3	4	4	3	3	3	2	3
Tagging	3	4	2	3	3	3	3	3
Videos	0	0	0	2	1	0	2	0
Summe	24	39	24	30	24	14	31	28
Rang	(5.)	(1.)	(5.)	(2.)	(5.)	(6.)	(3.)	(4.)
Variationskoeffizient	0,792	0,541	0,730	0,557	0,757	1,095	0,457	0,559

Tab. 3.1: Übersicht über die Analyses der Funktionalitäten von Sozialen Netzwerken

4 APIS IN SOZIALEN NETZWERKEN

Im vorangegangenen Kapitel lag der Fokus auf allen Funktionalitäten Sozialer Netzwerke. Dies war insbesondere notwendig, um die im Kapitel 5 auf Seite 42 erfolgte Analyse der Frameworks vornehmen zu können. Ferner konnte die erstellte Evaluierungsmatrix in der Anwendung erprobt werden.

Im folgenden Kapitel werden die Möglichkeiten betrachtet, wie ein Entwickler eigene Software für Soziale Netzwerke integrieren und auf die Daten der Nutzer zugreifen kann. Dabei stehen neben den Funktionalitäten auch der Aufbau der API und die verfügbaren Wrappersprachen im Vordergrund. Dabei finden die in Abschnitt 2.4 auf Seite 8 erläuterten Technologien Anwendung. Aus den Ergebnissen sollen Schlüsse gezogen werden, auf welches Netzwerk ein Zugriff des Prototypen zum Datenaustausch möglich ist.

4.1 KATEGORIEN VON APIS

Für die Realisierung von APIs gibt es zwei unterschiedliche Ansätze. Zum einen bieten Betreiber die Möglichkeit, eigene Programme zu schreiben und damit zusätzliche Funktionalitäten bereitzustellen. Der zweite Ansatz konzentriert sich auf den Zugriff auf Nutzerdaten. In der Praxis werden beide Ansätze aber oft vermischt. Dieses Kapitel betrachtet im ersten Teil zuerst die APIs mit dem Fokus auf die Applikationsentwicklung. Im zweiten Teil werden APIs untersucht, welche die Verknüpfung von Nutzerdaten mit externen Systemen erlauben.

4.2 APIS FÜR APPLIKATIONEN IN SOZIALEN NETZWERKEN

Mit einem geschlossenen System und einer recht einfachen und übersichtlichen Struktur war StudiVZ lange in Deutschland erfolgreich. Das Vorbild Facebook öffnete im Mai 2007 seine Pforten für fremde Applikationen (vgl. [39]). Im Mai 2008 entschied sich auch StudiVZ dem OpenSocial¹ Verbund von Google beizutreten² (vgl. [71]). Diese Entwicklung von StudiVZ und Facebook steht stellvertretend für den Trend, dass die Öffnung für externe Applikationen immer mehr an Bedeutung gewinnt.

Die Bandbreite der möglichen Applikationen ist weit gestreut. Sie reicht von Erweiterungen für schon vorhandene Funktionalitäten, wie z.B. die SuperWall bei Facebook, welche Multimedia Applikationen im Gästebuch erlaubt, bis hin zu einer ganz eigenen Quizanwendung, die nur den einfachen Zugang zu einer großen Menge an Anwendern nutzt. Auch eigene Anwendungen für den Desktop, um Fotos hochzuladen, sind möglich.

Im Folgenden wurden von den zuvor ausgewählten Netzwerken diese näher betrachtet, welche bereits über eine implementierte API verfügen. Eine Ankündigung, dass man diese bald bereitstellen würde, reichte nicht aus.³

Untersucht wurden die folgenden Netzwerke⁴:

¹Framework von Google, welches das Entwickeln von Applikationen für Soziale Netzwerke ermöglicht. Weitere Informationen finden sich im Abschnitt 4.2.6 auf Seite 35)

²Die eigentliche Aktivierung der Funktion ist aber erst für den Sommer 2008 geplant.

³Stand: 26. Mai 2008

⁴Das Soziale Netzwerk Xing hätte eigentlich auch zu den untersuchten Netzwerken gehört. Unter https://develop.xing.com/api_docs waren auch schon erste API Spezifikationen bereit gestellt. Inzwischen ist die Seite aber nicht mehr aufrufbar. Allerdings gehört Xing auch dem OpenSocial Verbund an, weshalb davon auszugehen ist, dass sie über diesen Weg auch noch Applikationen erlauben werden (vgl. [174]).

- Facebook
- MySpace
- Orkut
- Bebo
- Hi5
- OpenSocial⁵

Soweit möglich wurde versucht eine Vergleichbarkeit der verschiedenen APIs zu gewährleisten. Mit Hilfe geeigneter Kriterien soll eine Einschätzung der Relevanz, der technischen Grundlage und der Einsetzbarkeit in der Praxis erfolgen. Um dies zu erreichen, wurden für jede API die folgenden Kriterien untersucht:

- Statistische Daten zur API (Aktivierungsdatum und Anzahl der verfügbaren Programme)
- Technischer Aufbau der API
- Angebotene Wrappersprachen
- Codetestingtools
- Dokumentation

4.2.1 Facebook

Die Facebook API wurde im Mai 2007 aktiviert und wird heute von mehr als 20.000 Applikationen verwendet, zu denen jeden Tag mehr als 140 neue hinzukommen (vgl. [45]). Sie besteht im Wesentlichen aus drei Komponenten: der API selber, einer Query- (Facebook Query Language - FQL) und einer Markup Sprache (Facebook Markup Language - FBML).

Die API ist nach dem REST-Prinzip (Representational State Transfer) implementiert. Der Entwickler hat durch die zur Verfügung gestellten Funktionen Zugriff auf Profile, Freunde, Seitendaten⁶, Gruppen, Photos und Termine. Die Antworten der API erfolgen im XML Format. Auch der Fotoupload ist über die API möglich. Insgesamt stehen dem Entwickler über 50 Funktionen zur Verfügung.

Die Facebook Query Language bietet die Möglichkeit, direkte Anfragen an die Datenbank zu stellen. Sie ist angelehnt an SQL, aber nicht ganz so reich an Funktionen. So kann z.B. der „FROM“ Teil einer Abfrage nur eine Tabelle enthalten. Im Vergleich zur API lassen sich mit der FQL wesentlich genauer und ressourcenschonende Anfrage stellen. Die Rückgabe der Werte erfolgt im XML Format.

Die FBML bietet dem Entwickler die Möglichkeit sich beim Look and Feel sehr eng an Facebook anzulehnen. Konkret gliedert sich die FBML in fünf Funktionsbereiche (vgl. [40]):

⁵OpenSocial ist in dem Sinne kein eigenes Netzwerk, sondern eine reine API, aber auf Grund der Relevanz und der vielfachen Verwendung wurde es hier auch mit untersucht.

⁶Hiermit bezeichnet Facebook kleine Homepages von Unternehmen und Vereinen, die sich innerhalb von Facebook losgelöst von einem Profil präsentieren können.

- **“Social data tags”** - Hierüber kann der Entwickler auf die Daten des Nutzers zugreifen. Dies kann z.B. der Name des Nutzers sein (fb:name). An das Tag selber ist zusätzlich das komplette Rechtesystem von Facebook gekoppelt. So wird bei einem nicht eingeloggten Nutzer nur ein Teil des Namens angezeigt. Zusätzlich sind noch grammatikalische Regeln für die Attribute hinterlegt.
- **“Sanitization tags”** - Hierbei handelt es sich um eine genauere Spezifikation von der Tag-klasse „objekt“ aus HTML. Es werden diverse Tags für z.B. Flashinhalte bereitgestellt, denen dann wiederum bestimmte Verhaltensweisen von Facebook zugewiesen werden. Ein Beispiel dafür ist die Regelung von Facebook, dass ein Film nicht ohne Nutzerinteraktion starten darf.
- **“Design tags”** - Diese ermöglichen es dem Nutzer Facebook typische Bedienelemente zu übernehmen. So können z.B. mit „fb:tags“ Tabs für die Bedienung einer Applikation erstellt werden.
- **“Component tags”** - Hierbei handelt es sich um eine einfache Möglichkeit für den Entwickler, Elemente wie z.B. eine Kommentarfunktion in seine Applikation einzubinden.
- **“Control tags”** - Diese ermöglichen es, die Anzeige bestimmter Informationen auf einen definierten Nutzerkreis, wie beispielsweise den Besitzer der Applikation zu begrenzen. Allerdings ist dies nur ein grafischer Eingriff in die Seite - im Quelltext sind die Informationen immer noch zu finden. Die Privacyfunktion wird über die „social data tags“ abgedeckt.

Mit FBJS (Facebook JavaScript) besteht die Möglichkeit, JavaScript in den Facebook Applikationen zu verwenden. Anstatt einem iFrame werden die JavaScript Befehle mit einer eindeutigen ID benannt und geparsed.

Für die Verwendung der API bietet Facebook zahlreiche Wrappersprachen an. Viele davon sind Produkte der Communities und werden nicht offiziell von Facebook unterstützt. Zur Verfügung stehen: ASP.NET, ASP (VBScript), Cocoa, ColdFusion, C++, C#, D, Emacs Lisp, Java, JavaScript, Lisp, Perl, PHP, Python, Ruby on Rails, VB.NET,| Windows Mobile (vgl. [47]).

Für das Codetesting stehen den Entwicklern drei Tools zur Verfügung. In der API Test Konsole können Methoden direkt ausprobiert werden und die Rückgabewerte, entweder allgemein oder auf eine bestimmte Applikation bezogen, angezeigt werden. In der FBML Test Konsole können FBML Elemente ausprobiert und getestet werden. Eine Vorschauanzeige gibt Auskunft über das erwartete Resultat. In der Feed Vorschau Konsole kann der Entwickler sich Vorschauen von den erzeugten Feeds anzeigen lassen⁷.

Facebook stellt eine sehr ausführliche Dokumentation zur Verfügung (vgl. [41]). Zur API, FBML und FQL gibt es jeweils eine ausführliche Dokumentation mit Beispielen. Bei den Wrappersprachen ist der Stand der Dokumentation recht unterschiedlich, da diese oft von der Community gepflegt werden.

Es ist auch möglich, die Facebook API für das eigene soziale Netzwerk zu lizenzieren. (vgl. [168]). Deuteten sich in der Vergangenheit schon Bestrebungen an, die Facebook API unter die Open-Source Lizenz zu stellen (vgl. [12, 141]), ist dies mit dem 03.06.2008 auch tatsächlich geschehen. Große Teile der API wurden unter die Common Public Attribution License (CPAL) und die Mozilla Public License (MPL) gestellt (vgl. [156]).

⁷Diese dienen dazu, bei Facebook unterschiedlichen Nutzern anzuzeigen, welche Aktionen ihre Kontakte in letzter Zeit durchgeführt haben.

4.2.2 MySpace

Bei MySpace wurde die API am 30.01.2008 (vgl. [171]) aktiviert. Auch hier gibt es einen dreigliedrigen Aufbau, der sich jedoch etwas anders darstellt als bei Facebook (vgl. [118]). (1) Über die REST API stehen Funktionen für Alben, Daten von Applikationen, Kontakten, Fotos, Profilen, Stati, Nutzer, Videos und Moodstati zur Verfügung. Die Rückgabe der Ergebnisse erfolgt entweder im XML oder JSON (JavaScript Object Notation) Format. (2) Die OpenSocial Implementierung unterstützt die Versionen 0.6 und 0.7 sowie die Gadget Version 0.7. (3) MyOpenSpace stellt erweiterte Funktionalitäten zur Verfügung, die von OpenSocial nicht abgedeckt werden. Dies betrifft Funktionen für Videos, Fotos, Alben, Gruppen und den erweiterten Zugriff auf Felder im Profil der Nutzer. Diese sind ebenfalls in JavaScript implementiert, funktionieren dann aber nur auf der MySpace Plattform.

Für die REST API stehen als Wrappersprachen PHP, Ruby, Python und eine VB.Net Library zur Verfügung. Die OpenSocial API kann über JavaScript, ActionScript3 und ActionScript2 angesprochen werden.

Mit Hilfe von OSTool/Harness kann der Entwickler seinen kompletten Code ausprobieren, ohne dass Profile oder andere Dinge verändert werden. OSTool kann in drei verschiedenen Modi laufen: „Normale Ansicht - mit der Applikation installiert“, „Normale Ansicht - ohne die Applikation installiert“ und „Benutzer sieht das eigene Profil“

Über das API Tool können verschiedene REST API Befehle erprobt werden. Nach Auswahl des „Resource Call“ und des „Response Format“ (entweder XML oder JSON) wird die URI für den entsprechenden Befehle generiert, aber auch der komplette HTTP Header, der Header der Antwort und der Body der Antwort. Zusätzlich liefert das Tool eine kurze Beschreibung der Funktion und des zu erwarteten Ergebnisses.

Mit Hilfe des OAuth Tool kann die Authentifizierung des Zugriffs auf die MySpace Ressourcen von der eigenen Applikation getestet werden. Diese erfolgt über eine Kombination aus dem „Consumer Key“ und dem „Consumer Secret“. Das Tool zeigt dem Entwickler dann wiederum die Request URI und den Antwort Body an. Für Applikationen, die außerhalb von MySpace laufen, benötigt es noch ein anderes Authentifizierungsverfahren, welches über External App Auth Tool abgedeckt wird.

Die Dokumentation der Code Referenz ist übersichtlich und vollständig. Allerdings fehlen einführende Erklärungen bzw. sind nur sehr vereinzelt vorhanden. Auch erklärende Tutorials oder eine Beispielapplikation, anhand welcher der Entwickler einzelne Funktionen kennen lernen könnte, sind nicht verfügbar.

4.2.3 Orkut

Orkut hat seine API am 16.04.2008 (vgl. [136]) aktiviert und es stehen 120 Programme (Stand: 30.05.2008) zur Verfügung.

Da Google der Betreiber von Orkut ist, hat es die OpenSocial API in der Version 0.8 und die damit einhergehende RESTful API⁸ implementiert.

Als Wrappersprache für den Entwickler steht nur JavaScript zur Verfügung.

⁸Sie stellt eine wesentliche Neuerung in der OpenSocial Version 0.8 da, aber eine ausführliche Betrachtung dazu wird im OpenSocial Abschnitt durchgeführt.

Vor der Veröffentlichung müssen Orkut Applikationen von Google freigegeben werden. Für den Test steht eine so genannte Developer Sandbox zur Verfügung, die im eigenen Profil läuft und mit deren Hilfe der Entwickler seine Applikation testen kann. Spezielle Tools sind aber nicht vorhanden.

Orkut verweist für die Dokumentation fast vollständig auf die OpenSocial Dokumente. Es gibt ein eigenes Developer Forum und eine Mailingliste. Außerdem ist eine Übersicht über den Aufbau der eigenen Orkut Applikation verfügbar.

4.2.4 Bebo

Die Aktivierung der Bebo API erfolgte am 10.01.2008 (vgl. [9]), am 30.05.2008 standen für die Nutzer 3715 Applikationen zur Verfügung.

Bebos Aufbau der API ist an die Verfahrensweise von Facebook angelehnt. Auch sie besteht grundsätzlich aus drei Teilen (vgl. [10]). (1) Die SNML (Social Network Markup Language) baut auf HTML auf. Es schließt auch gleichzeitig bestimmte HTML Tags von der Nutzung aus. Grundsätzlich lassen sich die Tags in drei Kategorien unterteilen:

- **„Rendering tags“** - Hiermit können Kontrollelemente erstellt werden, die ohne zusätzliche Angaben im Bebo Look and Feel dargestellt werden.
- **„Data backend tags“** - Hiermit können Informationen aus dem Sozialen Netzwerk abgerufen werden.
- **„Conditional tags“** - Diese ermöglichen die dynamische Anpassung einer Applikation an die Umgebung - beispielsweise in Bezug darauf, ob zwischen zwei Nutzern eine Verbindung besteht.

Die SNQL (Social Network Query Language) bietet die Möglichkeit, direkt auf die Bebo Datenbank zuzugreifen. Bebo definiert es als Untergruppe von SQL.

Für die Remote API stehen 38 Funktionen zur Verfügung, mit denen Informationen über Bands, Freunde, Photos und Nutzer abgerufen werden können. Funktionen, die in Facebook und Bebo zur Verfügung stehen, sind zusätzlich noch mit ihrem Facebooknamen vorhanden. Somit können Facebookapplikationen im gewissen Umfang ebenfalls auf der Bebo Plattform verwendet werden (vgl. [119]).

Zwar ist Bebo Mitglied der OpenSocial Vereinigung, aber in der Dokumentation, Blog und Pressemitteilung lassen sich keinerlei Hinweise darauf finden, dass die API auch implementiert wurde. Laut Ankündigung hätte dies bereits bis Ende letzten Jahres geschehen sollen (vgl. [56]).

Als Wrappersprachen für die Entwickler stehen Java, PHP und Ruby⁹ zur Verfügung.

Zum Testen der eigenen Codestücke stehen eine Sandbox, eine Remote API Testumgebung sowie die Auswahl verschiedener Sichtbarkeitsstufen zur Verfügung. In der SNML Sandbox kann der Code ohne Auswirkungen auf Profile etc. getestet werden. Über den Remote Tester können API Befehle ausprobiert und deren Response getestet werden. Zusätzlich kann der Entwickler seine Applikationen in verschiedenen Sichtbarkeitsstufen verwalten: die Applikation ist entweder nur für ihn verfügbar, für alle verfügbar, aber nicht im Verzeichnis gelistet oder mit einem Eintrag im Verzeichnis für alle verfügbar.

Für die drei Hauptbestandteile der API sind jeweils eigene Dokumentationen verfügbar, die zwar vollständig, aber ohne umfassende Einführung gestaltet sind. Es sind ferner keine Tutorials oder Beispiele vorhanden. Dem Austausch unter den Entwicklern dient ein Forum. FAQs sind zwar aufgeführt, aber bisher ohne Inhalte.

⁹Der Ruby Wrapper wurde von einer externen Firma bereitgestellt.

4.2.5 Hi5

Hi5 bietet seit dem 04.04.2008 (vgl. [113]) Entwicklern die Möglichkeit, eigene Programme anzubieten, am 30.05.2008 standen 649 Programme zur Verfügung.

Die API besteht ähnlich wie bei Bebo und Facebook aus drei Kernkomponenten:

1. Die SOAP und REST API befinden sich zurzeit beide noch im Betastadium, aber schon jetzt stehen zahlreiche Funktionalitäten für den Zugriff auf die Daten von Musik, Updates von Freunden, Profildaten und Statusdaten zur Verfügung. Alle zugänglichen Daten sind über ein XML Schema beschrieben.
2. Hi5 hat die OpenSocial Plattform in der Version 0.7 (Stand: 03.06.2008) implementiert. Um auch den Zugriff auf Hi5-spezifische Felder, die nicht von OpenSocial abgedeckt sind zu gewährleisten, bietet Hi5 entsprechende Erweiterungen an (vgl. [73]).
3. Custom Tags bieten zusätzlich zu OpenSocial dem Entwickler die Möglichkeit, seine Applikation an bestimmten Stellen in der Hi5 Plattform einzuklinken.

Als Wrappersprachen stehen für die REST und SOAP API, Java 1.4+ und Java 5 Client Bibliotheken zur Verfügung. Darüber hinaus wird die API wie bei OpenSocial üblich per JavaScript angesteuert.

Applikationen können zuerst in einer Sandbox entwickelt werden, bevor sie in das öffentliche Verzeichnis eingestellt werden. Andere Tools zum Testen des Codes existieren nicht.

Das Zentrum der Dokumentation ist ein Wiki, welches teilweise von den Benutzern selber editiert werden kann. Neben den zahlreichen Referenzen zu der API stehen auch Code-Beispiele zur Verfügung. Außerdem ist verzeichnet, was den typischen Aufbau einer Hi5 Applikation auszeichnet. Praktisch ist der Teil der Dokumentation, in dem die Möglichkeiten zur Internationalisierung der eigenen API erklärt werden.

4.2.6 OpenSocial

Die erste Version der API wurde am 01.11.2007 (vgl. [17]) aktiviert. Am 11.06.2008 hatte die Initiative 75 Mitglieder (vgl. [63]).

Die aktuell vorliegende API Version ist 0.8 (Stand: 02.06.2008). OpenSocial Applikationen laufen auf der entsprechenden Plattform in einem iFrame und können daher nicht direkt mit der restlichen Umgebung des Sozialen Netzwerks interagieren. Die API kann von den Entwicklern mit JavaScript angesprochen werden. Seit der Version 0.8 steht auch eine Rest API zur Verfügung. Google unterteilt die Funktionen der JavaScript API grundsätzlich in die folgenden Bereiche (vgl. [61]):

- **„People“** sind Funktionalitäten, mit denen der Entwickler Profilinformationen über die betreffende Person abrufen kann¹⁰.
- **„Relationships“** sind Funktionalitäten, die Zugriff auf die Freunde einer Person gewährleisten.
- **„Activities“** sind Nachrichten darüber, was Kontakte während der letzten Zeit im Sozialen Netzwerk getan haben.

¹⁰Eine Dopplung bei den Nutzer IDs wird vermieden, indem der Domainname als Präfix vorangestellt wird.

- **„Persistence“** ermöglicht es dem Entwickler, Daten des Benutzers über eine Session hinweg zu verwalten.
- **„Views“** beinhaltet Funktionen, die adaptives Rendering ermöglichen, je nachdem in welcher Ansicht (Profil, Homepage, etc.) die Applikation gerade angezeigt wird.
- **„Lifecycle“** Events ermöglichen es dem Entwickler, bei bestimmten Aktionen wie z.B. dem Installieren der Anwendung, benachrichtigt zu werden.

Die REST API ermöglicht die Verwendung von weiteren Programmiersprachen. Antworten werden entweder im XML Format oder im JSON Format übermittelt. Für die Funktionalitäten der JavaScript API gibt es keine weiteren Wrappersprachen.

Auf den OpenSocial Seiten selber bietet Google keine Möglichkeit an, Code zu testen. Der Entwickler ist folglich auf die Angebote der Netzwerke angewiesen, welche OpenSocial unterstützen.

Es existiert ein grober Überblick über die vorhandenen APIs und grundlegenden Funktionen in Form ausformulierter Texte. In der ausführlichen Code Referenz befindet sich eine Auflistung aller vorhandenen Funktionen. Für die REST API existiert ebenfalls eine Seite mit Einführungsbeispielen. Zum Erstellen einer OpenSocial Anwendung gibt es ein Step-to-Step Tutorial mit Einführungsvideos. Um den Entwickler zu Beginn zu unterstützen, stehen verschiedene Codebeispiele zur Verfügung, die für die eigene Applikation verwendet werden können.

4.2.7 Fazit des API-Vergleichs

„Ohne API geht es nicht.“ - so könnte kurzgefasst das Fazit des Vergleichs lauten. Selbst Betreiber wie Xing und StudiVZ, die bisher eine Plattform ohne API angeboten haben, haben sich nach der Veröffentlichung von OpenSocial dazu entschlossen, in Zukunft Applikationen zuzulassen. Facebook kann sicherlich als Wegweiser bei der API Gestaltung gesehen werden. Kein anderes Netzwerk bietet eine derart umfangreiche Funktionalität. Für den Entwickler besonders hilfreich ist die Möglichkeit, über die FBML von Facebook vordefinierte Funktionen direkt mitzubeneutzen. Ein großer Nachteil war bisher allerdings die Geschlossenheit der Facebook API. Mit dem ersten Schritt in Richtung einer Öffnung, der eine Lizenzierung erlaubte, und dem zweiten Schritt in Richtung OpenSource beseitigt Facebook aber auch diese Schwäche weitestgehend.

Das größte Konkurrenz-Framework für Facebook ist Googles OpenSocial. Während bei der Facebook API zunächst der Einsatz im eigenen Netzwerk Priorität hatte, ist OpenSocial von Anfang an darauf angelegt gewesen, von möglichst vielen Netzwerken implementiert zu werden, was jedoch auch gewisse Nachteile birgt. So kann nicht direkt mit dem Sozialen Netzwerke interagiert werden, wie dies zum Beispiel mit der Markup Language bei Facebook möglich ist. Sehr attraktiv ist hingegen, dass eine Applikation nur einmal entwickelt werden muss und dann auf vielen Plattformen lauffähig ist.

Den Zugriff auf Nutzerdaten bieten alle Netzwerke an. Problematisch ist, dass beispielsweise über OpenSocial nicht alle Profilinformationen abgefragt werden, wenn diese nicht durch die OpenSocial Spezifikationen abgedeckt sind. MySpace bietet aus diesem Grund noch erweiterte Funktionalitäten an, um die Abbildung der Profildaten dennoch zu ermöglichen. Auch stehen in den unterschiedlichen Netzwerken nicht alle Funktionalitäten (Lesen und Schreiben) für alle Profelfelder zur Verfügung.

Vergleicht man die Strukturen der APIs, kristallisieren sich zwei große Gruppen heraus. Auf der einen Seite stehen die APIs, die aus der eigentlichen API, Markup Sprache und Datenbankabfrage aufgebaut sind (Facebook und Bebo). In der Regel sind hier auch zahlreiche Wrapper-sprachen vorhanden, welche die eigentliche API in der für den Entwickler favorisierten Sprache zur Verfügung stellen. Diese Dreiteilung existiert bei OpenSocial bisher nicht. Allerdings stehen

bisher auch nicht so plattformnahe Funktionalitäten wie eine Markupssprache zur Verfügung. Mit einer zusätzlichen REST API in der Version 0.8 schlägt aber auch OpenSocial den Weg ein, mehr Programmiersprachen und verschiedene Zugangsmöglichkeiten zu den Daten der Nutzer zu unterstützen. Facebook nähert sich andererseits OpenSocial an, in dem es seine Plattform zum größten Teil unter OpenSource Lizenzen zur Verfügung stellt. Die zwei Tendenzen sind also ganz klar die Öffnung für verschiedene Programmiersprache und die kostenlose Lizenzierung für weitere Netzwerke.

4.3 APIS FÜR DIE VERKNÜPFUNG VON NUTZERDATEN MIT EXTERNEN WEBSEITEN

Die Betreiber der Sozialen Netzwerke haben kein Interesse daran, dass der Nutzer seine Daten aus dem Netzwerk exportiert und sie damit die Kontrolle über die Verwendung der Daten verlieren. Damit der Nutzer aber trotzdem auf möglichst vielen Internetseiten die Daten aus ihrem Sozialen Netzwerk verwendet, bieten die Betreiber dem Anwender oft, statt dem Export der Daten die Möglichkeit, sein Profil mit externen Webseiten zu verknüpfen. So kann der Nutzer seine Daten und Kontakte aus dem Sozialen Netzwerk auch dort verwenden. Im folgenden Abschnitt wird untersucht, inwiefern es möglich ist, sein eigenes Profil bzw. die Daten aus diesem mit anderen Webseiten zu verbinden oder die Daten ganz zu exportieren.

4.3.1 Google Friend Connect

Google bietet die Möglichkeit, Kontaktdaten aus dem Sozialen Netzwerk auf der eigenen Webseite zu integrieren. Die Beschreibung der API ist zwar verfügbar, aber die Funktionalität ist noch nicht aktiviert (Stand: 03.10.2008). Die „People API“ gehört zur Google Data API Client Library. Für diese stehen Wrappersprachen in Java, .Net, PHP, Python, Objective-C und JavaScript zur Verfügung.

Betrachtet man die Funktionen, ist die API weniger umfangreich als z.B. die Yahoo und Microsoft API. Neben den Funktionalitäten zur Authentifizierung gibt es nur die Möglichkeit, Details über eine Person herauszufiltern oder eine Liste mit Kontakten zu erhalten (vgl. [62]). Anhand einiger Beispielwebseiten zeigt Google, wie sie sich die Benutzung von Google Friend Connect vorstellen. So sollen OpenSocial Applikationen bzw. Google Gadgets auf der eigenen Seite die Kontaktdaten aus den Sozialen Netzwerken beziehen, um dann die bereits bestehenden Communities auf der eigenen Webseite zu verwenden (vgl. [59]).

4.3.2 Facebook Connect

Facebook Connect befindet sich noch in der Betaphase, aber im Laufe des Oktobers 2008 sollen die ersten Webseiten freigeschaltet werden (vgl. [43]). Im wesentlichen bietet die API die folgenden Funktionen:

- Verknüpfung des eigenen Profils mit einer externen Seite und Anzeige der Informationen dort
- Verlinkung von Kontakten aus Facebook zur externen Seite
- Informationen von der externen Seite im Facebook Newsfeed veröffentlichen
- Privacyeinstellungen aus Facebook für die externe Seite übernehmen

Hier liegt der Fokus mehr auf den Daten der Nutzer und den Kontakten und nicht so sehr auf den Applikationen, wie dies bei Google der Fall ist.

4.3.3 Data Availability von MySpace

In Kooperation mit Yahoo, eBay, Photobucket und Twitter hat auch MySpace angekündigt, Nutzerdaten außerhalb seines Netzwerks zur Verfügung zu stellen. Entwickler sollen Zugriff auf die Profildaten des Nutzers, Fotos und die Kontaktliste erhalten. Den umfassenden Ankündigungen folgte bisher nur eine spezielle REST API, die sich weitestgehend an die Spezifikationen von Open-Social anlehnt (Stand: 03.10.2008 - vgl. [117]).

4.3.4 Windows Live

Bei vielen APIs zum Austausch von Kontaktdaten existieren bisher nur die Ankündigungen der Betreiber. Für WindowsLive ist hingegen eine API implementiert, die sowohl von den Entwicklern selber verwendet werden kann als auch von den großen Sozialen Netzwerken bereits verwendet wird (vgl. [20, 137]). So können Nutzer von Facebook, Bebo, Hi5, Tagged und LinkedIn über die API direkt auf die eigenen Kontakte zugreifen.

Die API ist nach dem REST Prinzip aufgesetzt. Es können Kontaktdaten ausgelesen, verändert, neu hinzugefügt oder gelöscht werden. Diese Funktionalitäten stehen ebenfalls für Fotos zur Verfügung, wobei hier der Zugriff entweder über das WebDAV (Web-based Distributed Authoring and Versioning) oder ATOM (Atom Syndication Format) Protokoll stattfindet. Mit der „Applikation-Based Storage“ bietet WindowsLive eine Möglichkeit an, Nutzereinstellung zentral zu speichern und diese dann wiederum über das WebDAV oder ATOM Protokoll abzurufen. Auf den Entwicklerseiten steht ein Editor zur Verfügung, mit dem die verschiedenen Funktionalitäten ausprobiert werden können und das entsprechende Feedback ausgewertet werden kann. Für die Verknüpfung von Accounts bietet Windows Live den Service „invite2messenger“ an, mit dem Nutzer ihre Freunde aus Facebook, Windows Live Hotmail, Bebo, LinkedIn, hi5 und Tagged direkt zum Windows Live Messenger einladen können¹¹.

4.3.5 Yahoo! Address Book API

Yahoo bietet mit seinem Service „Adress Book API“ ebenfalls Zugriff auf die Adressbücher des jeweiligen Nutzers an (vgl. [173]). Die Daten können mit Plaxo¹² synchronisiert und mit dem eigenen Account bei LinkedIn verknüpft werden (vgl. [51]).

Die Authentifizierung wird über den Yahooeigenen Service „Browser Based Authentication“ abgewickelt (vgl. [176]). In Zukunft soll auch das OAuth Verfahren unterstützt werden (vgl. [51]). Neben dem einfachen Zugriff auf die Daten im Adressbuch stehen auch Funktionalitäten für die Suche im Adressbuch sowie den Zugriff auf einzelne Entities zur Verfügung. Entwickler können so z.B. auch Adressgruppen abfragen. Die Antworten der API erfolgen im XML oder JSON Format. Für den Abgleich mit anderen Programmen, die ein Adressbuch integrieren, steht außerdem eine Synchronisierungsfunktion zur Verfügung.

¹¹Tatsächlich implementiert war diese Funktion am 09.06.2008 nur bei Facebook.

¹²Soziales Netzwerk: <http://www.plaxo.com/>

4.3.6 Zusammenfassung

Die Grundidee ist bei allen Netzwerken gleich: Die Anbieter möchten ihr soziales Netzwerke zu dem zentralen Datenpool für die Benutzer werden lassen. Das eigene soziale Netzwerk soll die einzige Stelle sein, an der der Nutzer seine Daten pflegt und aktualisiert. Neben den eigentlichen personenbezogenen Daten sind der zweite wichtige Faktor die Kontakte der Nutzer. Nachdem es über die Applikations-APIs teilweise möglich war, Kontaktdaten auszulesen, bieten die Sozialen Netzwerke vermehrt offizielle APIs an, um diese Daten auch direkt verknüpft auf anderen Webseiten zu verwenden.

Der Nutzer muss sich hierbei entscheiden, in welchem Netzwerke er seine Daten letztlich verwaltet. Funktional sind die APIs nur bedingt vergleichbar, da bisher oft nur eine Ankündigung existiert (Stand: 07.10.2008). Am weitesten implementiert sind die APIs von Windows Live und Yahoo! Adress Book API. Besonders die Synchronisationsfunktion bei Yahoo sticht aus den gängigen Funktionen heraus. Zusammenfassend lässt sich sagen, dass es zur Zeit in diesem Bereich eine hohe Dynamik gibt, der Trend aber ansonsten recht einheitlich in Richtung einer bedingten Öffnung der Plattformen zeigt.

Bezüglich der Relevanz der einzelnen APIs gilt es zu bedenken, dass momentan nur OpenSocial und die Facebook API über eine signifikante Reichweite verfügen. Für OpenSocial ist dies zum einen durch die Marktposition von Google und zum anderen durch die Anzahl und Bekanntheit der Partner zu begründen. Zwar ist Facebook nicht das größte Soziale Netzwerk, es verfügt aber unter anderem auf Grund der Pionierstellung bei der API Entwicklung über weitreichende Bekanntheit. Die Einschätzung hinsichtlich der Bedeutung von OpenSocial wird unterstützt durch den Trend vieler Sozialer Netzwerke keine eigene API mehr zu entwickeln, sondern Googles OpenSocial Plattform zu implementieren.

Es ist aber davon auszugehen, dass die Betreiber ihr Bemühen verstärken werden, Daten nicht mehr zum Export freizugeben, sondern nur über von ihnen kontrollierte Wege für externe Seiten zu öffnen. Dies birgt auf der einen Seite den Vorteil, dass überall simultan die aktuellen Daten zur Verfügung stehen, aber erlaubt dem Nutzer auf der anderen Seite nur wenig Handhabe über seine Daten.

5 FRAMEWORKS FÜR SOZIALE NETZWERKE

Im bisherigen Verlauf der Arbeit wurden bestehende öffentliche Soziale Netzwerke mit ihren Funktionalitäten und APIs betrachtet. Hierdurch konnte eine Evaluierungsmatrix für Funktionalitäten entwickelt werden. Nachdem die Matrix auf öffentliche bestehende Netzwerke angewendet werden konnte, werden in diesem Kapitel Frameworks evaluiert, die den Aufbau eines eigenen Sozialen Netzwerks ermöglichen. Hierfür wird zuerst eine grobe Kategorisierung der Netzwerke vorgenommen. Anschließend werden die bei der Auswahl der Frameworks angewendeten Kriterien erläutert. Aufbauend auf diesen Kriterien werden die gewählten Frameworks kurz vorgestellt. Die Evaluierung findet mit der bereits entwickelten Evaluierungsmatrix statt und wird um einige Bewertungskriterien erweitert. Ein Schwerpunkt liegt auf den Möglichkeiten der APIs. Den Abschluss des Kapitels bildet die detaillierte Evaluierung sowie eine Zusammenfassung der Ergebnisse.

5.1 KATEGORIEN VON FRAMEWORKS

Nachdem sich Soziale Netzwerke im öffentlichen Bereich immer mehr verbreitet haben, stellt sich auch für viele Firmen die Frage, ob sich das Potential entweder im eigenen Unternehmen oder zur Zielgruppenbindung nutzen lässt. Viele Gruppen, beispielsweise Automobilhersteller, betreiben schon seit längerem Communities, die hauptsächlich auf Foren basieren. Für den Hersteller ist es ein Ziel, die Nutzer über die Vielzahl von Möglichkeiten der Sozialen Netzwerke - insbesondere die gutsortierten und gepflegten Daten - stärker und zielgerichteter an das eigene Unternehmen zu binden.

Für eine erste Kategorisierung der Frameworks wurden diese in drei Gruppen eingeteilt. Folgende drei Möglichkeiten stehen für den Aufbau eines eigenen Sozialen Netzwerks zur Verfügung:

(1) Die erste Möglichkeit ist, sich in ein bestehendes Netzwerk zu integrieren und über die Möglichkeiten der API den Aufbau einer eigenen Community zu fördern.

(2) Die zweite Möglichkeit besteht darin, einen der vielen Services zu nutzen, bei denen das Netzwerk als SaaS (Software as a Service) über eine grafische Benutzeroberfläche verwaltet werden kann. Je nach Ausgestaltungsspielraum treten diese dann wirklich als selbständiges Netzwerk auf oder verhalten sich eher wie eine große Gruppe innerhalb des bestehenden Sozialen Netzwerks.

(3) Die dritte Möglichkeit und damit die maximale Freiheit erhält der Entwickler, wenn eine eigenständige Software verwendet wird, um ein Soziales Netzwerk aufzubauen. Die Installation der Software kann in der Regel entweder im eigenen Rechenzentrum oder als SaaS erfolgen.

Die Möglichkeiten des ersten Ansatzes, sich in ein bestehendes Netzwerk über Gruppen, Netzwerke oder die API zu integrieren, wurden in den vorangegangenen Kapiteln analysiert. Bei der zweiten Variante ist der Betreiber sehr stark auf den Hersteller angewiesen, da weitergehende Veränderungen nur von diesem durchgeführt werden können. Aus diesem Grund soll der Fokus auf Frameworks der dritten Art liegen. Einen kurzen Einblick in die Vor- und Nachteile der drei Ansätze zeigt Lee Aase auf (vgl. [1]).

5.2 KRITERIEN FÜR DIE AUSWAHL DER FRAMEWORKS

Die Auswahl aus der Vielzahl an Frameworks gestaltete sich sehr aufwendig. Um eine möglichst effiziente und fundierte Auswahl der Frameworks treffen zu können, wurden dafür grundlegende Kriterien definiert:

(1) Grundsätzlich wurden Netzwerke bevorzugt, die die Möglichkeit einer Installation im eigenen Rechenzentrum bieten. Dies leitet sich aus dem Problem der Datensicherheit und der Frage, wem die Daten gehören, ab¹. Zwar sichern die meisten Betreiber den Kunden zu, dass die Rechte an den Daten immer bei ihm verbleiben. Allerdings kann der Entwickler bei einer fremdinstallierten Software immer nur auf das Versprechen des Herstellers hinsichtlich der Datensicherheit vertrauen und hat keine Möglichkeit diese zu überprüfen. Insbesondere bei der Installation innerhalb von Unternehmen und Vereinen sollen aber keine Daten nach außen dringen.

(2) Wie schon beim Vergleich der öffentlichen Sozialen Netzwerke konzentriert sich auch dieser Vergleich auf Frameworks, die eine „All-in-One“ Lösung bieten. Zwar gliedern sich viele Produkte wieder in die einzelnen Funktionen, aber auch das Zusammenspiel der einzelnen Funktionen spielt eine wichtige Rolle und fließt in die Bewertung mit ein.

(3) Ein Hauptgrund für den Aufbau eines eigenen Sozialen Netzwerks ist die Möglichkeit, den Auftritt frei anpassen zu können. Hinzu kommt oftmals der Wunsch bestimmte Funktionalitäten zusätzlich zu implementieren. Aus diesen Gründen müssen die Netzwerke so weit wie möglich anpassbar sein. Dieses Kriterium ist eng mit der Möglichkeit einer eigenen Installation verknüpft. Neben der Möglichkeit das Layout anzupassen, wurde es positiv bewertet, wenn außerdem die Möglichkeit existiert, über eine API eigene Applikationen zu erstellen.

(4) Als ein weiteres Kriterium wurde die Existenz einer ausführlichen Dokumentation herangezogen. Einige Hersteller, wie z.B. Sparta Social Networks² werben mit Features wie eine OpenSocial Integration auf Ihrer Webseite, aber bei intensiverer Recherche sind auch direkt vom Hersteller kaum Informationen oder nähere Angaben zu den beworbenen Features zu bekommen. Eine Dokumentation ersetzt noch keine Funktionsüberprüfung, aber ohne diese lassen sich viele Eigenschaften gar nicht oder nur unzureichend überprüfen.

(5) Zuletzt wurden nur Frameworks mit einer gewissen Bedeutung im Markt evaluiert. Dies gestaltet sich schwierig, da der Markt noch sehr unübersichtlich ist. Um aber eine möglichst fundierte Auswahl zu treffen, wurde die Größe des Unternehmens und Anzahl und Reputation von Kunden, soweit diese verfügbar waren, herangezogen. Oft war aber gerade die Angabe der Referenzkunden nur schwer zu überprüfen, da nur der Kunde an sich genannt wird, aber nicht das konkrete Projekt.

Die vollständige Liste der in Betracht gezogenen Frameworks findet sich in Tabelle A.16 auf Seite 121.

5.3 AUSWAHL DER FRAMEWORKS FÜR DIE ANALYSE

Ähnlich wie zu Beginn der Content Management System (CMS) Ära gibt es zahlreiche Anbieter von Frameworks für Soziale Netzwerke („White Label Social Networks“, „Private label Social Networks“ oder „Social Network Engines“). Neben den eigentlichen Anbietern, deren Software von Anfang an für die Erstellung eines Sozialen Netzwerks geschrieben wurde, gibt es zahlreiche CMS Systeme wie z.B. Drupal³, die zusätzliche Funktionen für Soziale Netzwerke anbieten. Zur Auswahl für die Untersuchung standen über 35 Frameworks. Zusammengetragen wurden die Netzwerke im Rahmen einer ausführlichen Recherche. Eine nur sehr unvollständige Übersicht bietet Wikipedia an (vgl. [161]). Eine Linkliste mit kurzen Kommentaren über zwölf OpenSource Lösungen findet sich bei Enzo Silva (vgl. [143]) und Vivalogo Resources listet weitere Frameworks (vgl. [155]). Die umfangreichste Aufzählung von Frameworks - mit Hilfe von vielen Kommentaren - bietet Jeremiah Owyang (vgl. [126]). Für eine Untersuchung im Rahmen seiner Tätigkeit für

¹Ein vollständiger Zugriff auf die Daten kann für den Betreiber nur gewährleistet sein, wenn die Software in seinem eigenen Rechenzentrum betrieben wird.

²<http://www.spartasocialnetworks.com/>

³<http://www.drupal.org>

Forrester Research betrachtet Owyang 34 Netzwerke (vgl. [128]). In einem weiteren Artikel propagiert er zugleich den Gedanken, dass es in nicht all zu ferner Zukunft zu einer Konsolidierung des Markts kommen wird (vgl. [127]). Bereits eine Untersuchung in zwei Teilen hat Techcrunch durchgeführt (vgl. [67, 68, 69, 70]). Auch auf Konferenzen sind die Hersteller der Frameworks (vgl. [147]) zu finden. Einen weiteren knappen Vergleich zwischen verschiedenen Frameworks liefert auch Robert Basis (vgl. [6]).

Mit dem im vorigen Abschnitt definierten Raster stimmen besonders die folgenden Frameworks überein:

- (1) Peopleaggregator von Broadband Mechanis⁴
- (2) Clearspace Community von Jive Software⁵
- (3) Community Server von Telligent⁶
- (4) Lotus Connections von IBM⁷
- (5) Ning von der gleichnamigen Firma⁸

Zusätzlich zu den oben genannten Kriterien überzeugten die Netzwerke jeweils mit einigen Besonderheiten, welche sie für den Vergleich interessant machten:

(1) Peoplaggregator zeichnete sich als PHP Lösung aus, die mit einer großen Offenheit gegenüber Standards wirbt und vollständigen Zugang zum Quellcode bietet. Dies bietet die Möglichkeit, auch externe Dienste einzubinden, bzw. für den Login und Datenimport zu verwenden. Im Unterschied zu vielen anderen Lösungen sind die Hardwareanforderungen auch eher niedrig angesetzt.

(2) Jive Software mit seinem Produkt Clearspace Community gehört sicherlich zu den größeren Firmen auf diesem Gebiet (vgl. [83]). Auch die namhafte Liste von Kunden unterstützt diese These (vgl. [88]). Hinzu kommen sehr gute Evaluationsmöglichkeiten. So kann eine Testversion - zwar mit begrenzter Nutzerzahl - mit unbegrenzter Laufzeit heruntergeladen werden. Auch eine Onlineinstallation der auf Java basierenden Software steht zur Verfügung. Positiv sind auch die zahlreichen Funktionalitäten der API. Mit den zwei Produkten Clearspace (Enterprise) und Clearspace Community bietet Clearspace zudem für die jeweilige Zielgruppe zugeschnittenes Produkt.

(3) Der Community Server als .NET Lösung bietet ebenfalls eine Probeinstallation und sehr gute Referenzkunden (vgl. [27]). So stellt Telligent die Software für die Foren bei MySpace bereit. Besonders interessant an diesem Produkt ist die Integration von Microsoft Sharepoint, das selber zwar auch im Bereich Soziale Netzwerke aktiver werden möchte, aber hier eher noch am Anfang steht (vgl. [13, 64]). Zwar hat das Produkt, wie der Name nahe legt, seinen Ursprung bei den Community Funktionalitäten, allerdings verfügt es inzwischen auch über weitgehende Funktionen, die es als Soziales Netzwerk qualifizieren.

(4) Ebenfalls auf Java basierend bietet IBM das Produkt Lotus Connections an. Traditionell werden die Lotus Produkte in großen Konzernen eingesetzt. Hier ist besonders die An- und Einbindung in die bestehende Produktlandschaft bei dem Unternehmen von besonderem Interesse. Im Vordergrund bei der Produktbeschreibung steht hier die Produktivitätsförderung, die durch den Einsatz von Lotus Connections möglich ist. Nicht zuletzt war die Bedeutung im Softwareumfeld von IBM und damit Lotus Connections ein Argument, es in den Vergleich mitaufzunehmen. Dies ergab u.a. ein Direktvergleich mit Microsoft Sharepoint auf der Enterprise 2.0 Konferenz (vgl. [65]).

⁴<http://www.broadbandmechanics.com/>

⁵<http://www.jivesoftware.com/products/clearspace-community>

⁶<http://communityserver.com/>

⁷<http://www-306.ibm.com/software/lotus/products/connections/>

⁸<http://www.ning.com/>

(5) Zwar sollte der Fokus auf Netzwerken liegen, die eine Selbstinstallation ermöglichen, aber Ning bietet - trotz SaaS - weitreichende Möglichkeiten die Plattform den eigenen Bedürfnissen anzupassen. So bekommt der Entwickler Zugriff auf den Quellcode und kann diesen auch an die eigenen Bedürfnisse anpassen. Als einziger Vertreter im Testfeld ist Ning in der Basisversion ohne Gebühren nutzbar⁹. Aus diesen Gründen und um einen Einblick in alternative Ansätze zu gewährleisten, wurde das Produkt der Ning Inc. ebenfalls in den Testkanon aufgenommen.

5.4 KRITERIEN DER ANALYSE

Aufbauend auf der Auswahl der zu betrachteten Frameworks sollen die Kriterien für den Vergleich festgelegt werden. Grundlage für diese ist die in Kapitel zwei entwickelte Evaluierungsmatrix. Dies sichert eine Vergleichbarkeit der beiden Analysen. Hinzugefügt wurden Kriterien, die entweder direkt für die Implementierung wichtig sind oder im geschäftlichen Umfeld - welches oft der Einsatz für eine eigene Implementierung sein kann - von besonderer Bedeutung sind. Nur die neu hinzugefügten Kriterien werden in diesem Kapitel erläutert¹⁰.

- **Applikationen und APIs** (Definition in Kapitel 3.2.2 auf Seite 21)
 - API (Import)
 - API (Export)
 - Widgets / Applikationen
 - Importmöglichkeiten von Adressen
- **Blog** (Definition in Kapitel 3.2.3 auf Seite 22)
 - Blog führen
 - Blog abonnieren
- **Suche** (Definition in Kapitel 3.2.11 auf Seite 26)
 - Funktionalitäten der Suche
- **Kalender & Aktionen** (Definition in Kapitel 3.2.9 auf Seite 25)
 - Verwalten von Aktivitäten und Aufgaben
 - Kalender- und Terminverwaltung
- **Profil** (Definition in Kapitel 3.2.10 auf Seite 25)
 - Modifikationsmöglichkeiten Profil
 - Aktivitätsanzeige
- **Datenfreigabe & Privacy** (Definition in Kapitel 3.2.4 auf Seite 22)
 - Granularität der Einstellungen auf Elementebene
 - Granularität der Einstellungen auf Personenebene
 - Security
- **Gruppen** (Communities) (Definition in Kapitel 3.2.7 auf Seite 24)

⁹In der kostenlosen Basisversion enthält das eigene Netzwerk Brandings - diese können allerdings gegen monatliche Gebühren entfernt werden.

¹⁰Die neuen Kriterien sind zur besseren Unterscheidung kursiv gesetzt.

- Gruppen
- Netzwerke
- Eventzentrierte Gruppen
- **Nachrichten** (Definition in Kapitel 3.2.8 auf Seite 25)
 - Nachrichten verschicken
 - Nachrichten mit Multimedialen Inhalten anreichern
- **Tagging** (Definition in Kapitel 3.2.12 auf Seite 26)
 - Datentagging
- **Accountfunktionalitäten** (Definition in Kapitel 5.4)
 - Registrierungsinformationen
 - Login
 - Hinterlegbare Daten
 - *Reputation aufbauen*
- **Inhalte** (Definition in Kapitel 5.4 auf der nächsten Seite)
 - *Import/Export von Inhalten*
 - *Versioning, Backup / Wiederherstellung*
 - *Inhalts-Workflow*
 - *Verschiedene Arten von Inhalten*
 - *Zugriffskontrolle*
- **Implementierung** (Definition in Kapitel 5.4 auf der nächsten Seite)
 - *Anforderungen*
 - *Hosting*
 - *Lizenz & Kosten*
 - *Skalierbarkeit*
 - *Dokumentation*
- **Kollaboration** (Definition in Kapitel 5.4 auf der nächsten Seite)
 - *Inhalte mit anderen zusammen verfassen*
 - *Kommentare & Bewertungsmöglichkeiten*

Accountfunktionalitäten Die grundlegenden Kriterien für die Accountfunktionalitäten wurden in Kapitel 5.4 erarbeitet. Für den Unternehmens Einsatz wurde als weiteres Kriterium „Reputation“¹¹ hinzugefügt. Hierunter wird verstanden, inwiefern es möglich ist, mit den Tools der Frameworks eine Einschätzung der Nutzerreputation zu ermöglichen.

¹¹Reputation bezeichnet in dieser Arbeit die Angabe darüber, welchen Ruf jemand in einem Sozialen Netzwerke genießt. Dies kann sich an sehr unterschiedlichen Kriterien wie beispielsweise Aktivität oder Inhalt seiner Äußerungen festmachen.

Inhalte Für die Untersuchung bestehender Sozialer Netzwerke wurde jeder Inhaltstyp (Forumbeitrag, Blogbeitrag usw.) als eigenes Kriterium herangezogen. Verfügen die bestehenden Sozialen Netzwerke in der Regel über weniger Inhaltstypen, dafür aber über spezifischere Funktionalitäten, stellt sich dies bei den Frameworks anders da. Für den Betreiber steht zumeist die Dokumentenvielfalt und nicht der Funktionsreichtum einer bestimmten Inhaltsform im Vordergrund. Aus diesem Grund werden die Inhaltstypen im kommenden Vergleich als ein Punkt zusammengefasst.

Bei Im- und Export von Inhalten wurde vorrangig bewertet, ob und wie es möglich ist, Inhalte in das System zu integrieren bzw. zu erstellen. Zusätzlich wurde untersucht, ob die Möglichkeit besteht alte Versionen von Dokumenten wieder herzustellen. Eine weitere wichtige Funktion im Unternehmensfeld stellen Workflows da, beispielsweise um Inhalte erst nach einer Freigabe sichtbar zu machen. Welche Arten von Inhalten überhaupt verfügbar sind, wurde im Punkt „Verschiedene Arten von Inhalten“ bewertet. Sollen Inhalte geschützt werden, muss das Framework Maßnahmen zum Schutz von Dokumenten anbieten.

Implementierung Von großer Bedeutung bei der Inbetriebnahme eines eigenen Sozialen Netzwerks sind die Anforderungen, die an die Implementierung gestellt werden. Zuerst muss die grundsätzliche Entscheidung getroffen werden, ob das System selber gehostet werden soll oder auf die Dienstleistung eines Providers zurückgegriffen wird. Einige Hersteller der Frameworks bieten auch beide Optionen an. Fällt die Entscheidung zu Gunsten einer eigenen Installation, sind die Anforderungen an die Hostingplattform von besonderer Bedeutung. Neben den Leistungsparameter, bei denen dann auch die Skalierbarkeit eine Rolle spielt, ist hier insbesondere die Unterstützung verschiedener Systeme - auch bei den Datenbanken - von großem Interesse, um eine Installation in der schon vorhandenen Umgebung zu ermöglichen. Von nicht zu unterschätzender Bedeutung ist die Verfügbarkeit einer ausführlichen und vollständigen Dokumentation, die im Optimalfall auch Beispiele für eigenen Applikationen und Veränderungen bietet.

Kollaboration Während die Kollaboration¹² bei den klassischen öffentlichen Netzwerken faktisch keine Rolle spielt, ist sie oft ein von den Herstellern der Frameworks hervorgehobene Funktionalität. Wichtigster Punkt bei der Kollaboration ist die Möglichkeit, Inhalte mit anderen Personen zusammen zu erstellen. Dies kann sowohl synchron als auch asynchron geschehen. Um sich einen Überblick über die Qualität der Beiträge zu verschaffen, sind Kommentare und andere Bewertungsmöglichkeiten von großer Hilfe. Bei diesem Kriterium wurde untersucht, inwieweit es möglich ist Bewertungen vorzunehmen und welche Rolle diese bei der Anzeige von Inhalten spielen.

5.5 EVALUIERUNG DER FRAMEWORKS

Nach der Auswahl der Frameworks und einer leichten Modifizierung der Evaluierungsmatrix wird in diesem Abschnitt die Analyse durchgeführt. Der ausformulierte Vergleich ist dabei nach Softwarelösungen gegliedert, wohingegen die Tabellen funktionspezifische aufgebaut sind. Die Betrachtungen der APIs sind nicht separat aufgeführt, wie dies bei den öffentlichen Sozialen Netzwerken der Fall war, sondern werden zusammen mit dem jeweiligen Framework evaluiert.

¹²Kollaboration bezeichnet die Möglichkeit mehrere Akteure zeitgleich Inhalte zu bearbeiten oder sie in Workflowprozesse zu organisieren.

5.5.1 Peopleaggregator

Das Framework von Broadband Mechanis heißt „Peopleaggregator“. Die Software auf PHP Basis fungiert sowohl als White Label Network als auch als Multi-Framework. So kann, wie bei der Beispielapplikation zu sehen¹³, jeder Nutzer ein eigenes Netzwerk erstellen. Bei der Erstellung kann gewählt werden, ob das Netzwerk öffentlich zugänglich sein soll oder nicht. Positiv zu bewerten ist die Einbindung verschiedener externer ID Dienste¹⁴, die als alternative Registrierungsform zur Verfügung stehen.

Tags Durch das gesamte Framework zieht sich die Möglichkeit, Tags zu vergeben, wobei jedoch die Implementierung der Taggingfunktionalität verbesserungswürdig ist. So fehlt z.B. eine automatisch Ergänzung bei der Eingabe. Auch auf optisch eingängige Tagclouds wird verzichtet, sondern lediglich mit einer Zahl die Häufigkeit visualisiert.

Profil Im Nutzerprofil können zahlreiche Daten von der eigenen Religion bis zu einem Lebenslauf hinterlegt werden. Auch hier findet sich das Taggingsystem wieder; der Nutzer kann sich selber mit Tags versehen. Um den Zugriff der anderen Nutzer auf die eigenen Profildaten zu begrenzen, kann für jedes einzelne Datum festgelegt werden, ob es nur einen selber, den eigenen Kontakten, jedem oder der „Familie“¹⁵ zugänglich sein soll. Explizite Sicherheitsfeatures bietet Broadband Mechanics nicht an, aber bei der Registrierung sollte ein CAPTCHA Spambots fernhalten.

Blogs Jeder Nutzer hat die Möglichkeit, einen Blog zu führen. Beim Veröffentlichen kann er auswählen, ob sein Beitrag nur im eigenen oder auch in den Blogs seiner Gruppen veröffentlicht werden soll. Formatierungen des Inhalts werden durch den TinyMCE Editor ermöglicht. Der Blog kann per RSS abonniert werden.

Applikationen und APIs Broadband Mechanis versteht sich als Vorreiter bei den Möglichkeiten, sich mit anderen Diensten zu verknüpfen (vgl. [23]). In Ansätzen ist dies durchaus realisiert. So werden zwei Arten von APIs angeboten, zum einen eine Web Service API (vgl. [24]) mit der direkt Netzwerkelemente angesteuert werden können, zum anderen die Möglichkeit, direkt mit PHP Befehlen zu arbeiten (vgl. [21]). Allerdings ist die Dokumentation zu den APIs mangelhaft und die Funktionen sind zumeist gar nicht erläutert. Auf der Seite des Endanwenders ist die Situation etwas besser. Hier steht ein RSS Import zur Verfügung, mit dem der Nutzer eigene Datenquellen einbinden kann. Dies kann er ebenfalls über so genannte Widgets tun, die sich von den Nutzern direkt im Browser erstellen lassen. Außerdem wird der Export und Import von „hCards“ angeboten. Dem Nutzer bietet sich außerdem auch die Möglichkeit, Kontakte von Facebook, AIM und Flickr zu importieren¹⁶. In einer der nächsten Versionen soll zusätzlich FOAF und XFN implementiert werden.

¹³<http://peopleaggregator.net/>

¹⁴score.com, openID, flickr

¹⁵Dies ist eine Möglichkeit Kontakte zu einer Gruppe zusammenzufassen.

¹⁶Sowohl auf dem offiziellen Testsystem als auch auf einer eigenen Installation war es nicht möglich diese Funktionen fehlerfrei zu verwenden.

Netzwerkstruktur Um Nutzer zu kategorisieren bietet Broadband Mechanis die Möglichkeit an, ein eigenes Netzwerk zu erstellen. Die nächste Ebene sind Gruppen, die über zahlreiche Funktionalitäten verfügen. Zusätzlich kann der Nutzer einer Gruppe ein eigenes Aussehen geben und neben einer Mailingliste auch die Sichtbarkeitseinstellungen festlegen. Gruppen werden wiederum in Kategorien eingeordnet und können mit Icon und Beschreibung versehen werden. Zwar kann eine Gruppe mit einem Termin versehen werden, aber ein Feature, um Funktionen zu einem Termin zuzuordnen, steht nicht zur Verfügung. Bei der Terminvergabe kann zwischen drei Termintypen (Allgemein, Konferenz und Konzert) gewählt werden. Je nach Terminart stehen verschiedene Metaangaben zur Verfügung.

Suche Für die gezielte Suche steht nur eine sehr eingeschränkte Funktionalität zur Verfügung. Bis auf das Datum kann keine Einschränkung vorgenommen werden, eine gezielte Suche ist daher mit der momentanen Funktionalität nicht möglich.

Kollaboration Ein Import und Export von Dokumenten oder Daten - außer für den Lebenslauf - ist nicht möglich. Allerdings kann der Nutzer zahlreiche Metainformationen bei der Erstellung von Inhalten vergeben. Funktionen für die Versionierung und Workflows sind nicht vorhanden. Zur Verfügung stehen Funktionalitäten um Textbeiträge, Video- und Audioobjekte einzustellen. Zugriffsbeschränkungen sind nicht implementiert, dafür kann der Nutzer aber bei jedem Upload auswählen, unter welche Lizenz er das Video- oder Audioobjekt stellt.

Die Verbindung zu anderen Kontakten erfolgt anhand einer fünfstufigen Skala. Zusätzlich kann ein Kontakt den Status „Familie“ bekommen, womit ihm erweiterte Zugriffsrechte zugewiesen werden. Die Einstufung in eine der fünf Kategorien hat keinen Einfluss auf Funktionalitäten¹⁷. Nachrichten können nur an Kontakte geschickt werden und bestehen nur aus Text. Dafür kann der Nutzer aber in seiner eigenen Mailbox Ordner anlegen und eine Konversationsansicht aktivieren.

Über spezielle Tools zur Kollaboration verfügt das Framework nicht. Über eine Kommentar- oder die Bewertungsfunktion können die Inhalte mit einer Einschätzung versehen werden. Außer dem Aus- und Einblenden von Informationen stehen dem Nutzer auch keine Modifikationsmöglichkeiten für das eigene Profil zur Verfügung. Über „Neuste Beiträge“ kann er zwar sehen, welche Beiträge vor kurzem geschrieben wurden, aber eine ausgefeilte Benachrichtigungsfunktionalität über die Aktivitäten von Kontakten beinhaltet das System nicht.

Implementierung Als Mindestvoraussetzung für das Betreiben der Plattform ist ein LAMP System auf einem Unixbetriebssystem mit PHP5 notwendig. Laut Angaben des Herstellers ist das System für über 5 Millionen Nutzer entwickelt¹⁸. Die Lizenzierung unterscheidet zwischen einem geschäftlichen und einem nicht-kommerziellen Einsatz. Der nicht-kommerzielle Einsatz ist kostenlos. Für den geschäftlichen Einsatz richten sich die Kosten nach Größe der Community und liegen zwischen 2.500 und 20.000\$. Der Quellcode kann von der Webseite herunter geladen und beliebig modifiziert werden¹⁹. Eine Übersicht über die Bewertungen nach Funktionen findet sich in den Tabellen A.17 auf Seite 123ff.

¹⁷Beispielsweise ist hier eine Auswirkung auf die Privacyeinstellungen denkbar.

¹⁸Die Aussage basiert auf Grundlage einer Nachfrage per Email an den Hersteller.

¹⁹Die Software steht unter keiner bestimmten Lizenz. Es wird lediglich erwähnt, dass der Quellcode modifiziert werden darf.

Zusammenfassung Peopleaggregator deckt mit seinen Funktionalitäten die gängigen Möglichkeiten von Sozialen Netzwerken ab. Leider funktionieren nicht alle Bereiche fehlerfrei. So gibt es zwar die Möglichkeit, Daten aus Facebook zu beziehen, als Statusmeldung verbleibt aber dennoch die Nachricht, dass dort keine Daten vorliegen würden. Dieser Eindruck setzt sich auch bei den APIs fort. Zwar gibt es diese, aber die Dokumentation ist sehr mangelhaft, so dass nur schwer zu überprüfen ist, ob die versprochenen Funktionalitäten auch wirklich verwendbar sind. Die Blogfunktionalität hingegen ist vollständig und umfangreich implementiert. Hervorzuheben ist an dieser Stelle die Möglichkeit, einen eigenen Beitrag gleichzeitig im Gruppenblog zu veröffentlichen. Auch die Einteilung in Gruppen und Netzwerke entspricht dem erwarteten Funktionsumfang und erweist sich als nützlich. Sehr mager fällt die Suche aus, hier wäre mehr Funktionalität wünschenswert und sicherlich auch einfach zu implementieren gewesen. Zwar bietet Peopleaggregator die Möglichkeit an, beim Upload von multimedialen Inhalten eine Lizenz anzugeben, ansonsten stehen aber keine Funktionen zur Verfügung, den Zugriff auf Dokumente zu begrenzen. Eine Anzeige, was welcher Kontakt in der letzten Zeit auf der Plattform gemacht hat, ist leider nicht vorhanden. Leider waren auch keine konkreten Angaben zur Skalierbarkeit des Systems zu erfahren. Zusammenfassend bietet Broadband Mechanis ein Produkt an, welches einen sehr großen Funktionsumfang hat, aber an vielen Stellen unfertig wirkt.

5.5.2 Clearspace Community

Jive Software wurde 2001 gegründet und hat einen beachtlichen Kundestamm vorzuweisen (vgl. [84]). Sie bieten zwei Produkte an (Clearspace und Clearspace Community). In Bezug auf die konkreten Funktionen unterscheiden sich die beiden Produkte fast nicht. Bei Clearspace liegt der Fokus jedoch auf der Kollaboration, wohingegen Clearspace Community auf Nutzergruppen außerhalb fester Firmengrenzen fokussiert ist. Diese Ausrichtung zeigt sich weniger in der Unterschiedlichkeit der Funktionalitäten, als vielmehr darin, dass bestimmte Funktionen in der Communityvariante standardmäßig deaktiviert sind. Da bei dem Vergleich nicht die Kollaboration im Vordergrund stand, wurde als zu untersuchendes Produkte Clearspace Community gewählt²⁰.

Accountfunktionalitäten Für die Registrierung sind ein Loginname, Vor- und Nachname sowie eine Emailadresse Voraussetzung. Allerdings kann auf Wunsch die Authentifizierung auch von einem LDAP (Lightweight Directory Access Protocol) Server abgewickelt werden. Im Backend kann der Administrator beliebige Profelfelder definieren. Für jedes Feld kann festgelegt werden, ob es durchsuchbar, schreibgeschützt oder verpflichtend sein soll. Felder, die von einer externen Datenquelle gefüllt werden sollen, können als solche definiert werden. Ob es möglich ist, die Daten für andere Nutzer unzugänglich zu machen, hängt von den Einstellungen des Administrators ab; generell lässt sich jedoch jedes Feld vom Nutzer verstecken. Dies kann allerdings nicht in einem abgestuften System für unterschiedliche Personenkreise erfolgen. Die zusätzliche Definition, ob ein Feld durchsuchbar sein soll, ermöglicht es dem Nutzer zwar, ein Datum in seinem Profil anzuzeigen, aber nicht über dieses gefunden zu werden. Es wäre aber auch denkbar und möglich ein Feld anzulegen, welches nicht sichtbar ist, extern gepflegt wird, aber von der Suche erfasst werden kann. Damit stehen vielfältige Möglichkeiten zur Verfügung, die Plattform hinsichtlich der Nutzerdaten dem eigenen Nutzungszweck anzupassen.

Netzwerkstruktur Strukturiert werden kann das System mit Hilfe von Communities. Diese können beliebig verschachtelt werden. Neben der Verschachtelung sind auch die Funktionalitäten (Diskussionen, Wikis, Blogs, Community Blog) an die jeweilige Community gekoppelt. Einstellungen für eine Community können beim Erstellen vererbt werden. Auch alle weiteren Inhalte wie Tags etc. werden jeweils einer Community zugeordnet. Eine direkte Implementierung einer Gruppenfunktion gibt es nicht. Abgebildet werden können Gruppen nur über verschachtelte Communities, wobei Communities auch nach der Erstellung zusammen gelegt werden können. Eventzentrierte Gruppen können in der aktuellen Version nicht angelegt werden.

²⁰Evaluert wurde die Version 2.0.5.

Kollaboration und Inhalte Da das Ursprungsprodukt „Clearspace (Enterprise)“ seinen Fokus auf Kollaboration legt, unterstützt auch Clearspace eine Reihe von Kollaborationsfunktionen. So können online erstellte Dokumente von mehreren Personen bearbeitet werden. Inhalte können kommentiert und auf einer fünfstufigen Skala bewertet werden. Die eigene Bewertung kann jederzeit verändert werden, wobei dem Nutzer außerdem die durchschnittliche Bewertung präsentiert wird. Es gibt keine Möglichkeit sich mit anderen Personen zu verbinden, allerdings gibt es die Möglichkeit, Nutzer je nach Aktivität in den verschiedenen Bereichen oder auch manuell einen bestimmten Status zuzuweisen. Hierüber kann in einem bestimmten Maße auch die Reputation des Nutzers abgebildet werden.

Inhalte können als Blogeintrag, Dokumente, Diskussionen, Kommentare oder Fragen online erstellt werden. Der Autor legt dabei fest, inwieweit das Dokument auch von anderen Mitgliedern der Plattform bearbeitet werden darf. Bei den Personengruppen kann zwischen jedem, spezifischen Nutzern und einem selber gewählt werden. Zusätzlich können Personen festgelegt werden, die eine Version des Dokuments genehmigen müssen, bevor es veröffentlicht wird. Die Eingabe erfolgt entweder im Freitext oder über eine Nutzerauswahlliste. Alle Inhalte können auch kommentiert werden. Ob dies erlaubt sein soll, kann ebenfalls bei der Erstellung festgelegt werden. Textdokumente können über einen Rich-Content-Editor formatiert werden. Sowohl das Einfügen von Bildern in den Fließtext als auch das Anhängen von beliebigen Dateien ist erlaubt. Neben dem Onlineerstellen von Dokumenten gibt es die Möglichkeit, beliebige Dateien auf die Plattform hoch zu laden. Blogs der Nutzer können sowohl per RSS als auch per Email abonniert werden, das gleiche gilt für Kommentare. Für jedes Dokument ist automatisch eine Versionisierung vorhanden. Diese zeigt sowohl die verfügbaren Versionen als auch einen Vergleich zwischen ihnen an. Für den schnelleren Einstieg können aus diversen Blogsoftwares Einträge importiert werden, was die Übernahme von älteren Blogs der Nutzer ermöglicht. Für Blogeinträge steht der Rich-Content-Editor für die Formatierung mit denselben Optionen wie für Textdokumente zur Verfügung. Inhalte können auf jeder Seite direkt mit einem Stichwort durchsucht werden. Außerdem bietet die Tagcloud mit dynamischen Wörtergrößen auf vielen Seiten die Möglichkeit direkt nach bestimmten Tags zu filtern.

Suche Über die erweiterte Suche kann die Suche auf Blogs, Diskussionen, Kommentare, Menschen oder bestimmte Communities beschränkt und zusätzlich noch mit einer Zeitangabe versehen werden. Die Ergebnisse werden nach den Inhaltstypen sortiert aufgelistet.

Nachrichten Die interne Nachrichtenfunktion bietet eine Reihe von Funktionalitäten. So kann der Nutzer Nachrichten an andere Nutzer der Plattform versenden, jedoch nicht an externe Emailadressen. Nachrichten können sowohl in vordefinierte Ordner als auch in selber erstellte Ordner einsortiert werden. Als zusätzliche Funktionalität steht dem Nutzer ein Adressbuch zur Verfügung, allerdings muss dieses vom ihm durch die manuelle Eingabe von Nutzernamen selbst gepflegt werden. Für die Benachrichtigung über neue Nachrichten steht neben Email auch ein RSS Feed zur Verfügung. Es ist allerdings nicht möglich, multimediale Inhalte in die Nachrichten einzubetten.

Tags Durch das gesamte System zieht sich die Möglichkeit Inhalte mit Tags zu versehen²¹. Zwar verfügt das System nicht über eine Funktionalität, um bei der Eingabe intelligente Ergänzungen zu empfehlen, allerdings schlägt es basierend auf den häufig in dieser Community verwendeten Tags verschiedene Optionen vor. Außerdem steht dem Nutzer eine Tagcloud mit dynamischen Textgrößen zur Verfügung. Eine extra Ansicht ermöglicht es, sich eine Übersicht über alle existierenden Tags zu verschaffen. Die Tagübersicht kann entweder nach Namen oder nach Häufigkeit sortiert werden, was dann wiederum über die Schriftgröße symbolisiert wird.

²¹Nur bei Announcements ist dies nicht möglich.

Profil Das eigene Profil besteht aus den Profildaten des Nutzers und einem Foto. Der Nutzer selber hat aber keinen Einfluss auf die Gestaltung seines Profils. Das Profil beinhaltet außerdem eine Aktivitätsanzeige, auf der zu sehen ist welche Aktionen der Nutzer zuletzt vorgenommen hat. Neben der nutzerzentrierten Ansicht steht auch eine dokumentenzentrierte Ansicht über alle Dokumente zur Verfügung. Allerdings beschränkt sich die Angabe hier allgemein darauf, welche Dokumente betrachtet wurden.

Applikationen und APIs Clearspace bietet verschiedene Ansätze, das vorhandene System zu verändern. So kann der Nutzer vorgefertigte und selbst erstellte Widgets auf den Seiten platzieren. Diese können HTML, JavaScript und CSS enthalten, wodurch es möglich ist, fast beliebige Inhalte zu platzieren. Zusätzlich stehen dem Entwickler eine Web Services API (SOAP und REST) zur Verfügung. Weiterhin besteht auch die Möglichkeit, direkt die Java Core API anzusteuern. Über diese beiden Wege hat der Entwickler die Möglichkeit, alle Funktionsbereiche der Plattform für sein eigenes Plugin zu verwenden. Die Java Dokumentation ist sehr ausführlich, einführnde Beispiele fehlen allerdings. Auf Wunsch kann die Verbindung zu der WebServices API nur per SSL zugelassen werden. Für die Entwickler steht eine eigene Community zur Verfügung, die sich aber erst im Aufbau zu befinden scheint²². Clearspace bietet die Möglichkeit Daten von alten Versionen zu importieren und wie oben schon erwähnt, alternative Blogformate zu integrieren. Per Mausklick kann eine Visitenkarte im „vcf“ Format exportiert werden. Per RSS kann der Nutzer sich über Änderungen von Inhalten informieren lassen. Diese Funktionen steht sowohl sortiert nach Inhaltstypen als auch personenzentriert zur Verfügung. Für das Nachrichtensystem kann ein Import von bestehenden mbox Dateien erfolgen und News können über das NNTP bezogen werden.

Implementierung Clearspace Community unterstützt eine sehr breite Hardware- und Softwarebasis. Das System auf Java Basis kann entweder im Standalonbetrieb oder integriert in eine bestehende Plattform laufen. Als Mindestanforderungen wird ein Rechner mit 2 DualCore 2 GHZ mit 2GB RAM und 1GB Java Heap benötigt. Als Betriebssystem werden Unix und Windowsbetriebssysteme mit Apache Tomcat, JBoss, BEA WebLogic Server oder IBM WebSphere akzeptiert. Als Datenbankplattform können Mysql, DB2, SQL, Oracle und PostgreSQL eingesetzt werden. Skalierbar ist das System laut Auflistung auf der Webseite bis 25 Millionen Seitenaufrufen oder 100.000 erstellten Seiten im Monat. Die Preise basieren auf der jeweiligen Anzahl der Prozessoren. Die erste CPU kostet ~35.000 €/Jahr und jede weitere ~13.000 €/Jahr.

Zusammenfassung Insgesamt macht Clearspace Community einen sehr guten Eindruck. Dies betrifft sowohl die Verfügbarkeit der Funktionalitäten als auch deren Implementierung. Bei der Anbindung an externe Systeme wird zwar mit LDAP eine externe Datenquelle unterstützt, allerdings wären auch weitere Anbindungen, beispielsweise an OpenID, wünschenswert²³. Hervorzuheben ist in diesem Zusammenhang die freie Definition von Datenfeldern mit den weitreichenden Attributen. Zwar bietet das System keine Gruppenfunktionalität an, aber die beliebige Verschachtelung von Communities ist ein gleichwertiger Ersatz. Hervorzuheben ist die Möglichkeit Communities zu migrieren und bei der Erstellung die Funktionalitäten zu vererben. Sehr konsequent durch das System ziehen sich die zwei Ansätze zur Datenansicht - personenbezogen und dokumentenbezogen. Dies zeigt sich auch bei der Aufgliederung der angebotenen RSS Feeds. Konsequent wurde auch die Implementierung von Tags betrieben, wobei hier z.B. eine automatische Ergänzung wünschenswert wäre. Such- und Nachrichtenfunktion liefern das erwartete Ergebnis, wobei eine etwas komfortablere Möglichkeit der Adresseingabe in das Adressbuch hilfreich wäre. Die einzige Kernfunktion, die Clearspace Community fehlt, ist die Möglichkeit, sich mit anderen Nutzern zu verbinden. Zwar ist es möglich, geschlossene Communities zu erstellen, diese ersetzen eine Verbindungsfunktion aber nur unzureichend. Besonders hervorzuheben ist die weitgehende API, mit der praktische alle Funktionalitäten der Plattform von einem Plugin angesteuert werden können. Zusammenfassend bietet Clearspace fast alles, was von einem Sozialen Netzwerk zu erwarten ist und setzt diese Funktionen auch solide um. Einen direkten Vergleich mit den anderen Frameworks findet sich in den Tabellen A.17 auf Seite 123ff.

²²Es gibt erst sehr wenige Einträge (vgl. <http://www.jivesoftware.com/dev/>).

²³Laut Aussage des Herstellers ist dies für die kommende Version auch geplant.

5.5.3 Ning

Ning ist das einzige Framework in diesem Vergleich, welches nicht die Möglichkeit einer Installation im eigenen Rechenzentrum bietet. Wie in Abschnitt 5.3 auf Seite 45 bereits erläutert gab es aber trotzdem gute Gründe, es in die Analyse mit aufzunehmen, u.a. die kostenlose Nutzung und die Möglichkeit trotz Hostinglösung eine Modifikation des Quellcodes vorzunehmen zu können.

Erstellung des Netzwerks Grundsätzlich werden neue Netzwerke über die Webseite ning.com eingerichtet. Voraussetzung dafür ist nur eine so genannte „NingID“. Mit der einmal eingerichteten ID kann der Nutzer sich bei allen weiteren Ning-Netzwerken anmelden. Der Einrichtungsprozess ist in drei Schritte gegliedert: (1) Grundlegende Angaben über das Netzwerk, (2) Aktivierte Funktionalitäten und (3) Design.

(1) Nachdem der Entwickler einen Namen und eine URI gewählt hat, kann er bestimmen, ob es sich um ein öffentliches oder privates Netzwerk handeln soll. Zusätzlich ist die Angabe von Tags, einer kurzen Beschreibung und Schlüsselwörtern möglich. Als letztes erfolgt die Angabe einer Sprache, wobei 21 Sprachen zur Auswahl stehen.

(2) In einem zweiten Schritt können die grundlegenden Funktionalitäten aktiviert und grafisch angeordnet werden. Neben den klassischen Funktionen (Blog, Video, Foto, Forum, Musik und Termine) können ebenfalls eine Aktivitätsanzeige, eine Mitgliederübersicht, die Beschreibung des Netzwerks, eine Textbox, eine Gruppenübersicht, Ankündigungen und Gadgets²⁴ hinzugefügt werden. Für die Anordnung steht ein klassisches dreispaltiges Design zur Verfügung. Die Position der Anzeigen, die Angabe, wer das Netzwerk erstellt hat, sowie die Anzeige der Nutzerinformationen können nicht verschoben werden. Während die linke Seitenleiste und der mittige Hauptblock sich je nach Seitenanzeige verändern, werden Objekte in der rechten Spalte durchgehend in allen Ansichten gezeigt.

(3) Dem Entwickler stehen über 56 Themes zur Auswahl. Farben, Logo und Schriftarten können über einen grafischen Editor verändert werden. Für eine weitergehende Veränderung des Designs kann auch eigener CSS Code verwendet werden. Somit stehen weitreichende Modifikationsmöglichkeiten zur Verfügung, jedoch kann die dreispaltige Struktur einer Seite nicht verändert werden.

Zusätzlich zu seiner NingID wählt der Entwickler noch ein Masterpasswort, welches für alle Administratortasks benötigt wird.

Profil Um eine NingID anzulegen, muss der Nutzer Name, Geburtstag, E-Mail und ein Passwort angeben. Die Emailadresse dient gleichzeitig als Loginname, alternative Loginmethoden stehen nicht zur Verfügung. Der Betreiber des Sozialen Netzwerks kann beliebige weitere Felder festlegen, die der Nutzer bei der Registrierung für das spezifische Netzwerk ausfüllen muss. Jedes dieser Felder kann als Pflichtfeld definiert werden, außerdem kann dem Nutzer ermöglicht werden, das Feld als privat zu markieren. Es besteht keine Möglichkeit, die Reputation von Nutzern abzubilden oder zu erfassen. Allerdings kann der Administrator bestimmte Nutzer hervorheben, was jedoch nicht an Faktoren wie Aktivität gekoppelt werden kann.

Bei den Profilmodifikationsmöglichkeiten hat Ning sich an den Funktionen von Facebook orientiert, so kann der Nutzer die verschiedenen Inhaltsblöcke in seinem Profil verschieben. Des Weiteren stehen für die verschiedenen Inhaltsblöcke Optionen zur Anzeige zur Verfügung. Beispielsweise kann der Nutzer RSS Feeds zu seiner eigenen Seite hinzufügen. Jeder Nutzer hat seine persönliche Ansicht über die Aktivitäten der letzten Zeit. Dort werden Nachrichten aus dem Netzwerk und über die eigenen Aktivitäten dargestellt, allerdings sind keine Nachrichten über die Aktivitäten der anderen Kontakte vorhanden.

²⁴Dabei handelt es sich um ein kleines Stück Software, welches eine sehr spezifische Aufgabe erledigt.

Inhalte Als Inhaltstypen stehen Blogbeiträge, Fotos, Videos, Kommentare, Musik, Forumsbeiträge und Widgets zur Verfügung. Funktionalitäten, um bestehende Inhalte dauerhaft zu importieren, existieren nicht. Auch eine Versionierung oder Dokumentenwiederherstellung ist nicht vorhanden. Zentral definiert werden kann, dass für Fotos und Videos erst eine Freigabe erforderlich ist. Allerdings erlaubt dieser rudimentäre Workflow nicht, beliebige Nutzer als freigebende Person zu identifizieren, sondern ist eher eine vom Administrator gesteuerte Beschränkung. Musikstücke können entweder nur als Stream oder auch zum Download bereitgestellt werden.

Blogs Jeder Nutzer kann seinen eigenen Blog führen. Beiträge werden mit einem Rich-Content-Editor erstellt, der auch das Einbinden von Bildern ermöglicht. Zusätzlich können Dokumente als Anhang zu Beiträgen gespeichert werden. Blogeinträge können wie Personen ebenfalls besonders hervorgehoben werden. Für jeden Blogeintrag können separat Beschränkungen festgelegt werden, so kann die Kommentierungsfunktion auf jeden, nur Freunde oder den Blogschreiber selber begrenzt werden. Gleiches gilt für die Sichtbarkeit des Beitrags. Kommentare können zusätzlich mit einer Moderierungsfunktionalität versehen werden. Soll ein Beitrag nicht zum Zeitpunkt der Erstellung erscheinen, kann ein Datum mit Uhrzeit als Veröffentlichungsdatum definiert werden. In öffentlichen Netzwerken können die Blogs auch per RSS abonniert werden.

Netzwerkstruktur Ein vom Entwickler erstelltes Soziales Netzwerk kann nicht in weitere Netzwerke unterteilt werden. Allerdings kann der Administrator oder - wenn dies zugelassen ist - der Nutzer weitere Gruppen erstellen. Diese können allerdings nicht hierarchisch angeordnet werden, sondern verbleiben auf derselben Ebene. Bei der Erstellung werden Gruppen mit einem Namen, einer Beschreibung und einem Foto versehen. Auch die Funktionalitäten Forum, Textbox und RSS-Reader können einer Gruppe zugewiesen werden. Zusätzlich können Widgets angezeigt sowie der Zugang zu der Gruppe als privat gesetzt werden. Um alle Mitglieder einer Gruppe auf einmal zu erreichen, steht eine Messagingfunktion zur Verfügung. Diese kann je nach Einstellung von der gesamten Gruppe oder nur vom Administrator genutzt werden. Eventzentrierte Gruppen existieren in Form einer Gästeliste für Termine. Als Funktionalität können Gäste den Termin kommentieren.

Suche Die Suche gestaltet sich sehr schlicht, es ist lediglich die Eingabe eines Stichwortes möglich. Weitere Möglichkeiten zum Einschränken der Suche sind nicht vorgesehen. Positiv zu erwähnen ist, dass die Suche alle Inhaltstypen erfasst.

Nachrichten Die Nachrichtenfunktion ist mit den üblichen Funktionalitäten implementiert. So ist es möglich Nachrichten zu versenden und zu empfangen; zum Speichern stehen verschiedene Ordner zur Verfügung, wobei der Nutzer jedoch keine eignen Ordner erstellen kann. Dafür ist auch der Versand von Nachrichten an Nicht-Mitglieder über die Maske möglich. Es werden nur reine Textnachrichten unterstützt.

Kalender Ning verfügt ebenfalls über die Möglichkeit, Termine einzustellen. Diese werden mit einem Typ (Meeting, normaler Termine usw.) und Ortsangaben versehen. Die Gästeliste kann für die Teilnehmer einsehbar sein. Zusätzlich kann den Eingeladenen die Möglichkeit gegeben werden, ihr Kommen zuzusagen. Jede Gästegruppe verfügt über die Möglichkeit den Termin zu kommentieren. Um die Termine auf dem eigenen Rechner zu verwenden, steht ein Export nach Outlook oder iCal zur Verfügung; es kann aber immer nur ein Termin exportiert werden.

Tags Mit Hilfe der Taggingfunktionalität können Bilder, Videos und Blogeinträge markiert werden. Es besteht jedoch keine Möglichkeit, Personen mit einem Tag zu versehen. Eine tagbezogene Suche kann der Nutzer starten, indem er auf einen Tag klickt. Zu der Standardinstallation gehört allerdings keine Tagcloudanzeige.

Kollaboration Es ist nicht möglich, Inhalte zusammen mit anderen online zu erstellen oder Dokumente eines beliebigen Formats auf der Plattform abzulegen. Als Anlage zu einem Blogeintrag ist dies aber möglich. Eine Wikifunktionalität ist ebenfalls nicht vorhanden. Fotos können über eine fünfstufige Skala bewertet werden. Zusätzlich können Nutzer Kommentare abgeben. Somit besteht die Möglichkeit, einen Eindruck über die Qualität eines Objekts zu bekommen. Allerdings ist dies nicht in der Suchfunktion verankert und entfaltet daher seinen Nutzen nur sehr bedingt.

Applikationen und APIs Die API ist eine der Stärken von Ning. Ohne Programmierkenntnisse kann der Nutzer Textboxen in seinem Profil anzeigen lassen. Der Administrator hat diese Möglichkeit ebenfalls für die Hauptseite. Textboxen ermöglichen es auf einfache Art und Weise, externe Inhalte oder beliebigen Text darzustellen. Sie können reinen Text, HTML, Fotos, Widgets oder JavaScript (nur Administrator) enthalten. Für die einfache Erstellung von Textboxen steht ein Rich-Content-Editor zur Verfügung. Standardmäßig sind Anzeigewidgets für Fotos, Musik und Videos vorhanden. Zusätzlich besteht die Möglichkeit, Gadgets von Google (OpenSocial Gadgets) zu dem eigenen Netzwerk hinzuzufügen. Diese Funktion ist aber noch als Beta gekennzeichnet. Weiterhin bietet Ning die Möglichkeit, Ausschnitte aus dem eigenen Netzwerk direkt in Facebook zu integrieren, um dort auf das eigene Netzwerk aufmerksam zu machen. Per Mausklick können Video- und Audioobjekte sowie eine Fotoapplikation erzeugt werden. So wird ein Update der Inhalte in einem Ning Netzwerk auch automatisch in Facebook sichtbar. Außerdem ist der Import von Flickr²⁵ Fotos möglich.

Über CSS und JavaScript sind somit schon einige weitreichende Änderungen und Anpassungen für das eigene Netzwerk möglich. Für weitergehende Änderungen kann der Betreiber Zugriff auf den Quellcode beantragen, der dann per SFTP (SSH File Transfer Protocol) oder WebDav²⁶ erfolgt. Sobald der Zugriff auf den Quellcode aktiviert ist, spielt Ning nicht mehr automatisch Updates für den Quellcode ein. Der Entwickler bekommt die Updates aber weiterhin zur Verfügung gestellt und kann den Code dann selber hinzufügen. Alternativ besteht die Möglichkeit, den Codezugriff wieder zu sperren und zum Standardcode zurückzukehren, womit dann wieder Updates bereitgestellt werden.

Das Ning System selber beruht auf einer Java Implementierung. Wie auf der Abbildung 5.1 auf der nächsten Seite zu sehen, erfolgt der Zugriff seitens des Entwickler über eine PHP, JavaScript oder REST API.

Der Aufbau der Interaktionselemente erfolgt in Form von Ning Web Widgets (WWF). Sowohl die zu Beginn von Ning bereit gestellten Audio und Videowidgets, als auch die vom Entwickler generierten Widgets integrieren sich auf diese Art und Weise ins System. Die Widgets sind nach dem Model-View-Controller (MVC) Prinzip aufgebaut. Der Aufbau eines Widgets spiegelt sich auch in der URI wieder. Alle URIs haben die folgende Struktur:

`http://site.ning.com/index.php/widget-instance/controller/action`

So würde z.B.

²⁵<http://www.flickr.com/>

²⁶Der WebDav Zugriff befindet sich momentan noch in der Betaphase, ermöglicht aber beispielsweise auch die Übertragung von größeren Dateien als 2 MB, die im SFTP Modus begrenzt ist.

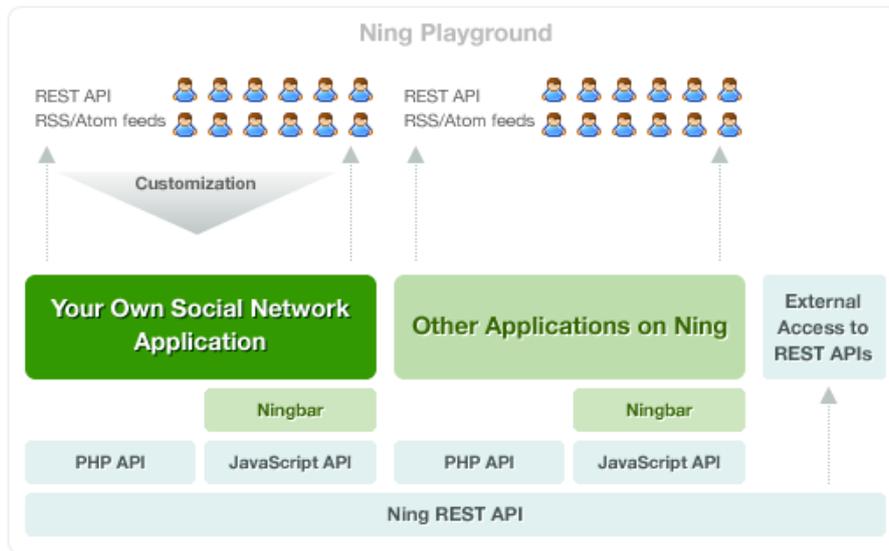


Abb. 5.1: Übersicht über die Ning API (vgl. [120])

<http://site.ning.com/index.php/photos/photo/view>

die „view“ Aktion im „Photo“ Controller in der Widget Instanz „Photos“ aufrufen. Über den Zugriff auf den Quellcode kann der Entwickler sowohl die bestehenden Widgets verändern als auch eigene Widgets erstellen. Der Zugriff auf die Daten der Nutzer erfolgt über den so genannten „Content Store“, dessen Aufbau in Abbildung 5.2 dargestellt ist. Statt einer relationalen Datenbank hat Ning einen objektorientierten Aufbau seiner Datenhaltung gewählt.

Der Zugriff des Entwicklers auf die API erfolgt auf drei verschiedene Arten. Der Zugang zur API mit den meisten Funktionalitäten steht über PHP bereit. Zusätzlich kann auch eine Java API verwendet werden, mit der ebenfalls ein Zugriff auf netzwerkspezifische Daten möglich ist. Über eine REST API besteht zudem die Möglichkeit, von außen Zugriff auf die Daten zu erhalten.

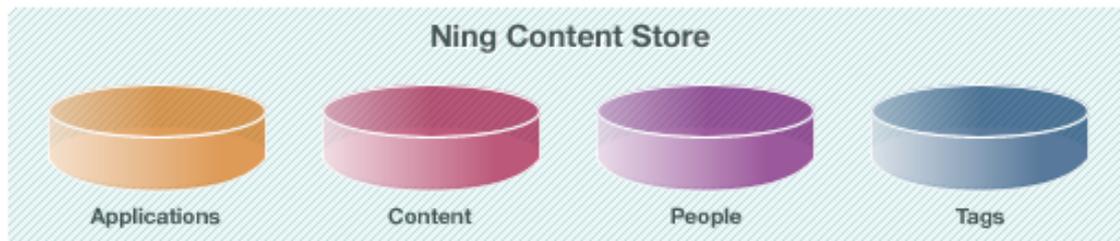


Abb. 5.2: Übersicht über den Ning Content Store (vgl. [120])

Dokumentation Die Dokumentation ist ausführlich und vollständig. Sowohl die PHP und JavaScript API als auch die grundlegende Struktur sind ausführlich erläutert und bieten Beispiele für den Einstieg. Es stehen auch einige Tutorials bereit, wobei viele eher für Anfänger gedacht sind.

Implementierung Da es sich um ein SaaS System handelt, sind bei der Implementierung keinerlei Hardwareanforderungen zu nennen. Zur Skalierbarkeit der Nutzerzahlen macht Ning keine Angaben. Allerdings ist auf Grund der Architektur davon auszugehen, dass auch größere Nutzer-mengen verarbeitet werden können. Die Nutzung der Standardversion ist kostenlos, auch für die

Nutzung der API oder die Möglichkeit Code zu editieren, fallen keine Kosten an. Allerdings berechnet Ning monatliche Gebühren für eine eigene Domain, ein werbefreies Soziales Netzwerk, das Einspeisen von eigener Werbung, mehr Speicherplatz und Traffic. In der kostenlosen Standardversion stehen dem Betreiber 10 GB an Speicherplatz und 100 GB an Traffic zur Verfügung.

Zusammenfassung Ning verfügt über umfassende Funktionalitäten. Besonders die Möglichkeit eigener Erweiterungen und der Zugriff auf den bestehenden Code heben es als SaaS System hervor und machen es gegenüber den Systemen mit einer eigenen Installation zu einer attraktiven Alternative. Alle grundlegenden Funktionalitäten werden von Ning abgedeckt. Bei den implementierten Funktionen gibt es aber noch Entwicklungspotential. Beispielsweise bietet das Tagging nicht alle Möglichkeiten an. Sowohl bei der Suche als auch in Bezug auf die Anzeige und Eingabe von Tags bleiben noch viele Möglichkeiten²⁷ ungenutzt. Zwar bietet die NingID die Möglichkeit, sich bei vielen Netzwerken anzumelden, aber alternative Services wie OpenID stehen nicht zur Verfügung. Mit einer Gästeliste und einer Terminzusage bietet Ning zwar schon einige Möglichkeiten, allerdings wären hier zusätzliche Funktionalitäten wie ein Forum von Vorteil. Enttäuschend ist die Suche; eine Suche nach Kategorien, Gruppen, etc. wäre hilfreich. Zwar existiert die Möglichkeit bestimmte Nutzer oder Inhaltsobjekte manuell hervorzuheben. Aber es wäre gut, wenn auch Nicht-Administratoren bestimmte Objekte hervorheben können²⁸. Für eine Zusammenarbeit zwischen den Nutzern wäre eine Möglichkeit, Dokumente online zu erstellen erstrebenswert. Bis auf kleine Optimierungsmöglichkeiten verbleibt aber auch im Vergleich mit den anderen Frameworks (A.17 auf Seite 123ff) der Eindruck einer soliden und guten Lösung.

5.5.4 Community Server

Community Server ist eine auf .NET basierende Lösung und wird von Telligent vertrieben. Insgesamt sind drei Produkte im Angebot:

- Professional Edition
- Intranet Edition
- Enterprise Edition

Die wesentlichen Unterschiede zwischen den Versionen liegen in der Anzahl an Blogs, Gruppen, etc. und der Möglichkeiten für den SSO. Als Alleinstellungsmerkmal bietet die Intranet Edition eine Integration von Microsoft Sharepoint an. Für den Vergleich wurde die Enterprise Edition gewählt, da diese - bis auf die Sharepointfunktionalität - die weitergehenden Funktionalitäten bietet.

Accountfunktionalitäten

Für die Anmeldung benötigt der Anwender Benutzernamen, Passwort, E-Mail und seine Zeitzone. Alternativ kann auch eine OpenID²⁹ oder das Windows Active Directory verwendet werden. Nach der Anmeldung kann der Nutzer zahlreiche weitere Daten wie Ort, Geschlecht, Biographie, öffentliche Emailadresse und zahlreiche Verlinkungen nach außen wie Blogadresse, interessante RSS Feeds und Homepage angeben. Die angebotenen Felder lassen sich allerdings vom Admin nicht über die grafische Benutzeroberfläche ändern. Um die Reputation eines Nutzers einzuschätzen, bietet der Community Server einige Automatismen. Das System vergibt Punkte für Beiträge, Antworten, Bewertungen, Downloads, Freunde und Favoriten. Zusätzlich kann der Administrator auswählen, welche Gewichtung den einzelnen Punkten zukommen soll.

Das System ist grundsätzlich in vier Komponenten aufgeteilt:

²⁷Hier wären z.B. eine Hilfe bei der Eingabe oder eine Anzeige einer Tagcloud wünschenswert.

²⁸Bisher können Nutzer Objekte nur in ihre eigene Favoritenliste aufnehmen.

²⁹Allerdings konnte diese Funktionalität nicht erfolgreich getestet werden.

- Blogs
- Medien
- Forum
- Gruppen

Blogs Blogs werden zentral vom Administrator erstellt und können bestimmten Nutzern zugeteilt werden, die damit ebenfalls Schreibrechte erhalten. Für den Blogeintrag steht ein Rich-Content-Editor zur Verfügung. In den Eintrag können Bilder und andere Medien eingefügt werden. Zusätzlich können an jeden Blogeintrag Dokumente angehängt werden. Kommentare können verboten bzw. nur nach Genehmigung zugelassen werden. Außerdem kann ein Emailabo der Beiträge sowie eine Bewertung zugelassen werden. Blogs können entweder per RSS oder E-Mail abonniert werden.

Foren Die Foren stellen das Kernstück des Community Server dar. So basiert die Forensoftware bei MySpace auf der Software von Telligent. Beiträge können - genau wie Blogbeiträge - mit Bildern, Dokumenten und anderen Medien angereichert werden. Des Weiteren steht eine Umfragefunktion zur Verfügung, außerdem versucht das System automatisch verwandte Forumseinträge zu ermitteln.

Netzwerkstruktur Die Nutzer eines Systems lassen sich in Gruppen einteilen. Diese können entweder offen oder privat sein. Jede Gruppe kann über Blogs, Diskussionen und Medien verfügen. Eine Auflistung der Mitglieder sowie ein RSS Abo stehen für jede Gruppe zur Verfügung. Die Anordnung der Elemente kann für jede Gruppe individuell festgelegt werden, außerdem kann jede Gruppe ein eigenes Design erhalten. Neben den dynamischen Elementen können für jede Gruppe auch Seiten angelegt werden, welche wiederum mit Medien angereichert werden können. Innerhalb der Gruppe können verschiedene Moderationsoptionen aktiviert werden. So kann beispielsweise der Dateiupload moderiert werden. Gruppen werden in Kategorien eingeordnet, können aber nicht verschachtelt werden. Auch die Möglichkeit, Nutzer in Netzwerken einzuordnen bzw. Funktionalitäten daran anzugliedern fehlt dem Community Server.

Inhalte Neben den großen Inhaltskategorien stehen den Nutzern noch Kommentare und Ankündigungen als Inhaltstypen zur Verfügung. Kommentare bestehen aus einem optionalen Verweis auf eine Webseite und dem eigentlichen Kommentar. Announcements können im eigenen Profil gemacht werden. Sie werden mit einem Rich-Content-Editor erstellt, der auch das Einfügen von Medienobjekten erlaubt.

In die so genannte Mediengalerie lassen sich alle Arten von Dokumenten laden. Dateien können entweder von der Festplatte oder aber direkt von einem anderen Webserver geladen werden. Neben einem Titel und der Beschreibung kann jedes Objekt auch mit Tags versehen werden.

Kollaboration Weder für Administratoren noch für die Nutzer besteht die Möglichkeit, Daten aus anderen Systemen zu importieren. Auch die Möglichkeit zum Zugriff auf gelöschte Dateien oder ältere Versionen besteht nicht. Für den Inhalts-Workflow in den Gruppen stehen einige wenige Funktionalitäten - wie zuvor beschrieben - bereit. Der Community Server verfügt über ein ausgeprägtes Rechtemanagement. So können neben den bestehenden Nutzergruppen beliebige weitere Gruppen definiert werden. Diesen oder einzelnen Nutzern können dann Rechte, wie z.B. für den Upload von Dokumenten zugewiesen werden.

Es gibt keine Möglichkeit Inhalte online zu erstellen. Allerdings können importierte Inhalte mit bis zu fünf Sternen bewertet und mit Kommentaren versehen werden. Zusätzlich können Nutzer und Inhaltsobjekte hervorgehoben werden. Bewertungen werden nur sichtbar, wenn ein Objekt aufgerufen wird.

Suche Die Suche wartet mit den üblichen Funktionalitäten auf. Neben der einfachen Stichwortsuche gibt es die Möglichkeit die Suche auf Tags, Autoren, Erstellungsdatum oder nur Objekte von Kontakten zu beschränken. Die Suchergebnisse werden dann nach Kategorien sortiert angezeigt.

Nachrichten Um mit anderen Nutzern direkt in Kontakt zu treten, können Nachrichten verschickt werden. Nachrichten können mit Medienobjekten angereichert und über einen Rich-Content-Editor gestaltet werden. Dargestellt werden die Nachrichten als Konversationen, allerdings gibt es keine Möglichkeit, diese Nachrichten in Ordnern abzulegen.

Kalender Funktionalitäten, um Termine einzustellen bietet der Community Server nicht an.

Tags Die Benutzung von Tags zieht sich konsequent durch das gesamte System. So können Blogeinträge und Inhalte der Mediengalerie getaggt werden. Über einen Dialog können bereits verwendete Tags ausgewählt werden. Eine intelligente Ergänzung ist aber nicht vorhanden. Zugriff auf die Tags ermöglichen die Tagclouds, sowie die Suche, welche auch Tags durchsucht.

Profil Im eigenen Profil kann der Nutzer keine optischen Modifikationen vornehmen. Die Blöcke - im Gegensatz zur Gruppenansicht - lassen sich vom Nutzer selber nicht verschieben. Allerdings gibt das eigene Profil eine gute Übersicht über die eigenen Aktivitäten. So werden hochgeladene Dateien und eigene Forenbeiträge angezeigt. Ankündigungen und Kommentare können auch per RSS abonniert werden.

Datenfreigabe Bezüglich der Datenfreigaben unterstützt der CommunityServer nur sehr wenige Funktionalitäten. Der Nutzer selber kann nur angeben, ob er in der Mitgliederliste erscheinen möchte und auswählen, ob seine E-Mailadresse öffentlich sein soll. Dementsprechend gibt es auch keine Möglichkeiten zwischen verschiedenen Personengruppen zu unterscheiden. Eine interessante Funktionalität ist der so genannte „sichere RSS Feed“, bei dem für jeden Nutzer eine URI mit Hash erzeugt wird. Daneben besteht in vielen Bereichen die Möglichkeit, die IP-Adresse der Nutzer zu loggen, wenn diese Beiträge erstellen.

Applikationen und APIs Wie in Abbildung 5.3 auf der nächsten Seite dargestellt, besteht das System aus drei Schichten. Auf unterster Ebene verfügt das System über eine Datenbank, welche von der Core Service Schicht angesprochen wird und auch den Zugang zur REST API bietet. RSS Feeds werden wiederum von der Präsentationsschicht angeboten. Um Zugang zur API zu erhalten, wird ein API Key benötigt³⁰.

Über die API kann auf die folgenden Bereiche zugegriffen werden bzw. diese erweitert werden:

- Blogs
- Foren
- Gruppen
- Mediengallerien
- Mitglieder

³⁰<http://api.communityserver.org/introduction/>

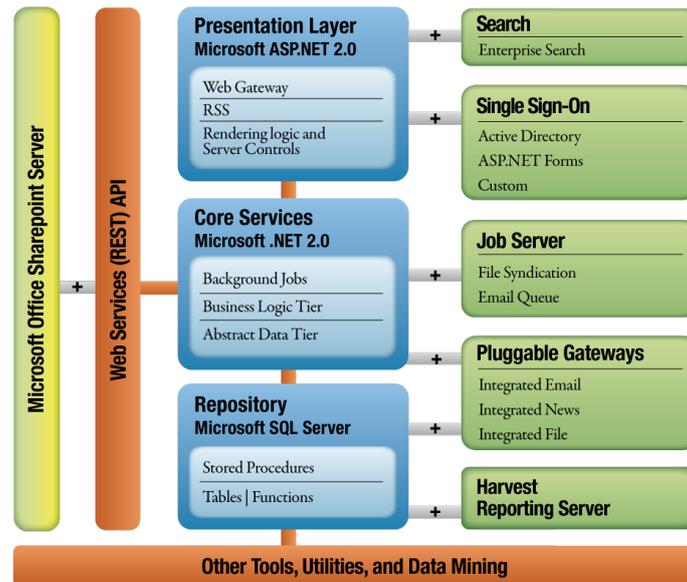


Abb. 5.3: Community Server Architektur (vgl. [149])

- Nachrichten

So können z.B. über die Blog API Beiträge und Blogs aufgelistet, aber auch neu erstellt werden. Auch das Löschen von Beiträgen und das Erstellen von komplett neuen Blogs ist möglich. Zugehörige Funktionalitäten wie das Setzen von Rechten können ebenfalls über die API angesteuert werden. Gleiches gilt für die weiteren Kategorien. Neben der API für die Funktionalitäten steht auch eine Theme Engine mit dem Namen „Chameleon“ für den Entwickler zur Verfügung. Wie zuvor erwähnt, ermöglicht das System zwar nicht die Strukturierung von Nutzern in Netzwerken. Allerdings ist es über die API möglich, Nutzerdatenbanken zwischen verschiedenen Installationen zu teilen und diese dann jeweils für ein System zu aktivieren.

Beschrieben sind diese Funktionen alle in einer ausführlichen und weitestgehend vollständigen Dokumentation³¹. Auch die Funktionalitäten der API sind mit Beispielen versehen.

Implementierung Da es sich um eine auf .NET basierende Software handelt, benötigt der Community Server eine Microsoft Umgebung. Als Server wird ein Microsoft Windows Server 2003 oder 2000 benötigt. Als Datenbanksystem ist ein Microsoft SQL Server notwendig. Für den Betrieb empfiehlt Telligent ein Cluster aus mindestens drei Rechnern, wobei es sich um Dual-Prozessoren mit 2-4 GB RAM handeln sollte. Für den Datenbankserver gelten die selben Anforderungen, allerdings sollte es hier ein 64Bit System sein. Die Lizenzkosten sind von den Nutzerzahlen unabhängig und betragen ~16.000 €/Jahr.

Zusammenfassung Die evaluierten Funktionalitäten des Community Servers überzeugen. Hervorzuheben ist die Möglichkeit, Nutzer in verschiedene Rollen einzuteilen. Auch die automatische Erstellung eines Nutzerrankings ist das umfangreichste unter den verglichenen Frameworks. Es können nicht nur zahlreiche Optionen ausgewählt werden, sondern der Administrator kann zusätzlich die Gewichtung selber einstellen. Insgesamt ist das System jedoch sehr vom Administrator abhängig. Blogs und Gruppen müssen von diesem und können nicht direkt, wie bei den anderen

³¹Bei der Theme Engine waren zwei Unterkapitel noch nicht mit Inhalte gefüllt.

Systemen, von einem Nutzer erstellt werden. Sowohl Tags und RSS Feeds ziehen sich konsequent durch das System und entfalten damit ihren Nutzen. Bei den Tags ist allerdings eine intelligente Ergänzung bei der Eingabe wünschenswert. Das Umsortieren von Inhaltsblöcken ist nur in der Gruppenansicht und nicht z.B. auf den Profilseiten möglich. Wünschenswert wäre auch eine globale Aktivitätsanzeige. Die Möglichkeit, Inhalte zur Bewertung sowohl über eine fünfstufige Skala als auch über Kommentare vorzunehmen, ist positiv hervorzuheben. Allerdings besteht außer in der direkten Dokumentenansicht keine Möglichkeit, auf diese Information zurückzugreifen. Auch alle weiteren Funktionalitäten zeigen keine größeren Schwächen.

Mit seinen starken Referenzkunden und im Vergleich mit den anderen Frameworks (A.17 auf Seite 123ff) scheint der Community Server auch für größere Projekte geeignet.

5.5.5 Lotus Connections

Hauptzielgruppe von Lotus Connections³² sind die Intranets von Unternehmen. Lotus Connections sollen den Unternehmen helfen, ihre Mitarbeiter besser zu vernetzen und dadurch einen Produktivitätsgewinn zu erzielen. Grundlage für die Untersuchung von Lotus Connections war eine öffentliche Testinstallation³³, die unter dem Namen „Greenhouse“ firmiert. Das Produkt Lotus Connections besteht aus fünf Funktionsgruppen³⁴:

1. Profile
2. Communities
3. Blogs
4. Dogear (Lesezeichen)
5. Activities

Accountfunktionalitäten Die Registrierung kann mit sehr wenigen Angaben erfolgen (Vor- und Nachname, Emailadresse und Firma)³⁵. Der Login erfolgt im Regelfall über die Emailadresse und einen eigenständige Account, kann aber auch an ein LDAP System oder an den Single Sign-on von Websphere angebunden werden. Bei den hinterlegbaren Daten liegt der Schwerpunkt auf für Unternehmen typische Elemente, wie dem Gebäude des Mitarbeiters und seinen Kontaktdaten (Telefonnummer, Handynummer, IP-Telefonnummer). Diese Felder können aber vom Betreiber frei definiert werden und es besteht auch die Möglichkeit, Templates für verschiedene Benutzer-typen wie z.B. Vertreter oder Entwickler anzulegen.

Blogs Ein zentrales Element stellt die Blogkomponente da. Jeder Nutzer hat hier die Möglichkeit, ein oder mehrere Blogs zu führen. Er kann außerdem Blogs mit anderen Personen zusammen führen. Blogeinträge werden mit einem Rich-Content-Editor erstellt; zusätzlich können beliebige Dateien als Anhänge hinzugefügt werden. Für Blogeinträge besteht außerdem die Möglichkeit, einen Genehmigungsprozess zu verwenden. Für die Darstellung bietet die Komponente die Möglichkeit an, Blogeinträge manuell hervorzuheben. Zusätzlich geben automatische Sichten sortiert nach „aktiven Tags“, „empfohlene Einträge“, „am meisten kommentierte“ und „am meisten besuchte“ einen Überblick über Aktivitäten. Blogeinträge sowie die Kommentare können per RSS/Atom abonniert werden.

³²<http://www-01.ibm.com/software/lotus/products/connections/>

³³<https://greenhouse.lotus.com/homepage/web/getuserpref>

³⁴Jedes dieser Produkte kann aber auch alleine eingesetzt werden.

³⁵In der Beispielininstallation ist noch eine Freischaltung seitens des Administrators notwendig.

Datenfreigabe Keine Funktionalitäten sind im Bereich der Datenfreigabe zu finden. So kann der Nutzer den Zugriff auf seine Daten weder für bestimmte Felder noch auf Personengruppen beschränken. Dies hat sicherlich auch mit der Herkunft aus dem Unternehmensumfeld zu tun, da hier generell von einer sehr geschlossenen Benutzergruppe ausgegangen wird. Allerdings wäre eine Beschränkung auf bestimmte Personengruppen sicherlich sehr sinnvoll, insbesondere unter Beachtung der schon vorhandenen Funktionalitäten, um sich mit anderen Personen zu verbinden und die Hierarchie in einem Unternehmen abzubilden.

Netzwerkstruktur Für die Gruppierung von Personen bietet Lotus Connections drei Elemente an. Zum einen können über LDAP die Hierarchien in einem Unternehmen abgebildet werden. Neben dieser „automatischen“ Zuordnung, kann man sich auch mit einer anderen Person manuell als Kollege verbinden. Die dritte Möglichkeit sind Communities, die neben der Zuordnung von Nutzern auch noch über weitere Funktionalitäten verfügen. So können die Mitglieder einer Community Benachrichtigungen über Aktivitäten (Community Feeds), Lesezeichen und ein Forum gemeinsam nutzen. Auch die Kommunikation über einen Emailverteiler ist möglich. Communities können geschlossen, offen oder moderiert sein. Für die Kollaboration bietet Lotus Connections die Einbindung von Confluence³⁶ oder Socialtext Wiki³⁷ an. Vom Administrator können über die API auch noch weitere communityspezifische Applikationen hinzugefügt werden.

Suche Die Suche erfasst Titel, Beschreibung und Inhalt von allen Dokumenttypen. Standardmäßig erfolgt die Suche per Stichwort in dem jeweiligen Bereich, wobei bereichsspezifische Vorsortierungsmöglichkeiten zur Verfügung stehen, beispielsweise die Beschränkung auf eine Community oder die Filterung nach den Inhalten eines bestimmten Profelfelds. Das Suchfeld auf der Startseite bietet wiederum die Möglichkeit, alle fünf Bereiche zu durchsuchen. Weitere Differenzierungen bei der Suche sind nicht möglich.

Nachrichten Eine direkte Kontaktmöglichkeit über ein systemeigenes Messaging System bietet Lotus Connections nicht an.

Kalender Eine Terminverwaltung im klassischen Sinne bietet Lotus Connections nicht an. Allerdings deckt die Funktionalität Activities viele Bereiche der klassischen Terminplanung ab. Der Nutzer kann als finales Ziel eine Activity anlegen. Innerhalb einer Activity können dann wiederum Nachrichten veröffentlicht, Dateien hoch geladen und Links gespeichert werden. Als Unterteilung für eine umfangreiche Activity besteht die Möglichkeit ToDos anzulegen. Außerdem ist eine Unterteilung in Sektionen möglich, um einen besseren Überblick zu gewährleisten. Zu einer Activity können mehrere Personen hinzugefügt werden, so dass eine gemeinschaftliche Bearbeitung möglich ist. Activities verfügen über ein Rechtesystem (Besitzer, Autor und Leser) und können sowohl für eine bestimmte Personengruppe als auch für die Öffentlichkeit zugänglich gemacht werden. Für die Erstellung von neuen Activities kann auf bereits bestehende Activities als Template zurückgegriffen werden.

Tags Zentral und konsequent zieht sich die Taggingfunktionalität durch das System. Sie ersetzt in den meisten Fällen hierarchische Ordnungsstrukturen. Personen können sowohl sich selber als auch andere Personen taggen. Lesezeichen, Blogeinträge und Activities können ebenfalls getaggt werden. Auf allen Seiten steht eine Ansicht der Tags entweder als Wolke oder als List zur Verfügung, wobei die Masse der angezeigten Tags vom Nutzer individuell bestimmt werden kann. Bei der Tagsuche erleichtert eine Autovervollständigung die Eingabe, diese fehlt allerdings unverständlicherweise bei der Eingabe von eigenen Tags bei der Erstellung von Inhalten.

³⁶<http://www.atlassian.com/software/confluence/>

³⁷<http://www.socialtext.com/products/>

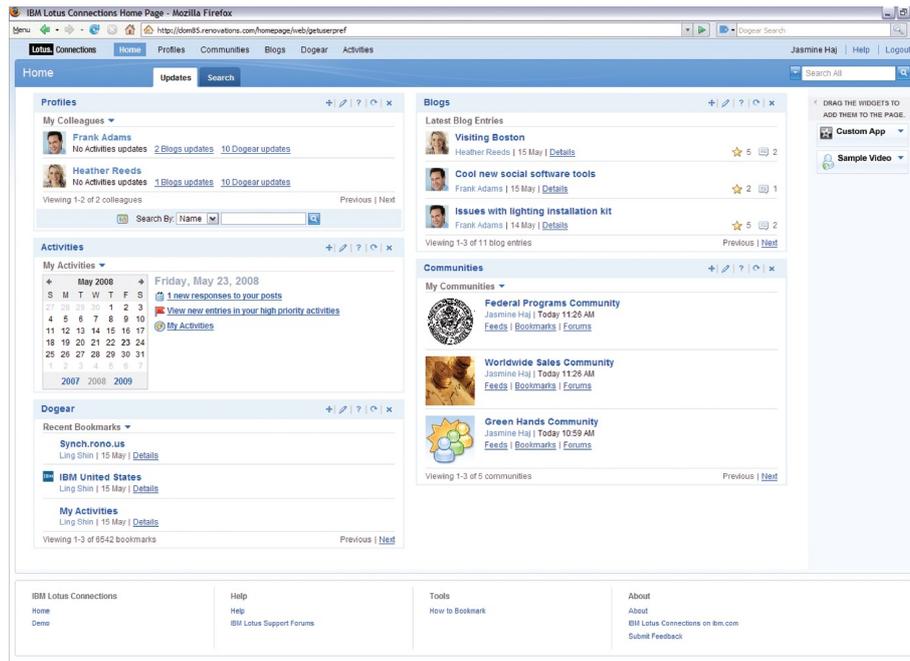


Abb. 5.4: Beispielhomepage für einen Benutzer in Lotus Connections (vgl. [77])

Profil Zentraler Ausgangspunkt für den Nutzer ist seine „Homepage“, wie in Abbildung 5.4 dargestellt. Hier erhält der Nutzer einen Überblick über Aktivitäten in den fünf Funktionsbereichen. Die Anzeigen sind als Widgets realisiert und können vom Nutzer beliebig angeordnet werden. Weitere Widgets können, wenn vom Administrator zur Verfügung gestellt, hinzugefügt werden. Beim Aufruf eines Profils kann der Nutzer neben den persönlichen Informationen über die Person auch seine erstellten Beiträge in den fünf Funktionsbereichen sehen und eine „vcf“-Karte herunterladen.

Lesezeichen Die Lesezeichenkomponente mit dem Namen „Dogear“ bietet dem Nutzer die Möglichkeit, zentral seine Lesezeichen abzulegen. Lesezeichen werden dabei nicht hierarchisch, sondern mit Tags versehen abgelegt. Beim Zugriff stehen insbesondere personenbezogene Sichten im Vordergrund. Neben den eigenen Lesezeichen stehen auch Ansichten für die Lesezeichen aller Kontakte oder bestimmter Personen zur Verfügung.³⁸ Eine Importfunktionalität für lokale Lesezeichen besteht ebenfalls.

Applikationen und APIs Abbildung 5.5 auf der nächsten Seite zeigt die grundlegende Struktur von Lotus Connections. Wie bereits erwähnt ist das Gesamtprodukt in die fünf Funktionsbereiche und die eigene Übersichtshomepage gegliedert. In einer zweiten Schicht werden LDAP, Datenbank und Datenspeichersysteme angebunden. Die Lotus Connection Services sind über eine REST API (vgl. [78]) angebunden und ermöglichen somit auch eine Verwendung der Services in anderen Systemen. Die segmentierte Struktur mit Servicecharakter ermöglicht außerdem die Einbindung einzelner Komponenten auf anderen Webseiten als JavaScript Snippet. Eine Anwendungsmöglichkeit ist beispielsweise ein Lesezeichenbutton. Eine Anpassung des Designs kann über CSS Style Dateien erfolgen.

³⁸Dies liegt der Annahme zu Grunde, dass bekannte Personen in der Regel ähnliche Webressourcen verwenden könnten.

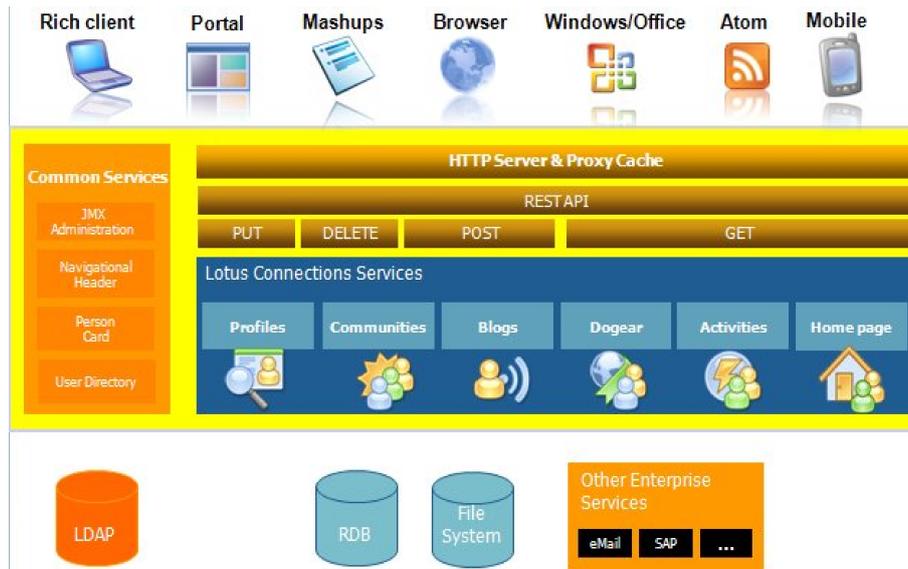


Abb. 5.5: Struktur von Lotus Connections (vgl. [79])

Implementierung Für eine Installation von Lotus Connections sind mindestens zwei Server (Intel 64 oder IA-32) mit jeweils zwei 2.6 GHz CPUs und 4 GB RAM notwendig. Als Betriebssysteme werden Red Hat und Suse Linux, Microsoft Windows Server 2003 und AIX unterstützt. Als Datenbankserver können IBM DB2, Oracle oder Microsoft SQL Server 2000 verwendet werden (vgl. [76]).

Dokumentation Die Dokumentation sowohl für die API als auch für die Installation und Einrichtung ist vollständig und gut sortiert. Insbesondere das „IBM Lotus Connections 2 Information Center“ (vgl. [78]) stellt die zentrale Anlaufstelle für Betreiber einer Lotus Connections Installation dar.

Zusammenfassung Lotus Connections erfüllt die Erwartungen an Funktionalitäten eines Sozialen Netzwerks. Auffällig ist die hervorgehobene Untergliederung in die fünf Funktionsbereiche, die so explizit bei keinem der anderen Produkte gemacht wird, sowie die Fokussierung auf den Einsatz im Intranet eines Unternehmens. Jeder Bereich enthält die grundlegenden Funktionen, verzichtet allerdings auf Funktionen, die beispielsweise von anderen Blogsystemen bekannt sind. Besonders überzeugend präsentiert sich die Taggingkomponente, die sich konsequent durch alle Funktionsbereiche zieht und als Ordnungskriterien in den meisten Bereichen eine hierarchische Sortierung ablöst. Ein konsistentes Bedienungskonzept erleichtert die Verwendung der verschiedenen Bereiche für den Nutzer. Standardmäßig bietet Lotus Connections keine Messagingfunktion an. Als besonderes Funktionsmerkmal bietet das System eine Lesezeichenverwaltung, die einen herausgehobenen Platz einnimmt. Die Implementierung dieser macht einen soliden Eindruck und die Möglichkeiten, Lesezeichen als privat zu markieren stellen eine sinnvolle Zusatzfunktionalität dar. Die Untergliederung in die fünf Produkte lässt Lotus Connections nicht immer als einheitliches Produkt erscheinen, bewirkt aber gleichzeitig einen sehr aufgeräumten und strukturierten Aufbau der Benutzeroberfläche. Zwar ist Lotus Connections auf homogene Benutzergruppen angelegt, dennoch wäre es sinnvoll, dass gewisse Informationen vom Nutzer nur bestimmten Personengruppen zugänglich gemacht werden können. Insbesondere die Möglichkeit der zweigliedrigen Struktur - über vergebene Hierarchieebenen und manuelle Zuordnung von Kollegen - könnte sich das System hier zu Nutzen machen. Einen alternativen Weg der Terminplanung schlägt Lotus Connections mit der Fokussierung auf Activities ein. Insbesondere die vielfältigen Möglichkeiten, weitere Inhalte zu einer Activity hinzuzufügen und zusammen mit anderen Personen zu bearbeiten bietet Chancen für eine produktive Arbeit. Auf den ersten Blick

sehr schlicht gestaltet, hat die Suche durch die kontextspezifischen Eingrenzungen einen hohen Nutzwert, ohne dass der Nutzer viele Einstellungen vornehmen muss.

Zusammenfassend notiert, macht Lotus Notes einen sehr soliden und ausgereiften Eindruck. Die Wahl der Funktionsbereiche ist in dieser Deutlichkeit ein Alleinstellungsmerkmal, welches sich wesentlich - in der Struktur und dem Angebot der Funktionen - von anderen Netzwerken unterscheidet (Tabelle A.17 auf Seite 123ff).

5.6 FAZIT DES FRAMEWORKSVERGLEICHS

Die Analyse hat gezeigt, dass die Frameworks trotz der Heterogenität des Untersuchungsfeldes sehr gut in Relation zueinander gesetzt werden können. Bis auf wenige Ausnahmen wie beispielsweise ein Nachrichtensystem oder die Installation waren keine signifikanten Unterschiede bei den Funktionalitäten zu beobachten, welche einen Vergleich erschwert hätten. Auch wenn die Gruppe der SaaS Frameworks nur durch Ning vertreten war, so zeigte Ning doch exemplarisch sehr gut, welche Grenzen aber auch welche Möglichkeiten dieses Konzept beinhaltet. Die adaptierte Evaluierungsmatrix hat sich als passend erwiesen. Einzig die Lesezeichenkomponente von Lotus Connections fiel aus dem Kriterienkatalog heraus.

Alle untersuchten Frameworks ermöglichen es dem Nutzer, ein Soziales Netzwerk für eine geschlossene Benutzergruppen anzulegen. Welche Funktionalitäten dabei implementiert wurden, ist sehr unterschiedlich. So gehört die Funktionalität der Verbindung mit anderen Nutzern, welche aus den öffentlichen Netzwerken bekannt ist, bei den Frameworks nicht immer zu den Standardfunktionalität. Allgemein lässt sich sagen, dass bei den Frameworks eher wenige Funktionalitäten als bei den öffentlichen Sozialen Netzwerken implementiert wurden, dafür aber konsequenter.

Bei den Accountfunktionalitäten hebt sich der CommunityServer ab, indem er eine Anbindung an den offenen Standard OpenID erlaubt. Als Standard bei den Profilen muss sicherlich gelten, dass der Administrator Felder frei definieren kann. Dies ist zwar bei allen Frameworks der Fall, allerdings nicht immer über die Benutzeroberfläche möglich. Die Möglichkeit, eine Reputation von Benutzern abzubilden, wird nur sehr wenig genutzt. Lediglich Clearspace und der Communityserver bieten hier automatisierte Funktionalitäten an. Eine Bloggingfunktionalität steht bei allen Frameworks zur Verfügung, wobei die Blogs von Lotus Connections die meisten Möglichkeiten bieten und auch jedem Nutzer zur Verfügung stehen. Ein Abo per RSS Feed gehört ebenfalls zum Standard.

Beim Schutz der eigenen Daten sind große Unterschiede zu verzeichnen. So bietet Lotus Connections gar keine Möglichkeiten und Clearspace Community, Ning und der Community Server lassen die Einstellungen nur als Vorgabe vom Administrator zu. Nur PeopleAggregator ermöglicht es dem Nutzer, die Privacyeinstellungen selber vorzunehmen. Die Möglichkeit, Schutzseinstellungen von Personengruppen abhängig zu machen, nutzen nur Ning und PeopleAggregator. Eine Einteilung der Benutzer muss als Standardfunktion erachtet werden und wird in der ein oder anderen Form auch von allen Frameworks angeboten. Beim Grad der Strukturierung bietet Clearspace die meisten Möglichkeiten, da hier eine beliebige Verschachtelung möglich ist. Ebenfalls üblich ist es, dass die Unterteilungen Funktionalitäten wie Foren und Fotos integriert, die dann wahlweise einem exklusiven Nutzerkreis zur Verfügung stehen. Nur Lotus Connections bietet die Funktionalität einer eventzentrierten Gruppe.

Bei der Suche schneiden PeopleAggregator und Ning schlecht ab, da sie keine Möglichkeiten der Eingrenzung bieten. Clearspace und Lotus Connections gefielen auf Grund der intelligenten Eingrenzung am besten. Beim Import von Dateien oder Daten sind alle Systeme sehr zurückhaltend; es sind fast keine Funktionalitäten verfügbar. Bei den Workflows ist die Funktionalität entweder nicht vorhanden oder auf bestimmte wenige Inhaltstypen begrenzt. Blogbeiträge, Dokumente jeder Art, Kommentare, Forenbeiträge und Audio bilden den Kern der verfügbaren Daten. Je nach

Netzwerk ist oft ein bestimmter Inhaltstyp nicht vorhanden. Lotus Connections fehlt beispielsweise eine Fotofunktion, dafür enthält es aber eine Activitiesfunktionalität. Über ein ausgeprägtes Gruppenrollensystem verfügen nur der Community Server und Clearspace Community.

Bis auf Lotus Connections bieten alle Frameworks eine mehr oder weniger gut ausgebaute interne Nachrichtenfunktion an. Eine ausgeprägte Kalenderfunktion im klassischen Sinne gibt es nur bei Ning. Lotus Connections und PeopleAggregator decken dies mit Activities bzw. Events ab. Alle untersuchten Frameworks bieten eine Taggingfunktion an. Allerdings ist diese unterschiedlich gut implementiert. Bei PeopleAggregator kann der Nutzer zwar Tags eingeben, diese werden jedoch kaum verwendet, um Inhalte leichter auffindbar zu machen. Clearspace und Lotus Connections implementieren die Taggingfunktion am konsequentesten und erzielen so einen echten Mehrwert. Bei der Kollaboration bieten nur Clearspace und Lotus Connections Funktionalitäten für Dokumente bzw. Blogs an. Eine Bewertungsmöglichkeit für ein oder mehrere Inhalte existiert bei allen Frameworks. Ning und Lotus Connections bieten die Möglichkeit, die eigene Übersichtseite per Drag&Drop zu modifizieren und somit einen individuellen vom Nutzer gewünschten Seitenaufbau zu schaffen. Einen unterschiedlichen Ansatz verfolgen die Frameworks bei der Darstellung von Aktivitäten. Während Clearspace Community einen dokumentenzentrierten Ansatz wählt, herrscht sonst eher ein personenzentrierter Ansatz vor, der dem Nutzer mitteilt, welche Person eine bestimmte Aktivität vorgenommen hat.

Für die Implementierung und den damit verbundenen Kosten lässt sich pauschal keine Aussage treffen, da hier die Anforderungen des Betreibers ausschlaggebend sind. Bei den APIs gibt es zwei grundlegende Herangehensweisen. Auf der einen Seite stehen REST/SOAP APIs, die eine Manipulation mit vielen Programmen erlauben. Dies bieten in einem mehr oder weniger ausgebauten Umfang alle Frameworks an. Eine API, um eigene Plugins zu schreiben, gibt es nur bei Clearspace Community und Ning. Bei PeopleAggregator besteht die Möglichkeit, den Quellcode direkt zu editieren.

In der Standardkonfiguration bieten die Frameworks dem Nutzer nur sehr wenige Möglichkeiten für einen Datenimport an. Einen funktionierenden Import von Emailadressen bietet nur Ning an. Möglichkeiten für den Nutzer, direkt seine Daten in einem standardisierten Format zu exportieren, bietet hingegen keines der Frameworks an. Unter diesen Bedingungen gewinnt die API an Bedeutung, da hierüber dem Nutzer ermöglicht werden kann, Daten zu exportieren. Clearspace und Ning bieten hier die ausgefeiltesten Möglichkeiten, eine Applikation direkt in das Framework zu integrieren und dem Nutzer eine Oberfläche für die Datenaggregation zu bieten, sowohl von den Zugriffsmöglichkeiten auf die Daten als auch durch die Möglichkeit der direkten Integration. Insgesamt lässt sich sagen, dass die Voraussetzungen für eine Datenaggregation durchaus vorhanden, aber in der Standardinstallation nicht oder nur unzureichend implementiert sind.

Zusammenfassend ist festzustellen, dass Clearspace Community, Ning, Community Server und Lotus Connections für den produktiven Einsatz empfohlen werden können, PeopleAggregator hingegen nicht. Während die ersten vier Lösungen durchweg einen soliden und ausgereiften Eindruck machen, verfolgt PeopleAggregator zwar einige interessante Ansätze, aber durch fehlerhafte Funktionalitäten und die Nichtverfügbarkeit einer vollständigen Dokumentation kann es nicht für den Einsatz empfohlen werden.

Tabelle 5.1 auf der nächsten Seite bietet einen Überblick über die Gesamtbewertung der Frameworks. Die Accountfunktionalitäten und die Implementierung wurden dabei nicht numerisch bewertet, da in diesen Bereichen keine pauschale Bewertung möglich ist. Zu beachten ist, dass die Summe der Bewertungen für den konkreten Anwendungsfall nur eine bedingte Aussagekraft hat. So erzielt beispielsweise Lotus Connections eine sehr hohe Gesamtpunktzahl, weist aber andererseits in einigen Bereichen eine sehr niedrige Punktzahl auf. Clearspace Community weist hingegen eine sehr viel geringere Varianz auf. Für die Verwendung in einem konkreten Anwendungsfall empfiehlt es sich, zusätzlich eine Gewichtung der einzelnen Funktionsbereiche vorzunehmen.

	Durschnitt	People Aggregator	Clearspace Community	Ning	Community Server	Lotus Connections
Account-funktionalitäten	-	-	-	-	-	-
APIs	3	3	4	4	4	2
Blog	4	3	4	5	5	5
Privacy	3	3	3	4	2	1
Gruppen	2	3	3	2	1	3
Suche	3	2	4	2	4	5
Inhalte	2	1	4	2	2	3
Nachrichten	2	2	2	2	4	0
Kalender	1	1	0	4	0	1
Tagging	4	3	4	3	3	5
Implementierung	-	-	-	-	-	-
Kollaboration	3	2	4	2	2	5
Profil	3	2	2	5	2	4
Summe		26	34	35	29	34
Rang		(4.)	(2.)	(1.)	(3.)	(2.)
Variationskoeffizient		0,345	0,420	0,392	0,569	0,604

Tab. 5.1: Evaluierungsmatrix der Frameworks

6 UNTERSUCHUNG UND BEWERTUNG DER DATENAGGREGATION

In den vorangegangenen Kapiteln wurde die Grundlage für eine fundierte Untersuchung und Bewertung der Datenaggregation gelegt. Kapitel 4 auf Seite 30 zeigt auf, welche Möglichkeiten es für den zu realisierenden Prototypen gibt, auf die Daten der Nutzer zuzugreifen. Um ein geeignetes Framework für die Implementierung zu finden, wurden fünf Frameworks im Kapitel 5 auf Seite 42ff analysiert. Auch hier stand speziell die Möglichkeit, eigene Plugins mit Zugriff auf die Nutzerdaten zu implementieren, im Vordergrund.

Im folgenden Kapitel wird zuerst der aktuelle Stand der Datenaggregation analysiert. Aufbauend darauf wird eine Matrix für die Bewertung der Problemkategorien bei der Datenaggregation erstellt. Im nächsten Schritt werden die Funktionalitäten konzipiert und die Implementierung beschrieben. Das Kapitel schließt mit einer Auswertung der Implementierung bezogen auf die zuvor definierten Problembereiche und gibt allgemein Empfehlungen für Datenaggregationen im Umfeld Sozialer Netzwerke.

Da im Rahmen dieser Arbeit nur eine begrenzte Anzahl an Netzwerken untersucht werden konnte, erheben die auf Basis dieser Datenlage getroffenen Aussagen keinen Anspruch auf Allgemeingültigkeit. Allerdings decken, wie die Auswahl der Netzwerke und Frameworks im Abschnitt 2.5 auf Seite 15 und 5.3 auf Seite 43 aufzeigt, die untersuchten Netzwerke die wichtigen Funktionalitäten und Eigenschaften für eine Analyse ab.

6.1 AKTUELLER STAND DER DATENAGGREGATION NACH DURCHFÜHRUNG DER ANALYSE

Soziale Netzwerke leben von den Daten der Nutzer. Dies trifft sowohl für die Nutzer als auch für die Betreiber zu. Für Erstere bietet sich die Möglichkeit mehr über seine Kontakte zu erfahren, die Betreiber hingegen gewinnen aus den Daten einen Mehrwert hinsichtlich zielgerichteter Werbung.

Die Möglichkeiten für den Nutzer, Daten manuell einzugeben, sind zahlreich. Dies zeigt sich insbesondere bei den Funktionalitäten. So dienen die meisten der in Kapitel 3.2 auf Seite 20 analysierten Funktionen hauptsächlich dazu Daten einzugeben. Dies ist natürlich auch im Interesse des Nutzers. Allerdings hat die Analyse gezeigt, dass es fast keine Möglichkeiten gibt, Daten zu importieren. Ohne zusätzliche Plugins fehlt auch die Möglichkeit, Fotos aus Desktopprogrammen zu importieren. Auch bei den Profildaten kann man nicht bereits bestehenden Daten beispielsweise aus einer vCard oder einem anderen Netzwerk importieren. Dies gilt auch für die Frameworks. Eine Ausnahme bildet hier „ClearSpace Community“, wo es die Möglichkeit für einen Import von Blogbeiträgen gibt. Auch eine Aggregationsmöglichkeit für die großen Foto- und Blogdienste existiert nicht.

Etwas besser stellt sich die Situation beim Import von Kontaktdaten dar. Hier besteht die Möglichkeit, Emailadressen aus anderen Datenquellen zu importieren und im System überprüfen zu lassen, ob diese Emailadressen schon bekannt sind. Wie im Abschnitt 3.2.2 auf Seite 21 zu sehen bestehen hier große Unterschiede bei der Anzahl von möglichen Quellsystemen für den Import. Dies ist darauf zurückzuführen, dass Soziale Netzwerke für den Nutzer nur eine Anwendung haben, wenn er auch Kontakte hat. Technisch ist der Import auf Grund der vielen unterschiedlichen Emailadressen als aufwendiger als ein generischer Datenimport einzuschätzen. Dass es hinsichtlich des Importes dennoch hier mehr Möglichkeiten für die Nutzer gibt, lässt sich nur damit erklären, dass das Anlegen von Kontaktdaten für den Betreiber von größerer Bedeutung ist als der Import von Daten.

Sind die Möglichkeiten hinsichtlich des Imports von Daten schon sehr gering, stellt sich diese Situation beim Export von Daten noch schlechter dar. Aus technischer Sicht ist dies völlig unverständlich, da hier, wie später gezeigt wird, sehr viel weniger Hürden bei der Implementierung zu

überwinden sind. Keines der untersuchten bestehenden Netzwerke bietet die Möglichkeit eines Exports von Daten. Auch rechtliche Schwierigkeiten dürften keine existieren. Der Export von Informationen, die zu einem Kontakt des Anwenders gehören ist zwar rechtlich geschützt, aber ein Export der eigenen Daten ist ohne Probleme möglich. Mit Initiativen wie „Facebook Connect“ und „Data Availability“, die im Kapitel 4 auf Seite 30 untersucht wurden, täuschen Facebook und MySpace schlussendlich nur eine Offenheit vor, die jedoch nicht erreicht wird. Es ist lediglich vorgesehen, dass man seine eigenen Datenbestände aus dem jeweiligen Netzwerk auch auf anderen Seiten verwenden kann. Diesen Initiativen liegt keine neue Offenheit zu Grunde, sondern das Interesse zum zentralen Datenspeicherplatz zu werden und auf möglichst vielen anderen Internetseiten präsent zu sein. Einzig prominente Ausnahme ist hier die API des Yahoo Adressbuches, wo auch eine Synchronisierungsfunktion zur Verfügung steht. Allerdings ist dies auch wiederum auf die Kontakte begrenzt und steht nicht für die eigenen Daten zur Verfügung.

Die aktuelle Situation bei den Netzwerken in Bezug auf eine Datenaggregation legt die Vermutung nahe, dass es technisch nicht oder nur sehr schwer möglich ist diese durchzuführen. Allerdings zeigt der Vergleich der Netzwerke und Frameworks auch, dass bei den Frameworks zwar immer noch geringe, aber schon erweiterte Möglichkeiten für die Datenaggregation vorhanden sind. Dies liefert erste Argumente dafür, dass es die wirtschaftlichen Überlegungen der Anbieter sind und nicht die technische Machbarkeit, die einer Datenaggregation im Weg stehen. Ein weiteres Indiz für die Machbarkeit sind die zahlreichen Technologien, die es in dem Umfeld von Sozialen Netzwerken gibt und welche auch den Betreibern bereits verwendet werden. So ist der Datenzugriff auf andere Netzwerke über vorhandene REST APIs möglich. Ein strukturierter Export könnte beispielsweise über das FOAF realisiert werden. Am Beispiel des Imports und Exports soll der Abschnitt 6.6 auf Seite 75ff zeigen, dass eine funktionale Implementierung möglich ist.

6.2 AKTUELLER STAND DER VISUALISIERUNG VON DATEN NACH DURCHFÜHRUNG DER ANALYSE

Bis auf wenige Ausnahmen¹ versuchen alle Betreiber von Sozialen Netzwerken das von Google mit AdSense eingeführte Konzept der kontextspezifischen Werbung anzuwenden. Im Unterschied zur Suche bei Google sind in den Sozialen Netzwerken viel weitergehende Informationen über den Nutzer verfügbar. Inwieweit sich dieser Nutzen auch wirklich finanziell niederschlägt, ist bisher nicht bekannt. Unabhängig von den Gewinnen durch kontextspezifische Werbung haben sich die Betreiber hier einen alternativen Zugriff auf die Daten gesichert, der den Nutzern in diesen Formen bisher verschlossen bleibt. Zwar ermöglichen die Suchen seitens der Nutzer der Sozialen Netzwerke oft auch den Zugriff auf die Interessen anderer, aber abseits von der Suche über Stichworte gibt es fast keine Zugriffsmöglichkeiten. Als Ausnahme zeigt Xing eine Karte an, auf der alle Kontakte verzeichnet sind. Allerdings betrifft dies wiederum nur die bereits bekannten Kontakte und unterstreicht wiederum den nutzerzentrierten Ansatz. Viele Navigationselemente sind darauf ausgelegt, dass ein Nutzer einen anderen Nutzer mit Namen oder Emailadresse sucht. Möglichkeiten, neue Nutzer anhand ihrer Daten zu entdecken, gibt es nur sehr wenig. Es wäre jedoch wünschenswert, dass die vorliegenden Daten auch für den Nutzer auf alternativen Wegen zugänglich gemacht würden.

Allerdings stehen bisher fast keine Visualisierungen der Nutzerdaten zur Verfügung. Außer Tagclouds, die sich dann aber auf Beiträge der Nutzer und nicht diese selber beziehen, gibt es keine grafische Repräsentation. Am Beispiel einer Kombination aus Tagclouds und Kartendarstellung, soll im Abschnitt 6.5.4 auf Seite 74 gezeigt werden, dass alternative Visualisierungsmöglichkeiten realisierbar sind. Gleichzeitig soll auf diese Art der Erfolg des Datenimports überprüft werden.

¹Bei Xing erfolgt die Generierung von Umsätzen aus einer Kombination aus Werbung und monatlichen Gebühren für Premiumnutzer.

6.3 PROBLEMKATEGORIEN EINER DATENAGGREGATION

Für eine Bewertung der Implementierung der oben dargestellten Problematik wurden verschiedene Problembereiche identifiziert:

- Authentifizierung
- Datenimport
- Datenexport
- Zusammenführung der Daten
- Repräsentation der Daten

In Anlehnung an die Bewertung in Kapitel 3 auf Seite 20 werden die Problembereiche auf einer Skala von eins (schwer zu implementieren) bis fünf (leicht zu implementieren) sortiert. Um die unterschiedlichen Aspekte besser zu erfassen, erfolgt die Bewertung unterteilt auf die Bereiche:

- Technisch
- Rechtlich
- Nutzer

6.3.1 Definition der Problembereiche und Bewertungsaspekte

Um eine Anwendung der Kriterien unabhängig vom konkreten Anwendungsfeld zu ermöglichen, erfolgt eine allgemeingültige Definition der Problembereiche und Bewertungsaspekte.

Authentifizierung Dieser Funktionsbereich behandelt den Aspekt, inwiefern es möglich ist, sich bei einem anderen Sozialen Netzwerke einzuloggen, um dort auf die Daten des Nutzer zuzugreifen. Ein solcher Login könnte beispielsweise über OpenID oder aber eine API erfolgen.

Datenimport Hierbei wurde untersucht, welche Schwierigkeiten es bei dem Import von Daten gibt. Vorrangig war hier von Interesse, wie Daten von der API in das System geschrieben werden können.

Datenexport Hierbei wurden die Restriktionen untersucht, die bestehen, wenn der Nutzer seine Daten und Daten über seine Kontakte exportieren möchte.

Zusammenführung der Daten Ist ein Nutzer bei verschiedenen Sozialen Netzwerken registriert, besteht die Notwendigkeit, dass er seine Daten zusammenführen möchte, um einen konsistenten Datensatz zu erstellen. Hinzu kommt, dass bestimmte Daten eventuell nicht übernommen werden sollen. Welche Schwierigkeiten hierbei auftreten können, zeigt dieser Bereich auf.

Repräsentation der Daten Für den Austausch der Daten zwischen den verschiedenen Netzwerken braucht es als Grundlage ein Schema oder eine andere Art der Repräsentation. Dieser Problembereich untersucht, welche Schwierigkeiten hierbei auftreten können.

Technisch Unter diesem Bewertungspunkt wurden Punkte wie die Verfügbarkeit von Technologien und die Möglichkeiten eines Einsatzes dieser betrachtet.

Rechtlich Auch wenn technisch etwas möglich ist, stehen der Implementierungen oft rechtliche Hindernisse im Weg. Eine Bewertung inwiefern dies eine Rolle spielt, wurde unter diesem Punkt vorgenommen.

Nutzer Unter diesem Punkt wurde die Implementierung auf ihre Nutzerfreundlichkeit untersucht.

6.4 ANFORDERUNGSANALYSE

Grundlage für den zu entwickelnden Prototypen ist das folgende Szenario:

Die Firma „Music Technic Inc.“ möchte für Ihre Kunden ein eigenes Soziales Netzwerke implementieren. Die Plattform soll es den Kunden ermöglichen, sich vor dem Kauf über Produkte zu informieren und Meinungen anderer Kunden einzuholen. Zusätzlich soll auch die Möglichkeit bestehen, andere Nutzer zu suchen, die ähnliche Produkte gekauft haben und beispielsweise eigene Erweiterungen für eine Software geschrieben haben oder Musikhardware auf eine spezielle Art zu Hause verwendet haben. Um dem Nutzer den Start zu erleichtern und mehr Nutzer auf die Plattform aufmerksam zu machen und sie daran zu binden, möchte Music Technic Inc. den Nutzern den Import von Kontakten und Daten von anderen Plattformen ermöglichen. Um aber nicht als Datensammler angesehen zu werden, soll das Soziale Netzwerk auch die Möglichkeit eines Daten- und Kontaktexports bieten.

Um die Funktionalitäten für das vorliegende Szenario zu erfüllen, soll das Plugin zur Datenaggregation und Visualisierung drei Kernfunktionalitäten unterstützen:

- Im- und Export von Kontakten eines Nutzers aus anderen Netzwerken
- Im- und Export von Daten eines Nutzers aus anderen Netzwerken
- Visualisierung von Nutzerdaten in Form einer Kartendarstellung

6.5 KONZEPTION DER FUNKTIONALITÄTEN

Im folgenden Abschnitt werden die Funktionalitäten beschrieben, die dem Nutzer bei der Verwendung des Prototypen zur Verfügung stehen.

6.5.1 Import

Zu Beginn des Importprozesses erhält der Nutzer eine Übersicht über die Daten, die zum Import bereitstehen. Zugleich werden ihm die im Netzwerk vorhandenen Felder angezeigt. Nun kann er über die Benutzungsoberfläche die verschiedenen Felder zuordnen und danach den Import erfolgen lassen. Um zukünftige Importe einfacher zu gestalten besteht die Möglichkeit, das Schema abzuspeichern.

Beim Import der Kontakte wird dem Nutzer angezeigt, welche Kontakte bereits in dem System vorliegen und die Möglichkeit gegeben mit ihnen Kontakt aufzunehmen.

6.5.2 Export

Der Nutzer hat hier die Möglichkeit, seine Kontakte und seine Daten zu exportieren, wobei er wählen kann, welche Daten und Kontakte exportiert werden sollen. Die Daten werden dann in eine Datei geschrieben und stehen im FOAF Format zum Download bereit. Alternativ steht dem Nutzer auch eine Facebook Anwendung zur Verfügung, die einen Vergleich der Daten in Facebook und dem Prototypen anbietet.

6.5.3 Definition der zu exportierenden Daten

Folgende Daten können im- und exportiert werden:

FOAF Basics

- nick
- mbox
- surname
- firstName

Personal Info

- knows
- interest
- workplace
- homepage

Online Accounts / IM

- OnlineChatAccount

Für die Kontakte steht aus rechtlichen Gründen nur ein reduzierter Datensatz zur Verfügung:

- Nutzer-ID
- Netzwerkname
- Vorname
- Nachname
- Emailadresse

6.5.4 Visualisierung

Ausgangspunkt für die Visualisierung sind die Daten, die in den Profilen der Nutzer vorliegen.

Die Anzeige der Daten erfolgt in Form einer Tagcloud. Hierfür aggregiert das System zuvor die Profildaten und speichert die Häufigkeit der vorkommenden Begriffe. Aus der Tagcloud lässt sich ablesen, inwiefern bestimmte Daten überhaupt in dem Netzwerk vorhanden sind. Der Nutzer kann nun einen Begriff aus der Tagcloud auswählen und die Anzeige in Form einer Karte auslösen. Daraufhin werden die aktuellen Positionen der Nutzer, bei denen dieser Begriff im Profil vorkommt, ausgelesen. Über die Karte wird dem Nutzer dann der Zugriff auf das ausführliche Profil der Nutzer ermöglicht.

6.6 IMPLEMENTIERUNG

Für die Implementierung des Prototypen fiel die Wahl auf die Software Clearspace Community. Grundlage dafür waren insbesondere die folgenden zwei Gründe:

(1) Bei der durchgeführten Analyse schnitt die Software Clearspace Community in fast allen Belangen sehr gut ab (vgl. 5.5.2 auf Seite 50). Für die Implementierung des Prototypen war insbesondere die vorhandenen Möglichkeiten der API von Belang. Neben der bei den Frameworks üblichen REST API können in Clearspace auch in Java entwickelte Plugins direkt ins System integriert werden. Zusätzlich bietet die API Zugriff auf alle für den Prototypen notwendigen Personenattribute.

(2) Auch bei der Evaluierung einer kommerziellen Software sollte die Lauffähigkeit des Prototypen über die Erstellung der Diplomarbeit hinaus gewährleistet sein. Clearspace Community bietet die Möglichkeit einer zeitlich unbegrenzten Evaluation mit fünf Nutzern. Zwar entfalten einige Komponenten des Prototyps erst mit einer hohen Nutzeranzahl ihren Nutzen², aber ein Grundeindruck erhält man bereits mit fünf Nutzern.

6.6.1 Funktionalitäten für die Problemidentifikation bei einer Datenaggregation

Wie Eingangs beschrieben verlangt die immer größere Anzahl an Sozialen Netzwerken eine Datenaggregationsfunktionalität. Auf Grundlage der in Abschnitt 3.2 auf Seite 20 analysierten Funktionen und insbesondere durch die Analyse der Möglichkeiten hinsichtlich der Datenspeicherung im Profil bei bestehenden Sozialen Netzwerken (vgl. Abschnitt 3.2.10 auf Seite 25) und Frameworks (vgl. Abschnitt 5 auf Seite 42) ist der Schluss zulässig, dass die folgenden Funktionalitäten von besonderem Interesse für den Nutzer sind.

Auf Grund der Vielzahl an Daten, welche in den Sozialen Netzwerken abgelegt werden können und mit denen die Netzwerke erst ihren wahren Nutzern entfalten, hat die Möglichkeit eines Datenimports oberste Priorität. Wie in Abschnitt 4.2 auf Seite 30 und 4.3 auf Seite 37 erläutert, bieten viele Netzwerke (Facebook, Bebo usw.) die Möglichkeit an, Daten über eine API zu exportieren. Die Wiederverwendung von bereits eingepflegten Daten stellt für den Nutzer sicherlich die komfortabelste Möglichkeit für den Import da, weshalb der Datenimport über eine Schnittstelle aus einem anderen Sozialen Netzwerke eine wichtige Funktionalität darstellt. Wie in Abschnitt 4.3.6 auf Seite 39 aufgezeigt, unterscheidet sich die Qualität der APIs deutlich, weshalb die Nutzer ebenfalls die Möglichkeit haben sollten, über ein standardisiertes Format Daten aus einer Quelle ihrer Wahl zu importieren. Dies wurde bei der Implementierung des Prototypen mit der Möglichkeit realisiert, Daten im FOAF Format zu importieren. Neben den Daten der Nutzer bestehen Soziale Netzwerke im Wesentlichen aus Kontakten. Der Nutzer hat folglich Interesse daran, seine Kontakte von einem Netzwerke in ein anderes zu transferieren oder aber die bereits existierenden Daten auf seiner lokalen Festplatte zu nutzen. Analog zum Datenimport ist deshalb auch eine Kontaktimportfunktion, entweder im FOAF Format oder über eine API, für den Nutzer interessant.

Aus wirtschaftlicher Sicht sicher gewollt, aber für den Nutzer bei zukünftigen Datenaggregationen hinderlich, ist die Geschlossenheit der Sozialen Netzwerke nach außen. Wie in Abschnitt 4.3 auf Seite 37 erläutert, bieten bestehende APIs oft nur die Verwendung von Daten und Kontakten in anderen Webseiten an, nicht aber deren Export. Um dem Nutzer jedoch den vollständigen Zugriff auf seine Daten zu ermöglichen, soll eine Möglichkeit des standardisierten Exports im FOAF-Format implementiert werden. Wie die Evaluierung der Frameworks im Abschnitt 5.5 auf Seite 47 zeigt, existiert teilweise auch die Möglichkeit eines strukturierten Zugriffs auf die Daten über die

²Beispielsweise zeigt die Kartendarstellung und die Tagcloud erst ab einer gewissen Nutzeranzahl ihren Nutzen.

API. Da dies wie beim Import in der Regel - auf Grund der bereits bestehenden Strukturierung - die vom Nutzer bevorzugte Möglichkeit ist, soll eine Funktion in den Prototypen integriert werden, die den Zugriff von einem externen System auf die im Prototypen gespeicherten Daten erlaubt.

Wie im Abschnitt 1.1 auf Seite 2 ausgeführt, entfalten die Nutzerdaten erst in alternativen Ansichten ihr volles Potential. Um dem Nutzer einen visuellen Eindruck der erfolgten Datenaggregation zu ermöglichen, soll eine Darstellung dieser in Kartenform ermöglicht werden.

Über die vorgestellten Funktionalitäten sollen häufige Vorgänge des Nutzers bei der Datenaggregation abgedeckt werden. Aus technischer Sicht ermöglichen die gewählten Funktionalitäten die folgenden Einblicke:

Daten und Kontaktimport aus einer XML Datei im FOAF Format Hierbei soll untersucht werden, inwiefern der Import aus einem standardisierten Format in der Praxis Probleme bereitet. Fragestellungen, die mit dieser Funktion untersucht werden können, beinhalten die Thematik, ob Abweichungen gegenüber der Spezifikation möglich sind und ob sich das FOAF Format für diese Art von Aggregation eignet.

Daten und Kontaktimport über eine Schnittstelle direkt aus einem anderen Sozialen Netzwerk Mit dieser Funktion soll insbesondere überprüft werden, inwiefern sich der Import aus anderen Netzwerken von einem Import aus einer Datei unterscheidet. Relevante Fragen sind hierbei, welche Beschränkungen die Betreiber der anderen Sozialen Netzwerke dem Export auferlegt. Auch mögliche technische Besonderheiten bei der Anbindung sollten untersucht werden.

Daten und Kontaktexport in eine XML Datei im FOAF Format Um aufzuzeigen, welche Probleme sich beim Export ergeben können, soll beispielhaft ein Export in das FOAF Format realisiert werden. Neben der Tauglichkeit des Formats ergibt sich damit auch die Möglichkeit der direkten Verarbeitung durch die Importfunktion. Wichtige Fragen sind hierbei die Abbildbarkeit der Inhalte im FOAF Format und die technische Weiterverwendbarkeit des Exports

Daten und Kontaktexport über eine Schnittstelle direkt aus einem anderen Sozialen Netzwerk Hierbei stand insbesondere die Überprüfung im Vordergrund, wie Daten über die Webservice API in ein anderes Netzwerk eingelesen werden können. Neben den Beschränkungen beim Auslesen wurden auch die Beschränkungen beim Schreiben in einem anderen Netzwerk in Betracht gezogen.

Darstellung der aggregierten Daten Um einen Verwendungszweck für die importierten Daten darzustellen, soll beispielhaft eine Darstellung der Daten auf einer Landkarte realisiert werden. Dabei wird analysiert, welche Darstellungsmethoden hierbei sinnvoll sind und der Erfolg der zuvor durchgeführten Vorgänge aufgezeigt.

6.6.2 Integration in Clearspace

Clearspace bietet die Möglichkeit, Plugins über die Adminkonsole ins System zu integrieren. Plugins müssen als jar-Datei vorliegen. Über die Plugins können die folgenden Arten von Integration implementiert werden:

1. Menü im Header

2. Menüpunkt im Aktionsmenü einer Community
3. Menüpunkt im Aktionsmenü im Profilbereich des Nutzers
4. Tab im Communitybereich
5. Tab im Profilebereich des Nutzers
6. Widgets in den Communitybereichen

Für den Prototyp fiel die Wahl auf die Integration eines Widgets im Communitybereich und ein Tab im Profilbereich des Nutzers. Der Profiltab wurde für den Im- und Export gewählt, da dieser nur vom Nutzer abrufbar ist. Für die Kartendarstellung wurde die Anzeige als Widget gewählt, da so eine flexible Einbindung in die Communities gewährleistet wird. Um die Installation einfach zu gestalten, wurden beide Komponenten in einem Plugin zusammengefasst.

Eine Demoversion der Clearspace Software kann von der Webseite bezogen werden, liegt in Version 2.0.5 aber ebenfalls auf der CD mit dem entsprechenden Lizenzschlüssel bei. Getestet wurde die Installation unter Windows XP, Ubuntu und Mac OS X Leopard. Ein Beispielprojekt für ein Plugin kann bei Clearspace bezogen werden (vgl. [85]).

Innerhalb des Projekts steuert die „struts.xml“ die Zuordnung von Template Dateien und den Java Klassen. Als Template Engine wird Freemarker (vgl. [54]) verwendet. Die Benennung der Menüpunkte und Tabs erfolgt über die „plugin.xml“ Datei. Bis auf Konditionen und das Auslesen von Arrays ist die Programmlogik in den Javaklassen implementiert.

Um eine Weiterentwicklung des Prototypen zu ermöglichen, sind im Folgenden einige der Implementierungsansätze beschrieben:

Aggregation von Kontakten In der vorliegenden Version von Clearspace gibt es für zwei Mitglieder keine Möglichkeit, sich direkt miteinander zu verbinden. Um dennoch die Möglichkeit eines Kontaktexports zu schaffen, wurde ein Algorithmus implementiert, der alle Personen als Kontakte exportiert, die sich in den gleichen Gruppen wie der exportierende Nutzer befinden. Die Kapselung der Zusammenstellung der Liste in einer Funktion, wie im Quellcode 6.1 dargestellt, ermöglicht den leichten Austausch durch eine andere Methode.

```

1 GroupManager gm = JiveApplication.getContext().getGroupManager();
2 Iterable<Group> ug = new HashSet<Group>();
3 ug = gm.getUserGroups(getUser());
4 List<User> contactList = new ArrayList<User>();
5 List<User> memberList = new ArrayList<User>();
6 Iterator<Group> iterator = ug.iterator();
7
8 while(iterator.hasNext()) {
9     memberList = gm.getGroup(iterator.next().getID()).getMembers();
10    Iterator<User> iterator1 = memberList.iterator();
11    while(iterator1.hasNext()) {
12        User currentUserToAdd = iterator1.next();
13        if(contactList.contains(currentUserToAdd) != true) {
14            contactList.add(currentUserToAdd);
15        }
16    }
17 }

```

Quellcode 6.1: Zusammenstellung der Kontaktliste

Tagcloud Für die Kartendarstellung wurde eine Tagcloudanzeige entwickelt. Wesentliches Element ist dabei die Formel zur Berechnung der Schriftgröße, die im Quellcode 6.2 gezeigt ist. Die Berechnung erfolgt auf Grundlage des Profelfeldes „Interessen“. Die dortigen Einträge werden anhand des Leerzeichens separiert und als Tags interpretiert.

```

1 (Math.log(occurrencesOfCurrentTag)-Math.log(minOccurs))/(Math.log(maxOccurs)-Math.log(minOccurs));

```

Quellcode 6.2: Formel zur Berechnung der Schriftgröße in der Tagcloud (vgl. [153])

Zuordnung von Profildatensätzen Die Zuordnung der zu importierenden Daten zu den passenden Profildatensätzen erfolgt vom Nutzer per Drag&Drop. Die zu importierenden Felder werden dabei dynamisch aus der XML Datei generiert. Für jeden Importvorgang kann eine Zuordnung erstellt werden, welche separat für jeden Nutzer abgespeichert wird und so für zukünftige Importvorgänge zur Verfügung steht. Wählt der Nutzer keinen Zuordnungsnamen, wird seine Wahl automatisch unter dem Profil „Letzte Zuordnung“ gespeichert.

Datenaggregation von Facebookdaten Für die Datenaggregation bei Facebook muss dort zuerst eine Applikation angelegt werden. Als „Application Type“ muss „Website“ angegeben werden und als Callback URI wird „http://localhost:8080/clearspace_community/example-include-facebook.jspa“ verwendet. Als API Wrapper wurde die OpenSource Java API (vgl. [42]) verwendet, da von Facebook keine offizielle Java Bibliothek angeboten wird. Im Plugin selber wird über den im Quellcode 6.3 angegebenen Code ein Token erzeugt und an Facebook übertragen. Die Rückgabewerte, wie im Quellcode 6.4 dargestellt, können dann als String ausgegeben oder wie in dem Fall des Prototypen als XML Datei im FOAF Format gespeichert werden. Die Speicherung der Werte in der XML Datei erfolgt im Hintergrund als Ajax Request.

```

1 String API_KEY = "xxx";
2 String SECRET = "xxx";
3 FacebookRestClient client = new FacebookRestClient(API_KEY, SECRET);
4 client.setIsDesktop(false);
5 token = client.auth_createToken();
6 AppLoginURL = "http://www.facebook.com/login.php?api_key=" + API_KEY + "&v=1.0" + "&
  auth_token=" + token;

```

Quellcode 6.3: Facebook API Token generieren und Autorisierung anfordern

```

1 Long userId = client.users_getLoggedInUser();
2 Set<ProfileField> friendsattributes = new HashSet<ProfileField>();
3
4 friendsattributes.add(ProfileField.FIRST_NAME);
  friendsattributes.add(ProfileField.LAST_NAME);
5 friendsattributes.add(ProfileField.NAME);
6 friendsattributes.add(ProfileField.BIRTHDAY);
7
8 client.friends_get();
9 FriendsGetResponse responsefriends = (FriendsGetResponse) client.getResponsePOJO();
  List<Long> friends = responsefriends.getUId();
10
11 client.users_getInfo(friends, friendsattributes);
12 UsersGetInfoResponse friendsResponse = (UsersGetInfoResponse) client.getResponsePOJO();
13 List<User> friendsusers = friendsResponse.getUser();
14
15 client.friends_get();
16 FriendsGetResponse response = (FriendsGetResponse) client.getResponsePOJO();
17 List<Long> me = new ArrayList<Long>();
18 me.add(userId);
19 Iterator<Long> meiterator = me.iterator();
20
21 Set<ProfileField> meAttributes = new HashSet<ProfileField>();
  meAttributes.add(ProfileField.FIRST_NAME);
22 meAttributes.add(ProfileField.LAST_NAME);
23
24 client.users_getInfo(me, meAttributes);
25 UsersGetInfoResponse userResponse = (UsersGetInfoResponse) client.getResponsePOJO();
26 List<User> users = userResponse.getUser();

```

Quellcode 6.4: Facebook API Antwort auswerten

Auslesen der XML Datei für den Import Die XML Datei für den Import muss im FOAF Format vorliegen. Ist diese Bedingung erfüllt, können beliebige Profildaten ausgelesen werden. Die Parsingfunktion ist so angelegt, dass alle Kinderknoten des Elements „Person“ ausgelesen werden. Als Testfall wurde die Kompatibilität zum Facebook FOAF Generator von Matthew Rowe (vgl. [139]) herangezogen.

Zugriff auf den SOAP Webservice aus Facebook heraus Der Zugriff für den Export über die Web API erfolgte von Facebook aus mit Hilfe einer in PHP geschriebenen Applikation. Der Zugriff auf Clearspace wird über die SOAP API realisiert. Der Quellcode 6.5 zeigt eine der verwendeten Methoden, die einen Vergleich mit denen in Facebook gespeicherten Daten ermöglichen. Der Quellcode 6.6 zeigt den Lesezugriff auf die in Facebook gespeicherten Daten.

```

1 $url = 'http://localhost:8080/clearspace_community/rpc/soap/UserService?wsdl';
2 $username = 'admin';
3 $password = 'xxx';
4
5 // Assemble the authentication header
6 $authheader = sprintf('
7 <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
8 wssecurity-secext-1.0.xsd"> <wsse:UsernameToken>
9 <wsse:Username>%s</wsse:Username><wsse:Password>%s</wsse:Password>
10 </wsse:UsernameToken>
11 </wsse:Security> ',
12 htmlspecialchars($username),
13 htmlspecialchars($password));
14
15 // Create the SoapHeader using a Soapvar. there's some magic here.
16 $authvars = new SoapVar($authheader, XSD_ANYXML);
17 $header= new SoapHeader("http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
18 wssecurity-secext-1.0.xsd", "Security", $authvars);
19 // Turn off tracing for better performance. $soapclient = new SoapClient($url, array('
20 trace'=>1));
21
22 $params = array('emailAddress'=>"email@email.com" );
23 try {
24 $response = $soapclient->__soapCall('getUserByEmailAddress', array('parameters'=>$params),
25 NULL, $header);
26 $response = obj2array($response);
27 }
28 catch (Exception $e) {
29 var_dump($e->getMessage());
30 var_dump($soapclient->__getLastRequest());
31 var_dump($soapclient->__getLastResponse());
32 }

```

Quellcode 6.5: Zugriff per SOAP aus einer Facebook Applikation (vgl. [86])

```

1 $appapikey = '7c225cb7a56cc4f974fd1fc4158200c8';
2 $appsecret = 'xxx';
3 $facebook = new Facebook($appapikey, $appsecret);
4 $user_id = $facebook->require_login();
5 $user_details=$facebook->api_client->users_getInfo($user_id, array('last_name', 'first_name',
6 'current_location'));

```

Quellcode 6.6: Zugriff über die Facebook API auf Nutzerdaten

Implementation des Geocodings Für das Geocoding und die Anzeige der Benutzer wird auf die API von Google Maps zurückgegriffen (vgl. [60]). Beim Aufruf der Kartenkomponente wird überprüft, ob schon ein Geocoding für den Nutzer vorliegt³. Der Vorgang wird nicht clientseitig sondern direkt vom Server aus durchgeführt, wie im Quellcode 6.7 dargestellt. Liegt noch kein Geocoding für den Nutzer vor, wird dieses durchgeführt und die Werte anschließend im Nutzerprofil abgespeichert. Die Kartendarstellung wird über einen JSON String erzeugt, der ebenfalls beim Aufruf aus den Profildaten erzeugt wird. Der JSON String wird je nach Wahl des Nutzers für die entsprechenden Tags erzeugt und verringert somit den Generierungsaufwand.

```

1 String q = URLEncoder.encode(address, "UTF-8");
2 String key = URLEncoder.encode("ABQIAAAo277Z4wyQHvldMnDD2MJhTwM0brOpm-All5BF6PoaKbXRWW",
3 "UTF-8");
4 String output = URLEncoder.encode("xml", "UTF-8");
5 URL u = new URL("http://maps.google.com/maps/geo?" + "q=" + q + "&key=" + key + "&output="
6 + output);
7
8 InputStream xmldocument = u.openStream();
9 int c;
10 String xmlstring = "";
11 while((c = xmldocument.read()) != -1){
12 xmlstring = xmlstring + (char) c;
13 }

```

³Dies reduziert die Anzahl der Geocodinganfragen an Google und verhindert somit ein Überschreiten der zulässigen Transaktionen.

```

12 }
13
14 File file = new File("ablage-google-antwort.xml");
15 OutputStreamWriter out = new OutputStreamWriter(new FileOutputStream(file), "UTF-8");
16 PrintWriter ausgabe = new PrintWriter(out);
17 ausgabe.println(xmlstring);
18 ausgabe.close();
19
20 String filename = "ablage-google-antwort.xml";
21 Document doc = new SAXBuilder().build(filename);
22
23 Element party = doc.getRootElement();
24 Namespace xmlns = Namespace.getNamespace("http://earth.google.com/kml/2.0");
25
26 if (party.getChild("Response", xmlns).getChild("Status", xmlns).getChildText("code", xmlns).
    compareTo("200") == 0) {
27     Element person = party.getChild("Response", xmlns).getChild("Placemark", xmlns).
        getChild("Point", xmlns);
28 }

```

Quellcode 6.7: Anfrage an Google Maps API

Kommunikationsstruktur des Plugins Das realisierte Plugin besteht im Wesentlichen aus fünf Elementen:

- Dem eigentlichen Plugin: Social Network Aggregation Tool (SNAT)
- Clearspace API
- Facebook API
- Google Maps API
- Dateisystem / XML Datei

Die Kommunikationsabläufe zwischen den einzelnen Elementen sind in den Abbildungen 6.1 auf der nächsten Seite, A.1 auf Seite 138, A.2, A.3 und A.4 dargestellt. Nutzeraktionen sind fett gedruckt und dadurch ausgelöste Aktionen des Systems kursiv. Durch die Segmentierung der Kommunikationsprozesse ist ein Austausch der einzelnen Elemente mit geringem Aufwand möglich. Sämtliche Importprozesse werden über eine XML Datei abgewickelt. Sowohl der Import aus einer Datei als auch der Import über Facebook greifen auf diese Datei zu. Auch der Prozess des Geocodings ist leicht austauschbar, da er in einer eigenen Klasse vorgenommen wird, die als Eingabe eine Adresse verwendet und als Output die Geokoordinaten liefert. Zentraler Punkte für die Speicherung und den Abruf von Nutzerdaten ist die Clearspace API. Die Programmablauffolge ist im SNAT Plugin implementiert.

Eignung des Frameworks für die Implementierung Zusammenfassend lässt sich feststellen, dass die durch die Analyse geweckten Erwartungen an Clearspace weitestgehend erfüllt wurden. Ohne Quellcode zu verändern, konnte an den entscheidenden Stellen Funktionalitäten hinzugefügt werden, die den Zugriff auf alle erforderlichen Nutzerdaten ermöglichen. Auch das Editieren des Profils über die API war im vollen Umfang möglich. Etwas gewöhnungsbedürftig war die Struktur des Nutzerprofils, welches aus dem eigentlichen Profil, erweiterten Profildaten und den eigentlichen erweiterten Nutzerdaten besteht. Hinderlich für die Implementierung gestaltet sich die Notwendigkeit eines Systemsneustarts von Clearspace bei jeder Änderung im Code. Zwar können neue Plugins zur Laufzeit geladen werden, aber Codeveränderungen werden nicht registriert. Die Anbindung an die Debug Komponente von Eclipse ist wiederum möglich. Die Dokumentation der Java API erwies sich als fast vollständig⁴, allerdings wären einige erklärende

⁴Eine nicht dokumentierte Klasse wurde während der Implementierung gefunden.

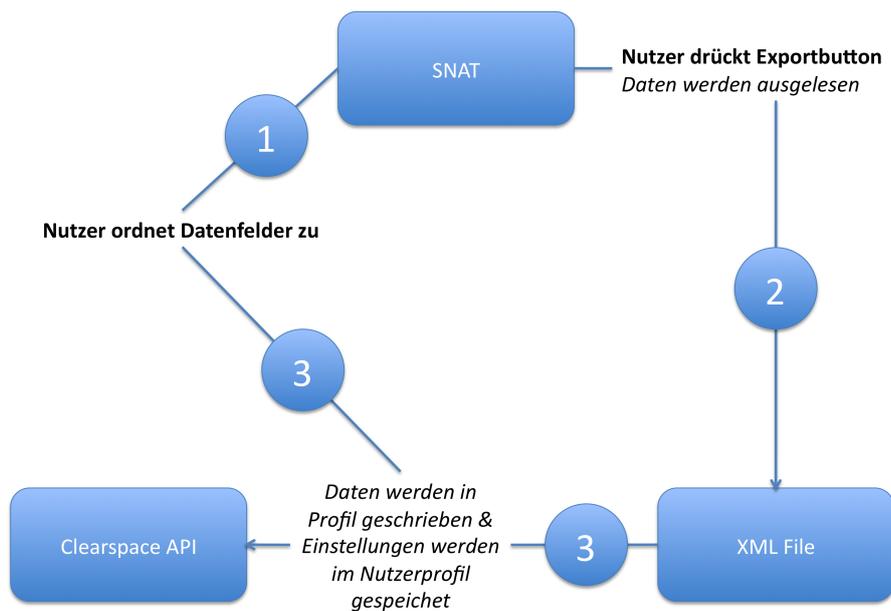


Abb. 6.1: Kommunikationsdiagramm zum Import von Daten aus einer Datei

Beispiele hilfreich gewesen. Wenige Informationen waren zu den in FreeMarker verfügbaren Befehlen vorhanden⁵. Die Einbindung von GoogleMaps in ein Widget war ebenfalls ohne Probleme möglich. Anfragen im Forum der Entwicklergemeinde wurden von einem Clearspace Mitarbeiter zeitnah beantwortet; außerdem stand Beispielquellcode zur Verfügung. Hilfe zwischen den Entwicklern war in der Community wenig bis gar nicht verbreitet. Förderlich für das Verständnis von manchen Funktionalitäten ist sicherlich, dass beim Erwerb der Software ein vollständiger Zugriff auf den Quellcode möglich ist.

6.7 AGGREGATIONSPROBLEME BEI DER IMPLEMENTIERUNG

Nach den Erläuterungen zum Aufbau des Plugins und zur Eignung der Clearspace API für die Implementierung sollen in diesem Abschnitt die Probleme bei der Datenaggregation anhand der in Abschnitt 6.3 auf Seite 72 aufgezeigten Problemfelder vorgenommen werden.

Authentifizierung Die Authentifizierung gestaltet sich technisch unproblematisch. Über die entsprechende API bei Facebook kann der Login vom Nutzer problemlos vollzogen werden. Für die Dauer der Session kann das Programm dann mit den Rechten des Nutzers auf die Daten zugreifen. Probleme ergeben sich eher aus datenschutzrechtlicher Sicht, da bei Facebook der Zugriff auf fast alle Daten⁶ möglich ist, die der Nutzer bei seinen Kontakten normalerweise auch sieht. Allerdings besteht durch die API nun die Möglichkeit einer automatisierten Verarbeitung. Die Exportbeschränkung der Emailadresse verhindert allerdings die effiziente Weiterverarbeitung der Daten in anderen Netzwerken. Zwar bietet Facebook die Möglichkeit an, einen Emailadressen-hash zu beziehen, dieses Feld war aber bei den vorgenommenen Versuchen leer. Somit fehlt ein eindeutiges Identifikationsmerkmal, was sich insbesondere bei dem Import von Kontakten - wie im Folgenden erläutert - negativ bemerkbar macht.

⁵So kann der aktuelle Nutzer beispielsweise über „user“ angesprochen werden, was sich aber in keiner Dokumentation fand.

⁶Die Emailadresse ist hiervon ausgenommen.

Datenimport Der Import von Daten wirft im Gegensatz zur Authentifizierung einige Probleme auf. Drei Problembereiche konnten hierbei identifiziert werden:

- Eindeutiges Identifikationsmerkmal
- Import von Kontakten
- Fremde und eigene Datenquellen

Die von Facebook exportierten Daten enthalten kein globales Identifikationsmerkmal. Zwar kann die Facebook ID exportiert werden, diese besitzt aber in anderen Systemen keine Abbildung auf einen bestimmten Nutzer. Üblicherweise wird die Emailadresse des Nutzers verwendet, diese ist aber nicht für den Export freigegeben. Somit entfallen sämtliche Möglichkeiten für automatische Importvorgänge. Wirkt sich dies beim Import der eigenen Daten noch nicht gravierend aus, da hier der Nutzer zumeist seinen Account selber auswählt, verhindert es den sinnvollen Import von Kontakten. Alternativ wäre auch eine Identifikation über den Emailhash möglich - diese Exportmöglichkeit sieht die Facebook API auch vor - eine solche wird noch nicht generiert. Als unscharfe Möglichkeit bleibt noch die Identifizierung über den Namen. Allerdings ist hier immer eine Kontrolle des Anwenders notwendig, um die Vorschläge zu verifizieren. Dieser in der Implementierung verwendete Ansatz ist bei einer großen Anzahl von Kontakten sehr aufwendig, erscheint aber als der einzig gangbare Weg. Die einzige Alternative ist die Facebookseiten der Nutzer zu parsen. Allerdings schließen die Facebook Entwicklernutzungsbedingungen Zugriffe außerhalb der API aus (vgl. [38, 4, 100]).

Nicht gelöst werden kann die Frage nach Kontakten, die noch nicht im System existieren. Technisch möglich, aber datenschutzrechtlich bedenklich wäre die Möglichkeit, diese Nutzer automatisch anzulegen und ihnen ein Passwort zu senden. Allerdings stellt sich hier die Frage, inwiefern die Daten überhaupt verwendet und in diesem Fall auch noch gespeichert werden dürfen. So bleibt nur die Möglichkeit, existierende Kontakte darzustellen, einen Link zu ihrem Profil anzubieten und dem Nutzer aufzuzeigen, welche Nutzer noch nicht im System existieren.

Unproblematisch gestaltete sich der Import aus verschiedenen Datenquellen. Die gewählten Quellen waren einmal eine lokale Datei, die der Nutzer mit einem beliebigen FOAF Tool selber generieren kann, sowie der Import aus einem anderen Netzwerk. Bis auf die erwähnte Problematik des eindeutigen Identifikationsmerkmals waren hier keine Unterschiede auszumachen.

Datenexport Beim Export sind vorrangig rechtliche und technische Probleme zu beachten. Technisch muss der Zugriff auf alle Daten ermöglicht werden, was in der konkreten Implementierung kein Problem darstellte. Rechtlich stellt sich die schon beim Datenimport erörterte Frage, ob Daten von anderen Nutzern exportiert werden dürfen. In der konkreten Implementierung wurde eine Zwischenlösung gewählt, bei der der Nutzer nur die Namen seiner Kontakte exportiert. Denkbar wäre auch den Emailhash zu exportieren, da dieser nur von Nutzen ist, wenn man bereits über die Emailadresse eines Kontakts verfügt. Um dem Nutzer die Kontrolle über seine Daten zu gewährleisten, sollte dieser wählen können, welche Daten er exportieren möchte. Dies kann über ein Formular geschehen, wobei die zuvor gewählte Einstellung automatisch gespeichert wird. Für die Bereitstellung des Exports bietet sich als Speicherform eine Datei im FOAF Format an.

Ein oft gewählter Verwendungszweck der exportierten FOAF Dateien sind die Onlinespeicherung für einen einfachen Abruf durch andere Nutzer und Analysetools, die so Kontaktnetze darstellen können. Hierdurch ergibt sich jedoch das Problem des Datenmissbrauchs. Hauptsächlich stellen automatisierte Bots, die beispielsweise Emailadresse auslesen, eine Gefahr da. Eine mögliche Maßnahme ist die Hashs der Emailadresse zu speichern. Des Weiteren bieten die FOAF Dateien die Möglichkeit, Einblicke zu erhalten, wer mit wem bekannt ist. Da dies aber ein generelles Problem von Sozialen Netzwerken ist, sei diese Thematik hier nicht näher diskutiert.

Repräsentation der Daten Für die Repräsentation der Daten wurde das FOAF Format verwendet. Dies gestaltet sich solange unproblematisch, wie die vorliegenden Daten von der FOAF-Spezifikation erfasst sind. Um eine konsistente Datenhaltung und eine Weiterverarbeitung in anderen Programmen zu ermöglichen, müssten jedoch die Spezifikationen des Schemas erweitert werden. Bei dem entwickelten Plugin wurde der Import so flexibel gestaltet, dass theoretisch auch Daten außerhalb des Schemas importiert werden können. Durch das dynamische Einlesen der Knoten und die Darstellung der Inhalte kann der Nutzer selbständig die Zuordnung vornehmen. Eine automatisierte Verarbeitung ist damit aber ausgeschlossen. Problematisch gestaltete sich die Tatsache, dass in den FOAF Spezifikationen keine Adressinformationen vorgesehen sind. Ansonsten konnten die restlichen Daten (Interessen, Chatnicknames usw.) sinnvoll in das Schema exportiert werden.

Zusammenführung der Daten Neben dem „technischen“ Vorgang des Imports ergeben sich bei der eigentlichen Zusammenführung der Daten mit dem Zielsystem eine Reihe von Problemen. Sechs Problemfelder konnten während der Implementierung identifiziert werden:

- Zuordnung der passenden Profelfelder
- 1:n oder n:1 Beziehungen
- Unterschiedliche semantische Bedeutung
- Nicht existierende Profelfelder
- Wiederkehrende Importe
- Bereits existierende Daten

Eine automatische Zuordnung der Profelfelder ist in der Regel nicht möglich, da sich die Benennung der Felder hierfür zu sehr unterscheidet. Das FOAF Format definiert zwar die Benennung der Felder, allerdings ist damit weder die Häufigkeit des vorkommenden Datums berücksichtigt noch die Abfrage, ob der Nutzer überhaupt das Datum importieren möchte. Hinzu kommen Exporte, die nicht aus einer FOAF Datei vorgenommen werden, wie beispielsweise Daten aus Facebook. Hier ist die Benennung (fast) beliebig und kann nicht automatisch zugeordnet werden. Die Zuordnung muss also vom Nutzer eigenhändig vorgenommen werden, da nur er die semantische Bedeutung der Felder kennt. Für die manuelle Zuordnung bietet sich eine Drag&Drop Implementierung an. Wichtig ist, dass dem Nutzer neben dem eigentlichen Datum auch der Knoten- bzw. Feldname des Nutzers dargestellt wird, damit er ähnliche Daten unterscheiden kann.

Für 1:1 Beziehungen kann so, zuverlässig und mit vertretbarem Aufwand, eine Zuordnung erfolgen. Problematisch sind 1:n oder n:1 Beziehungen, wie sie beispielsweise bei Adressen mit der Kombination aus Hausnummer und Straße auftreten⁷. Bei der Zusammenführung von mehreren Daten zu einem Datum müsste dem Nutzer die Möglichkeit gegeben werden, die Reihenfolge des Einfügens zu bestimmen. Dies wirft aber im nächsten Schritt die Frage nach dem Datentyp auf. Fügt der Nutzer beispielsweise eine URI und eine Zahl zusammen, gehen die mit der URI verknüpften Aktionen verloren. Sollte im Zielsystem nur ein Feld für die Homepage verfügbar sein, obwohl der Nutzer zwei URIs (Blog und Homepage) einfügt, führt dies zu einem Fehlverhalten beim Klick auf die URI im Zielsystem.

Neben Datentypproblemen ergeben sich auch durch die semantische Bedeutung der Felder Schwierigkeiten. So kann zwar beispielsweise in beiden Systemen ein Feld für Reputation existieren, aber mit völlig unterschiedlichen Werteskalen versehen sein. Während in einem System die Skala von 1-10 reicht, geht sie in einem anderen von 100-1, wobei hier eins der höchste Wert ist, im

⁷Je nachdem wie die Daten abgebildet werden, sind Straße und Hausnummer ein Feld oder zwei.

anderen System aber 10 (vgl. als Beispiel 5.5.4 auf Seite 57). Da das FOAF Format keine Informationen über die Werteskala eines Datums vorsieht, bleibt hier nur eine manuelle Kontrolle des Benutzers bzw. der Ausschluss des Datums aus der Zusammenführung als mögliche Lösung.

Ebenfalls nur unbefriedigend kann technisch das Importieren von Profildfeldern gelöst werden, die nicht im Zielsystem existieren. Hierfür müssten automatisch neue Profildfelder vom Nutzer generiert werden können. Eine globale Definition dieser ist aber nicht möglich, da dies zu Inkonsistenzen und Dopplungen im System führen würde. Ein lokales Profildfeld würde nur zu einem geringen Nutzern führen, dass dies zwar durchsuchbar sein könnte, aber weder Typ noch Vorhandensein global bekannt gemacht werden könnte.

Gut abgebildet werden können dagegen wiederkehrende Importe. Auf Basis der Annahme, dass sich beim Im- und Export zwischen den gleichen Netzwerken die Felderbenennung nicht ändert, kann der Nutzer die Zuordnungen für sich abspeichern und beim erneuten Export wieder auswählen. Aus technischer Sicht treten hier keine neuen Probleme auf, da auf die bereits implementierten Algorithmen zurückgegriffen werden kann. Problematisch sind lediglich eine Veränderung der Feldnamen bei einem der beteiligten Netzwerke oder der Wunsch des Nutzers, beim erneuten Import andere Felder zu importieren. Im ungünstigsten Fall muss der Nutzer eine erneute Zuordnung vornehmen.

Neben einer vollständigen Zusammenführung soll der Nutzer auch die Möglichkeit haben, bestehende Daten im System zu belassen und nur von ihm bestimmte Felder zu füllen. Durch die gewählte Zuordnung über die Drag&Drop Methode konnte auch dieser Anwendungsfall abgedeckt werden. Nicht implementiert wurde der Fall der im Abschnitt Import 6.7 auf Seite 82 als problematisch beschriebene Zusammenführung zweier Daten in einem Feld.

Software- und Frameworkbezogene Probleme bei der Implementierung Neben den oben aufgeführten konzeptionellen Schwierigkeiten bei der Implementierung ergaben sich auch einige framework- und softwarespezifische Probleme.

Bei der Authentifizierung gegenüber Facebook erforderte die Generierung eines Tokens eine gewisse Aufmerksamkeit. Facebook unterscheidet zwischen Desktop- und Webanwendungen. Je nach Anwendung kann eine unterschiedliche Authentifizierung verwendet werden. Der implementierte Prototyp ist zwar ein externes Programm, wird aber durch die Ausführung auf einen eigenen Webserver von Facebook als externe PHP Applikation behandelt. Um Zugriff über die Java Wrapper API auf die REST API zu bekommen, musste als Applikationstyp „client.setIsDesktop(false)“ gesetzt werden. Trotz der Verwendung als Webapplikation war das Auslesen der von Facebook zurückgelieferten Variable „\$_GET[‘auth_token’]“ nicht nötig.

Problematisch erwies sich der Umstand, dass die Facebook API zwar das Auslesen sämtlicher freigebender Daten zulässt, aber über die API keine Möglichkeit vorsieht, Daten in das eigene Profil zu schreiben. Aus diesem Grund musste auch die entwickelte Facebook Applikation auf einen reinen Vergleich der Daten in den zwei Systemen begrenzt werden.

Nicht ganz einfach gestaltete sich das Zusammenspiel von Java und der Google Maps API. Die von der Google API zurückgelieferte XML Datei sollte laut Beschreibung im UTF-8 Format vorliegen, aber erst das erneute Erzwingen von UTF-8 beim Schreibvorgang, wie im Quellcode 6.8 dargestellt, ermöglichte das korrekte Auslesen der Datei. Hinzu kam noch, dass Längen und Breitengrad in der XML Datei in umgekehrter Reihenfolge zur sonst üblichen Reihenfolge in der Google Map API ausgeliefert werden. Da es sich aber um ein Datenfeld handelt, war ein wie im Quellcode 6.9 auf der nächsten Seite dargestelltes Vertauschen nötig.

```
1 OutputStreamWriter out = new OutputStreamWriter(new FileOutputStream(file), "UTF-8");
```

Quellcode 6.8: Google Maps API Antwort in XML Datei schreiben

```
1 String[] coordinatsarray = coordinates.split(",");
2 coordinates = coordinatsarray[1] + "," + coordinatsarray[0];
```

Quellcode 6.9: Vertauschen der Geokoordindaten aus Google Maps Antwort

Für den Export wurden die Daten direkt in eine XML Datei in das Root Verzeichnis des Servers geschrieben. Dem Nutzer wird nach dem Export ein entsprechender Link zur Verfügung gestellt. Sowohl Firefox (3.0.1) als auch Safari (3.1.2) erlauben das Öffnen des Links per Mausklick nicht. Gibt man die URI allerdings direkt in die Adresszeile an, wird der Link ohne Probleme geöffnet.

Zu beachten ist die Tatsache, dass Checkboxen als Arrays übermittelt werden. Dies ist unabhängig davon, ob die Checkboxen auch mit einer Mehrfachbelegung angelegt wurden.

Um ein komfortables Parsen der XML Dateien zu ermöglichen wurde die Bibliothek „jdom“^[81] verwendet. Die benutzte Child-Funktion traversiert allerdings durch alle Kinder mit dem Knoten-namen „Person“ und schließt dabei die Kontakte im ersten Schritt nicht aus. Folglich war also eine zusätzliche Überprüfung notwendig.

Bei der Integration der Drag&Drop Funktion in Clearspace kam es zu einem CSS Konflikt, der die Boxen um 15 Pixel verschob. Dieser Wert wurde manuell im Skript angepasst, um so eine reibungslose Funktion der Drag&Drop Komponente zu ermöglichen.

Zusammenfassend lässt sich sagen, dass bis auf die frameworkspezifischen Rahmenbedingungen bei Facebook und die Eigenheit der Browser bei der Darstellung des Dateilinks alle anderen Probleme lösbar waren.

6.8 ALLGEMEINE AGGREGATIONSPROBLEME UND EMPFEHLUNGEN FÜR DIE REALISIERUNG VON AGGREGATIONEN

Nachdem im vorherigen Abschnitt die konkreten Herausforderungen bei der Implementierung einer Datenaggregation erläutert wurden, sollen in diesem Kapitel Schlussfolgerungen für zukünftige Datenaggregationsimplementierungen von Nutzer- und Kontaktdaten im Umfeld Sozialer Netzwerke im Vordergrund stehen. Dabei liegt der Schwerpunkt insbesondere auf den systemunabhängigen Empfehlungen. Um die Nachvollziehbarkeit der Ergebnisse zu gewährleisten, orientiert sich der Aufbau wieder an den fünf Problembereichen:

- Authentifizierung
- Datenimport
- Datenexport
- Repräsentation der Daten
- Zusammenführung der Daten

Authentifizierung Wie im vorangegangenen Abschnitt aufgeführt, gestaltete sich der Zugriff bezogen auf die Authentifizierungsmechanismen im konkreten Fall einfach. Allerdings war der Zugriff nur unter Verwendung der systemspezifischen API möglich. Dies hat zur Folge, dass für jede mögliche Datenaggregation aus einem anderen Netzwerk eine separate Implementierung vorgenommen werden muss. Wie die Untersuchung im Abschnitt 3.2.2 auf Seite 21 gezeigt hat, ist sowohl bei Facebook als auch bei anderen bestehenden Netzwerken der Zugriff nur über die systemeigene API möglich. Diese Tendenz bestätigte sich auch bei den Frameworks, wie im Abschnitt 5.5 auf Seite 47 ausgeführt wurde. Wie im Abschnitt 2.4.5 auf Seite 14 gezeigt, würde sich für den Vorgang der Authentifizierung auch OpenID eignen. Allerdings wurde dies an keiner Stelle für den Zugriff auf die API angeboten. OpenID wäre aus Gründen der Transparenz der Vorzug zu gewähren, da die Implementierung für jeden nachvollziehbar ist und die Möglichkeit bestünde es auf mehreren Seiten einzusetzen. Da die Betreiber der Sozialen Netzwerke bisher aber keine Möglichkeiten in diese Richtung bieten, bleibt nur die Möglichkeit, den Einsatz der jeweiligen API zu empfehlen, wobei angemerkt sei, dass hierbei mit wenigen Problemen zu rechnen ist.

Datenimport Vorrangiges Problem beim Import von Datensätzen ist das Vorhandensein einer eindeutigen ID. Bei der Prüfung einer Datenaggregation sollte dies als erstes Kriterium herangezogen werden. Als ID bietet sich die Emailadresse an. Aus Gründen des Missbrauchsschutzes könnte man auch den Hash der Emailadresse verwenden, wobei dies nur von Nutzen ist, wenn die Emailadresse im Zielsystem ebenfalls bekannt ist. Alternativ wäre als eindeutige ID noch eine Kombination aus einer ID des Quellsystems und einer URI möglich. Dieser Ansatz wurde beispielsweise beim FOAF Export aus Facebook gewählt[139]. Hierdurch erhält man zwar eine eindeutigen ID, allerdings besteht im Zielsystem keine Zuordnung und daher erweist sich dieser Ansatz in der Anwendung nicht als ausreichend.

Beim Import von Kontakten bieten sich grundsätzlich zwei Lösungen an. Zum einen besteht die Möglichkeit einer Anzeige von bereits existierenden Kontakten durch einen direkten Import, wie es auch beim Prototypen implementiert wurde. Wie in der Implementierung gesehen, gibt es aber nur sehr wenige Verwendungsmöglichkeiten für die importierten Daten über die bloße Anzeige dieser hinaus. Nutzbringender ist in diesem Fall die Verwendung von einem Service wie Facebook Connect. Hierbei werden die Daten nicht importiert. Stattdessen können Kontakte in einer beliebigen Applikation direkt mit einem Kontakt bei Facebook verknüpft werden. Dies hat den Vorteil, dass die Verknüpfung zum eigentlichen Profil des Nutzers erhalten bleibt, er aber auch gleichzeitig festlegen kann, welche Daten veröffentlicht werden dürfen. Der Nutzer selber muss die Verknüpfung bestätigen, womit auch Datenschutzprobleme eingeschränkt werden.

Aus technischer Sicht macht der Import aus einer eigenen Datenquelle (beispielsweise eine Datei auf der Festplatte) oder einem externen System keinen Unterschied (vgl. Abschnitt 6.7 auf Seite 82), da in beiden Fällen die Daten für den Import aufbereitet werden müssen. Aus Nutzersicht ist aber der Import aus einem anderen Netzwerk vorzuziehen. Damit entfällt zum einen der Umweg über die Speicherung einer Datei auf der Festplatte, außerdem muss der Nutzer kein separates Tool verwenden, um eine FOAF Datei zu erstellen. Zu beachten ist allerdings, dass für jedes externe System ein eigenes Mapping erstellt werden muss, da hier kein einheitliches Mapping verbreitet ist.

Als Zugriffsmöglichkeit auf externe Systeme ist REST oder SOAP zu empfehlen. Wie die Untersuchungen im Abschnitt 3.2.2 auf Seite 21 und 5.5 auf Seite 47 gezeigt haben, unterstützten fast alle Frameworks den Zugriff per REST. Da dies auf der Clientseite auch den Zugriff aus fast jedem System ermöglicht, ist dies die bevorzugte Herangehensweise für den Remotezugriff.

Datenexport Die Machbarkeit des Exports hängt im Wesentlichen von dem Framework des verwendeten Systems ab. Die Analysen im Rahmen dieser Untersuchung haben gezeigt, dass hier durchaus die Möglichkeit von weitreichenden Exporten besteht, sowohl im Bereich der bestehenden Netzwerke, als auch bei den Frameworks. Ein wichtiger Prüfstein bei der Implementierung ist der rechtliche Rahmen, insbesondere, wenn ein Export von Daten, die nicht unmittelbar dem agierenden Nutzer gehören, stattfindet. Als Exportformat, unter Beachtung der in Abschnitt 6.8 beschriebenen Einschränkungen der Abbildung der Daten, kann das FOAF Format empfohlen werden. Für das FOAF Format spricht unter anderem, dass die Datei von anderen FOAF Programmen nahtlos weiterverarbeitet werden kann. Allerdings besteht beispielsweise keine Möglichkeit eines direkten Imports in ein Adressbuch - hierfür bietet sich eher das hCard Format an. Für die Akzeptanz einer Exportfunktion soll dem Nutzer in jedem Fall die Möglichkeit gegeben werden festzulegen, welche Daten für den Export freigegeben werden sollen. Im Optimalfall hätte diese Einstellung dann auch Auswirkungen auf die Exporteinstellungen über die Webservice API, die von „fremden“ Systemen angesprochen wird.

Neben den technischen Schwierigkeiten im Quellsystem stellt die Möglichkeit des Schreibvorgangs im Zielsystem aber das wesentlich größere Problem dar. Selbst bei Facebook, welches eines der ausgefeiltesten APIs besitzt (vgl. 4.2 auf Seite 30) ist ein Schreibvorgang - auch für das eigene Profil - nicht möglich. Damit wird jede Exportmöglichkeit im Ansatz unterlaufen. Vor der Implementierung einer solchen Funktion sollte folglich die Prüfung von möglichen Quellsystemen im Vordergrund stehen. Erst im zweiten Schritt ist eine Überprüfung der Fähigkeiten des eigenen Systems notwendig.

Repräsentation der Daten Aus denen in Abschnitt 6.5.2 auf Seite 74 und 2.4 auf Seite 12 genannten Gründen (einfache Handhabung und die Möglichkeit Kontakte und Nutzerdaten abzubilden), wurde für den Prototypen das FOAF Format für den Export verwendet. Während das Format für diese konkrete Implementierung aus den oben genannten Gründen die passende Alternative war, kann es nicht uneingeschränkt für die generelle Verwendung empfohlen werden. Wie in Abschnitt 6.8 bereits beschrieben, sind die möglichen Datenfelder sehr beschränkt, u.a. fehlt die wichtige Möglichkeit, Adressinformationen abzulegen. Auch semantische Informationen über den Bedeutung bzw. die Definition eines Datums können nicht hinzugefügt werden. Zwar sieht FOAF durchaus eigene Erweiterungen durch Benutzer vor (vgl. [18]), allerdings verhindert dies wiederum die globale Verwendung. Durch eine Hinterlegung der Namespace URI kann zwar die Weiterverwendung gewährleistet werden, aber die globale Gültigkeit des Formats geht damit verloren. Verzichtet man auf den Export von Kontakten, stellt die hCard (vgl. 2.4.2 auf Seite 10) eine erwägenswerte Alternative dar. Problematisch ist ebenfalls die Verwendung von FOAF Dateien als öffentlicher Download auf Grund der in 6.5.2 auf Seite 74 thematisierten Missbrauchsproblematik. Eine eindeutige Empfehlung für weitere Implementierungen kann an dieser Stelle nicht getroffen werden. Dies ist sehr abhängig von der konkret geplanten Implementierung, insbesondere von der Anforderung Kontakte und Nutzerdaten gleichzeitig exportieren zu wollen. Eine mögliche Lösung könnte darin bestehen, direkt auf den Webservice eines externen Systems zuzugreifen, wobei hierbei für jeden Import ein systemspezifisches Mapping erstellt werden muss.

Neben den eigentlichen Nutzerdaten bestehen Soziale Netzwerke zum großen Teil aus nutzergenerierten Inhalt, wie die Analyse im Abschnitt 3.2 auf Seite 20 zeigt. Diese sind nur schwer bis gar nicht zu aggregierten. Die FOAF Spezifikationen sehen zwar die Möglichkeit vor, beispielsweise Angaben zu Dokumenten zu exportieren, allerdings wird damit noch nicht der Inhalt des eigentlichen Dokuments erfasst. Für diese weiteren Inhalte geht die Empfehlung eher dahin, auf einen direkten Zugriff zu setzen und im FOAF Dokument nur einen Verweis zu exportieren. Ist das Einbringen von Verweisen ein akzeptabler Weg, bietet das FOAF Format einige Möglichkeiten, da beispielsweise auch Verweise auf Bilder, Projekte und Veröffentlichungen möglich sind.

Zusammenführung der Daten Da es keine globale Bezeichnungen für die Datenfelder gibt, muss bei jeder Datenaggregation eine Zuordnung stattfinden. Das verwendete Drag&Drop Verfahren kann grundsätzlich empfohlen werden, weil es dem Nutzer eine einfache Möglichkeit bietet eine Zuordnung herzustellen. Problematisch wird dies allerdings bei großen Datenmengen. Die Felder würden hier den Rahmen des Anzeigbaren sprengen. Über Drag&Drop sollte dem Nutzer auch die Möglichkeit gegeben werden, nur bestimmte Daten zu importieren und bereits im Profil existierende Daten dort zu belassen.

Problematisch, insbesondere bei komplexen Aggregationen, gestaltet sich die Tatsache, dass im FOAF Format keine weitreichenden Annotationen⁸ abbildbar sind. Anders als bei der Zuordnung von Quell- zu Zielfeldern besteht hier fast keine Möglichkeit für den Nutzer korrigierend einzugreifen. Bei der Planung einer Aggregation sollte darauf geachtet werden, dass entweder keine Informationen exportiert werden, die eine Definition ihrer Bedeutung benötigen oder es muss sichergestellt werden, dass diese im Ziel- und Quellsystem gleich sind. Alternativ kann eine Kodierung der Felder mit Angaben zu möglichen Werten und deren Bedeutung direkt im Datenfeld vorgenommen werden. Dann muss das Zielsystem jedoch darauf ausgelegt sein, diese Informationen zu verarbeiten.

Bei dem Import von Datenfeldern, die noch nicht im System existieren, ist es empfehlenswert, einen Import dieser nicht zuzulassen. Eine weitere Lösungsmöglichkeit ist das Anlegen eines nutzerspezifischen Feldes, welches allerdings weitestgehend von den feldspezifischen Funktionalitäten ausgeschlossen ist. Somit können die Inhalte der Felder nur einen geringen Nutzen entfalten, und zwar lediglich in dem Fall, wenn ein Nutzer direkt einen anderen Nutzer aufruft oder eine globale Stichwortsuche startet. Allerdings ist für diesen Fall auch ein einfaches Textfeld ausreichend.

Als Hilfestellung für den Nutzer soll die Möglichkeit bestehen, ein Mapping eines bereits durchgeführten Imports zu speichern. Grundsätzlich stehen hier zwei Möglichkeiten zur Verfügung: Entweder wird das Mapping nur nutzerspezifisch gespeichert oder aber global. Letzteres hat den Vorteil, dass nicht jeder Nutzer ein eigenes Profil erstellen muss - allerdings muss er sich dann auf die bereits gemachten Einstellungen verlassen⁹. Das Vorgehen hängt damit im Wesentlichen von der Homogenität der Nutzergemeinde ab, wobei die lokale Lösung einfacher handhabbar ist. Zusätzlich ist die Bereitstellung von Mappings seitens der Netzwerkbetreiber wünschenswert.

Sowohl aus technischer als auch aus Nutzersicht nicht befriedigend lösbar ist die in Abschnitt 6.8 beschriebene Problematik des 1:n und n:1 Mappings. Hier kann deshalb keine abschließende Empfehlung ausgesprochen werden.

Allgemeine Empfehlungen Um den Zugriff für eine Datenaggregation zu ermöglichen, muss neben der eigentlichen API auch eine Wrappersprache vorhanden sein. Zwar ist theoretisch der REST Zugriff von jeder beliebigen Programmiersprache möglich, allerdings gestaltet sich der Zugriff über eine Wrappersprache wesentlich komfortabler und ermöglicht auch den Zugriff auf Funktionen, die nicht über die Schnittstelle verfügbar sind, wie beispielsweise Layoutelemente. Das selbst prominente Netzwerke verbreitete Programmiersprachen nicht mehr offiziell unterstützen, zeigt das folgende Beispiel: Facebook hat den offiziellen Support von Java im Mai 2008 eingestellt (vgl. [46]). Seitdem ist man auf die Weiterentwicklungen durch die Entwicklergemeinde angewiesen.

Bezüglich der Analyse passender Frameworks für eine Datenaggregation lässt sich sagen, dass grundsätzlich kein Unterschied bei der Eignung von gehosteten Frameworks und selbst zu installierenden Frameworks besteht. Wie die Analyse von Clearspace (vgl. 5.5.2 auf Seite 50) und Ning (vgl. 5.5.3 auf Seite 53) zeigt, sind grundsätzlich beide Netzwerke gleich gut für eine Aggregation geeignet. Zu berücksichtigen ist natürlich, dass dies u.a. der Tatsache geschuldet ist, dass auch

⁸Beispielsweise Wertebereiche für ein Datum.

⁹Denkbar sind hier sowohl falsche Zuordnungen, als auch nicht zugeordnete Felder.

	Nutzer	Technisch	Rechtlich	Gesamt
Authentifizierung	5	5	5	5
Datenimport	5	3	3	3.6
Datenexport	5	4	2	3.3
Zusammenführung der Daten	3	2	-	2.5
Repräsentation der Daten	-	3	-	3

Tab. 6.1: Bewertungsmatrix für eine Datenaggregation

bei Ning ein weitgehender Zugriff auf den Quellcode möglich ist. Hier ist eine Empfehlung für ein bestimmtes Framework sehr klar von den spezifischen Eigenschaften desselben abhängig.

Anders verhält sich die Situation, wenn die Anbindung an ein bestehendes Soziales Netzwerk geplant ist. Zwar verfügen die meisten Netzwerke über APIs (vgl. 4.2.7 auf Seite 36), allerdings geht der Trend dahin, eher eine Verknüpfung der vorliegenden Daten anzubieten (4.3.6 auf Seite 39). Diese Entwicklung ist insofern beachtenswert, da sie den Export in Zukunft eher erschweren wird, da die Betreiber natürlich ein Interesse daran haben den Export zu verhindern und stattdessen ihre Stellung als zentraler Datenhalter zu stärken. Hier besteht also eine enge Abhängigkeit von den Interessen der Betreiber. Für Neuentwicklungen sollte deshalb auf Technologien gesetzt werden, die eine Verknüpfung der Kontakte herstellt.

Tabellarische Auswertung In der Tabelle 6.1 sind die Ergebnisse der Aggregationsuntersuchungen aufgetragen. Die Skala reicht dabei von 1 (sehr problematisch) bis 5 (unproblematisch) bei der Implementierung. Kein Wert wurde angegeben, wenn beispielsweise eine Nutzerinteraktion nicht notwendig und somit auch nicht bewertbar ist. Deutlich zeigt sich, dass die Zusammenführung der Daten am problematischsten ist. Datenimport- und Export werden vor allem durch die rechtlichen Probleme erschwert, aber auch durch die Unsicherheit in Bezug auf die entsprechenden API Funktionen. Für die Repräsentation der Daten ist nur die technische Betrachtung relevant, allerdings stellt der technische Teil auch die größte Herausforderung da¹⁰. Die Authentifizierung gestaltet sich in allen drei Bereichen unproblematisch.

6.9 ZUSAMMENFASSUNG

In diesem Kapitel wurden die zuvor erarbeiteten Ergebnisse (Evaluierungsmatrix für Soziale Netzwerke, Frameworks und deren APIs) dazu verwendet, eine Datenaggregation zu konzipieren und implementieren.

Bei der Implementierung konnte sowohl auf die Erkenntnisse der Frameworkuntersuchung als auch auf die Diskussion zu den technischen Grundlagen (REST, FOAF usw.) zurückgegriffen werden (vgl. 2.4 auf Seite 8). Auf Basis dieser Analyse wurden die eingesetzten Technologien ausgewählt. Wie in der Auswertung der spezifischen Probleme bei der Datenaggregation beschrieben (vgl. 6.7 auf Seite 81), eigneten sich die gewählten Techniken (insbesondere die Verwendung von FOAF und die Nutzung von Webservices) für die Implementierung des Prototypen. Auch

¹⁰Rechtliche Probleme werden bereits über die Betrachtung beim Im- und Export abgedeckt.

die Entscheidung für Facebook und Clearspace als Beispielsysteme für die Implementierung und Anbindung erfüllte die Erwartungen. Die in Abschnitt 6.6.1 auf Seite 75 erarbeiteten Funktionalitäten ließen sich in vollem Umfang implementieren und erwiesen sich als geeignete Wahl, um eine beispielhafte Datenaggregation durchzuführen.

Abschließend wurde die Implementierung mit Hilfe einer Bewertungsmatrix analysiert (vgl. 6.1 auf der vorherigen Seite). In diese Matrix flossen sowohl die Ergebnisse aus den zwei Analyseblöcken (Frameworks und Soziale Netzwerke) sowie aus der Implementierung ein. Die Ergebnisse der Bewertungsmatrix bekommen hierdurch eine hohe Aussagekraft. Das gewählte Vorgehen für die Analyse hat sich somit bewährt und führte zu aussagekräftigen Ergebnissen.

7 ZUSAMMENFASSUNG DER ERGEBNISSE UND AUSBLICK

Nach der Betrachtung der Vorgehensweise sowie der Evaluierung der Implementierung wird in diesem Abschnitt ein abschließendes Fazit gezogen und mögliche weiterführende Fragestellungen im Bereich Sozialer Netzwerke aufgezeigt, die sich im Rahmen der Untersuchung und Implementierung des Prototypen ergeben haben.

7.1 FAZIT

Im Rahmen dieser Arbeit wurden die folgenden Ergebnisse erzielt:

1. Übersicht und Kategorisierung von bestehenden Sozialen Netzwerken und Frameworks
2. Entwicklung einer Evaluierungsmatrix für das Themenfeld Soziale Netzwerke
3. Analyse anhand der Evaluierungsmatrix von Funktionalitäten, insbesondere der APIs für den Datenaustausch
4. Konzipierung und Implementierung einer beispielhaften Daten- und Kontaktaggregation inklusive einer Datenvisualisierung
5. Evaluierung der Lösung hinsichtlich rechtlichen, technischen und Nutzeraspekten

Übersicht und Kategorisierung von bestehenden Sozialen Netzwerken und Frameworks

Unter diesem Punkt wurde im Rahmen eine Recherche eine Marktübersicht über bestehende Soziale Netzwerke und Frameworks erstellt. Die gewonnene Übersicht bildete insbesondere eine solide Grundlage für die Analyse der Netzwerke. Die Sozialen Netzwerke wurden dabei in drei Gruppen (All-in-One Lösung, Create your own Social Network und Tools) unterteilt, wobei die Untergruppe Tools noch einmal in sieben Subkategorien eingeteilt wurde (vgl. Abschnitt 2.5 auf Seite 15). Die Netzwerke für die Analyse der Funktionalitäten wurden auf Grundlage der Kriterien „Größe, Regionaler Einfluss, Erweiterungsmöglichkeiten und Inhaltlicher Fokus“, „Bebo, Facebook, Hi5, LinkedIn, MySpace, Orkut, studivZ und Xing“ ausgewählt (vgl. Abschnitt 2.2 auf Seite 17).

Entwicklung einer Evaluierungsmatrix für das Themenfeld Soziale Netzwerke Aus den ausgewählten Sozialen Netzwerken wurden typische Funktionalitäten (Applikationen und APIs, Blog, Datenfreigabe, Fotos, Gästebuch, Gruppen, Nachrichten, Kalender, Profil, Suche und Videos) extrahiert und mit Hilfe einer eigens dafür entwickelten Evaluationsmatrix untersucht (vgl. Abschnitt 3.2 auf Seite 20). Diese Matrix wurde auf ihre Kompatibilität für einen Vergleich der Frameworks überprüft. Hierbei wurden leichte Modifikationen an der Matrix vorgenommen, um im Bereich der Frameworks wichtige Aspekte, wie beispielsweise die Implementierung der Frameworks, analysieren zu können. Die Matrix wurde universell angelegt und kann daher für weitere Evaluationen im Bereich Sozialer Netzwerke verwendet werden.

Analyse anhand der Evaluierungsmatrix von Funktionalitäten, insbesondere der APIs für den Datenaustausch

Auf Grundlage der erstellten Matrix wurde die Evaluierung der Funktionalitäten vorgenommen. Die Darstellung der Ergebnisse erfolgte sowohl in ausformulierter als auch in tabellarischer Form. Für die Sozialen Netzwerke wurde die ausformulierte Fassung nach Funktionalitäten gegliedert, so dass hier ein schneller Überblick über die Besonderheiten der einzelnen Netzwerke gewonnen werden kann (vgl. Abschnitt 5.4 auf Seite 46ff). Die Ergebnisse in tabellenform ermöglichen einen direkten Vergleich zwischen den Funktionalitäten (vgl. Tabelle A.3 auf Seite 105). Es zeigte sich, dass die Funktionen bei fast allen Betreibern von Sozialen Netzwerken deckungsgleich sind. Unterschiede sind insbesondere dort zu beobachten, wo eine inhaltliche Schwerpunktsetzung des Sozialen Netzwerks vorliegt. Alle Funktionsbereiche zusammenfassend betrachtend bietet Facebook die ausgefeiltesten Funktionalitäten (vgl. Abschnitt 3.3 auf Seite 27 und 3.1 auf Seite 28). Dies trifft insbesondere auf die API zu, wenngleich Google mit OpenSocial eine gute alternative Softwarelösung mit Entwicklungspotential anbietet (vgl. Abschnitt 4.2.7 auf Seite 36). Für den Informationsexport mit Hilfe von APIs ließ sich noch kein endgültiges Fazit ziehen, da der Markt in diesem Bereich durch Ankündigungen und geschlossene Betatests momentan zu sehr in Bewegung ist. Facebook Connect und Googles Friend Connect erscheinen als APIs mit Zukunft, auch wenn Windows Live und Yahoo! Address Book API bereits für den reinen Export fortgeschrittene Funktionalitäten anbieten (vgl. Abschnitt 4.3.6 auf Seite 39).

Bei der Evaluierung der Frameworks wurde die ausformulierte Übersicht produktspezifisch aufgebaut (vgl. Abschnitt 5.5.1 auf Seite 48). In der Tabelle wurden die Ergebnisse nach Funktionsbereichen sortiert, um einen direkten Vergleich der Funktionalitäten zu ermöglichen (vgl. Tabelle A.17 auf Seite 123). Die Frameworks „Clearspace Community“, „Lotus Connections“, „Community Server“ und „Ning“ wurden in der Analyse positiv bewertet und für den Einsatz in Produktivumgebungen empfohlen. „Peopleaggregator“ hingegen wies zu viele fehlerhafte Funktionalitäten auf, um gewinnbringend eingesetzt werden zu können (vgl. Abschnitt 5.1 auf Seite 67). Über alle Funktionsbereiche hinweg betrachtet überzeugte Clearspace Community, insbesondere durch die Möglichkeiten der APIs und der Applikationen (vgl. Abschnitt 5.5.2 auf Seite 52).

Konzipierung und Implementierung einer beispielhaften Daten- und Kontakttaggregation inklusive einer Datenvisualisierung

Aufbauend auf den Erkenntnissen der Analyse der Sozialen Netzwerke und Frameworks wurden in einer beispielhaften Daten- und Kontakttaggregation die folgenden Funktionen realisiert (vgl. Abschnitt 6.6.1 auf Seite 75):

- Daten und Kontaktimport aus einer XML Datei im FOAF Format
- Daten und Kontaktimport über eine Webserviceschnittstelle aus einem anderen Sozialen Netzwerk
- Daten und Kontaktexport in eine XML Datei im FOAF Format
- Daten und Kontaktexport über eine Webserviceschnittstelle in ein anderes Soziales Netzwerk
- Datenvisualisierung auf Basis von Google Maps

Als Beispielsysteme wurden Clearspace Community und Facebook verwendet, welche sich wegen der guten Evaluierungsergebnisse als geeignete Wahl erwiesen (vgl. Abschnitt 3.3 auf Seite 27 und 5.6 auf Seite 65). Eingesetzte Technologien waren „REST“, „SOAP“, „FOAF“, „XML“, „PHP“ und „Java“. Alle Funktionalitäten wurden erfolgreich implementiert. Die verwendeten Technologien und Frameworks waren, bis auf geringe Einschränkungen (kein Export der Emailadresse aus Facebook und Probleme bei der Repräsentation von bestimmten Daten im FOAF Format), sehr geeignet für die Implementierung (vgl. Abschnitt 6.6.2 auf Seite 80 und 6.7 auf Seite 84).

Evaluierung der Lösung hinsichtlich rechtlichen, technischen und Nutzeraspekten Für die Auswertung der Implementierung wurde eine Evaluierungsmatrix mit fünf Problembereichen aufgestellt:

- Authentifizierung
- Datenimport
- Datenexport
- Zusammenführung der Daten
- Repräsentation der Daten

Diese Punkte wurden jeweils hinsichtlich technischer und rechtlicher Aspekte sowie aus Nutzersicht betrachtet. Diese Untersuchung wurde sowohl für die konkrete Implementierung, (vgl. Abschnitt 6.7 auf Seite 81ff) als auch unter allgemeinen Gesichtspunkten (vgl. Abschnitt 6.8 auf Seite 85ff) für weitere Datenaggregationen durchgeführt. Zusammenfassend lässt sich sagen, dass bei der Authentifizierung am wenigsten Schwierigkeiten auftraten. Import- und Export der Daten waren ebenfalls ohne große Probleme möglich. Bei der Repräsentation und Zusammenführung von Daten traten aus technischer Sicht die größten Probleme auf. 1:n oder n:1 Beziehungen, aber auch unterschiedliche semantische Bedeutungen von Profifeldern sind als Hauptprobleme zu nennen (vgl. Abschnitt 6.1 auf Seite 89). Dennoch gilt aber, dass die Daten- und Kontakttaggregation gut möglich und realisierbar ist. Empfehlungen für das Vorgehen in den vier Problembereichen wurden im Abschnitt 6.8 auf Seite 85 angeführt, dabei sind insbesondere die technischen als auch die konzeptionellen Empfehlungen von besonderem Interesse.

Abschluss Im Verlauf dieser Arbeit konnte gezeigt werden, dass sich eine Daten- und Kontakttaggregation sowohl konzeptionell als auch praktisch realisieren lässt. Dieses wurde mit Hilfe einer detaillierten Analyse im Umfeld Sozialer Netzwerke und einer prototypischen Implementierung nachgewiesen. Zusätzlich stellt diese Arbeit wertvolle Erkenntnisse für die Verwendung und den Vergleich von Sozialen Netzwerken bereit und bietet einen Leitfaden für die Planung und Implementierung von Datenaggregationen.

7.2 AUSBLICK

Aus dieser Arbeit ergeben sich für den Bereich Datenaggregation im Umfeld Sozialer Netzwerke eine Reihe möglicher weiterführender Fragen.

Obwohl die Zusammenführung von Daten weitestgehend problemlos gelang, wurden einige weitergehend Fragestellungen aufgeworfen. In Zukunft wird im Bereich der Zusammenführung von Daten von Interesse sein, ein Konzept für die 1:n und n:1 Beziehungen zu erarbeiten. Auch Ansätze, weitere semantische Informationen - wie beispielsweise den Wertebereich eines Datums - in das FOAF Format einzubetten, gilt es zu entwickeln. In diesem Zusammenhang wäre auch denkbar, eine auf Soziale Netzwerke und insbesondere die Datenaggregation zugeschnittene FOAF Spezifikation zu erarbeiten.

Um eine universale Daten- und Kontakttaggregation zu realisieren, sollte auch die Entwicklung einer eigenständigen Webapplikation in Betracht gezogen werden. Über eine modulare Aufbauweise mit Plugins kann eine Anbindung an sehr viele Systeme erreicht werden.

Unter Usabilitygesichtspunkten ist eine Untersuchung von anderen Eingabemethoden für die Datenzuordnung interessant. Insbesondere bei der Zuordnung von großen Datenmengen können andere Methoden als die implementierte manuelle Zuordnung nützlich sein.

Wurde in dieser Arbeit die Visualisierung nur als Kontrolle für die erfolgte Datenaggregation verwendet, wäre es durchaus denkbar, die Visualisierung für weitergehende Zwecke zu nutzen, wie beispielsweise eine Anbindung an virtuelle Welten. Hierdurch könnte eine weitere Verwendungsmöglichkeit der Daten für die Nutzer erschlossen werden.

Abseits der technischen Möglichkeiten ist auch eine Untersuchung interessant, die den Zusammenhang zwischen der Qualität und Anzahl der Funktionalitäten einerseits und dem Erfolg der Plattform andererseits betrachtet. Dies kann wichtige Hinweise für die Planung eines eigenen Sozialen Netzwerks liefern.

Zusätzlich konnten einige Trends im Umfeld Sozialer Netzwerke ausgemacht werden, die es sich lohnt in Zukunft zu beobachten. Im Einzelnen sind dies:

- Die Vormachtstellung von Google und Facebook als Anbieter der zwei relevanten APIs (vgl. Abschnitt 3.3 auf Seite 27).
- Die Öffnung von Sozialen Netzwerken gegenüber Entwicklern und deren Applikationen (vgl. Abschnitt 4.2 auf Seite 30).
- Die Möglichkeit mit Hilfe einer Verknüpfung die eigenen Daten aus einem Sozialen Netzwerk auch auf anderen Internetseiten zu verwenden, ohne diese aber exportieren zu können (vgl. Abschnitt 4.3.6 auf Seite 39)
- Die Sättigung des Markts in Bezug auf APIs und die netzwerkübergreifende Verwendung von einer API (vgl. Abschnitt 4.3.6 auf Seite 39)

A ANHANG

A.1 INHALT DER DVD

- **/Diplomarbeit/PDF/**
Die Diplomarbeit als PDF.
- **/Diplomarbeit/TEX/**
Die Diplomarbeit als Latex/Lyx Dokument mit dem Literaturverzeichnis im „bibtex-Format“.
- **/Code/Plugin/**
Der Quellcode des SNAT Plugins.
- **/Code/VM/**
Ein VMware-Image mit Cleaspace Community und dem installierten SNAT Plugin.
- **/Code/VM/Readme.txt**
Die Datei enthält die Anleitung zum Verwenden des Images und die nötigen Passwörter.
- **/Software/Cleaspace/**
Cleaspace Community mit Lizenzschlüssel für die Installation.

A.2 AKTUELLE ENTWICKLUNGEN IM VERLAUF DER ARBEIT

Soweit möglich wurden aktuelle Entwicklungen, die sich erst im Verlauf der Diplomarbeit ereignet haben, berücksichtigt. Auf einige Entwicklungen und ihren Verlauf sei an dieser Stelle noch einmal explizit hingewiesen:

Zu Beginn der Arbeit im Mai 2008 war Facebook Connect gerade angekündigt. Im Oktober 2008 läuft die Betaphase und jeder Entwickler kann auf die Schnittstelle zugreifen, auch wenn für den Produktivbetrieb eine Genehmigung von Facebook erforderlich ist (vgl. 4.3.2 auf Seite 37). Auch bei Googles Friend Connect gab es Fortschritte, wobei hier nur eine geschlossene Beta verfügbar ist. An diesen zwei APIs zeigt sich exemplarisch der Kampf um die Vormachtstellung in diesem Gebiet. Im Mai 2008 ermöglichte es Google, dass der Anwender seine Facebook Kontakte mit seinem Google Account verbinden konnte. Allerdings wurde dies nur wenige Tage später von Facebook unterbunden (vgl. [82]). Im Bereich der Applikationen herrscht weiterhin eine sehr große Dynamik. Standen bei Orkut am 30. Mai 2008 „nur“ 120 Anwendungen zur Verfügung (vgl. 4.2.3 auf Seite 33), stehen am 22.10.2008 über 500 Anwendungen zur Installation bereit. Bei Ning spielt das Thema Applikationen ebenfalls eine große Rolle. Am 21.10.2008 wurde angekündigt, dass eine direkte Manipulation des Quellcodes in Zukunft nicht mehr möglich sein wird (vgl. [14]). Stattdessen kann auf OpenSocial Applikationen zurückgegriffen werden. Zusätzlich wird, entgegen dem Trend, die REST API deaktiviert. Auch Yahoo reagiert auf die aktuellen Entwicklungen und möchte für alle seine Dienste ein zentrales Profil verfügbar machen (vgl. [172]). Ebenso lässt sich an den API- und Frameworkversionen die Entwicklung im Umfeld Sozialer Netzwerke gut ablesen. Im Verlauf der Diplomarbeit wurde OpenSocial von Google um zwei Versionen weiterentwickelt und erhielt u.a. eine REST-API. Auch Clearspace Community ist inzwischen (22.10.2008) in der Version 2.5 (vorher 2.0.5) verfügbar.

Es gibt aber auch Ankündigungen, die nicht umgesetzt wurden. So wurde bis zum heutigen Zeitpunkt (22.10.2008) die OpenSocial API bei StudiVZ nicht implementiert. Dies könnte u.a. einer der Gründe für den Führungswechsel bei StudiVZ sein (vgl. [19]). Auch Data Availability von MySpace hat seit der Ankündigung keine wesentlichen Weiterentwicklungen erfahren (vgl. 4.3.3 auf Seite 38).

A.3 GRUNDLAGEN

A.4 ÜBERSICHT ÜBER SOZIALE NETZWERKE OHNE INHALTLICHEN SCHWERPUNKT

Name	Gründung	Mitglieder	URL	GP/SI	Schwerpunkt	Tool
Asian Avenue	1999	1.400.000 [159]	asianave.com	SZ	nein	A
Bebo	2005	42.000.000 [11]	bebo.com	GP	nein	A
Black Planet	1999	18.000.000 [29]	blackplanet.com	SZ	nein	A
Consumating	2005 - 2008					
Cyworld	2001	20.000.000 [37]	us.cyworld.com	GP	nein	A
Facebook	2004	70.000.000 [44]	facebook.com	GP	nein	A
Flickr	2004		flickr.com	GP	nein	T-F
Friendster	2002	65.000.000 [55]	friendster.com	SZ	nein	A
Hi5	2003	80.000.000 [74]	hi5.com	GP	nein	A
Hyves	2004	5.000.000 [31]	hyves.net	GP	nein	A
lokalisten	2005	1.800.000 [97]	lokalisten.de	GP	nein	A
meinvz.net	2008	< 1.000.000 [145]	meinvz.net	GP	nein	A
Multiply	2004	5.000.000 [115]	multiply.com	GP	nein	A
MiGente	2000	2.000.000 [28]	migente.com	SZ	nein	A
Mixi	2004	10.000.000 [160]	mixi.jp	GP	nein	A
Ning	2005		ning.com	GP	nein	CSN
SecondLife	2003	13.547.628	secondlife.com	GP	nein	A
SixDegrees	1997- 2000		-			
schuelervz.net	2007	3.000.000 [145]	schuelervz.net	SZ	nein	A

Name	Gründung	Mitglieder	URL	GP/SI	Schwerpunkt	Tool
studivZ.net	2005	5.000.000 [145]	studivz.net	SZ	nein	A
tribe	2003	0.500.000 [158]	tribe.net	GG	nein	CSN
Twitter	2006		twitter.com	GP	nein	T-B
Orkut	2004	120.000.000 [167]	orkut.com	GP	nein	A
Piczo	2004	28.000.000	piczo.com	GP	nein	A
Yahoo!360	2005		360.yahoo.com	GP	nein	A
Yahoo! Mash	2007		blog.mash.yahoo.com	GP	nein	A
YouTube	2005		youtube.com	GP	nein	T-V
Windows Live Spaces	2006		spaces.live.com	GP	nein	A

Tab. A.1: Übersicht über Soziale Netzwerke ohne inhaltlichem Schwerpunkt

A.5 ÜBERSICHT ÜBER SOZIALE NETZWERKE MIT INHALTLICHEM SCHWERPUNKT

Name	Gründung	Mitglieder	URL	GP/SI	Schwerpunkt	Tool
Care2	2004	8.000.000 [25]	care2.com	GP	ja	A
Couchsurfing	2003	402.822 [30]	couchsurfing.org	GP	ja	T-T
del.icio.us	2005		del.icio.us	GP	ja	T-L
Dodgeball	2004		dodgeball.com	GP	ja	T-T
Dogster & Catster	2004	500.000 [34]	dogster.com	SZ	ja	A
Last.FM	2003	15.000.000 [99]	last.fm	GP	ja	T-M
Fotolog	2002	13.000.000 [72]	fotolog.com	GP	ja	T-F
Hospitalityclub	2000	53.000 [150]	hospitalityclub.org	SZ	ja	T-T
LinkedIn	2003	14.000.00 [95]	linkedin.com	GP	ja	A
LiveJournal	1999	15.600.000 [96]	livejournal.com	GP	ja	T-B
MyChurch	2006	146.511 [116]	mychurch.org	SZ	ja	A
MySpace	2003	> 200.000.000[144]	myspace.com	GP	ja	A
Xing	2003	4.834.000 [175, 16]	xing.com	GP	ja	A

Tab. A.2: Übersicht über Soziale Netzwerke mit inhaltlichem Schwerpunkt

A.6 FUNKTIONALITÄTEN VON SOZIALEN NETZWERKEN

A.6.1 Accountfunktionalitäten

	Registrierungsinformationen	Login	Hinterlegbare Daten
studiVZ	Vor- und Nachname E-Mail Passwort Geburtstag Geschlecht Hochschule	E-Mail	Hochschulinformationen Adresse IM-Benutzernamen Telefonnummern Beziehungsstatus Interessen Lieblings- (Filme/Bücher...) Jobangaben Lehrveranstaltungen Profilfoto
Facebook	Vor- und Nachname E-Mail Passwort Geburtstag	E-Mail	Geschlecht Religion Politische Richtung Adresse IM-Benutzernamen Beziehungsstatus Lieblings (Filme/Bücher...) Interessen Hochschulinformationen Jobangaben Profilfoto

	Registrierungsinformationen	Login	Hinterlegbare Daten
Xing	Vor- und Nachname E-Mail Passwort Geburtsstag Benutzername	Benutzername	Interessen Jobangaben Hochschulinformationen Sprachen Adresse IM-Benutzernamen
MySpace	Vor- und Nachname Email Passwort Geburtsdatum Geschlecht Anzeigename Land Region Postleitzahl Anzeigesprache	E-Mail	Interessen Jobangaben Adresse Ethnie Größe Sexuelle Orientierung Raucher / Trinker Kinderstatus Hochschulinformationen Religion Jobangaben Profilfoto
Orkut	Vor- und Nachname E-Mail Passwort	E-Mail	Geschlecht Adresse Land Hochschulangaben Ethnie Politische Richtung Raucher / Trinker Religion IM-Benutzernamen Jobangaben Interessen Datingangaben

	Registrierungsinformationen	Login	Hinterlegbare Daten
LinkedIn	Vor- und Nachname E-Mail Passwort Land Postleitzahl Beschäftigungsstatus Firma Position Branche	E-Mail	Hochschulangaben Jobangaben Profilfoto
Bebo	Vor- und Nachname E-Mail Passwort Geschlecht Land Geburtstag	E-Mail (vergibt zusätzlich automatischen Benutzernamen)	IM-Benutzernamen Profilfoto Adresse Interessen
Hi5	Vor- und Nachname E-Mail Passwort Geburtstag Geschlecht	E-Mail	Religion Beziehungsstatus Sprachen Profilfoto Adresse Hochschulinformationen Interessen IM-Benutzernamen

Tab. A.3: Accountfunktionaliten von Sozialen Netzwerken

A.6.2 Applikationen und APIs

	Applikationen möglich		Importmöglichkeiten von Adressen (Email / IM)	
studiVZ	1	Keine Möglichkeit (OpenSocial Mitglied)	1	Keine Möglichkeit
Facebook	4	Eigene Applikationen	4	gmx.de, hotmail, web.de, yahoo.de, gmail, AIM BuddyList, Upload von Textfile
Xing	1	Open Social Mitglied Keine Applikationen	4	Upload von Textfile Alle gängigen Emailkonten
MySpace	5	OpenSocial Mitglied Eigene Applikationen	2	Einige Emailkonten
Orkut	5	OpenSocial Mitglied Eigene Applikationen	4	Download der eigenen Kontakte als CSV Import für einige Emailkonten
LinkedIn	1	OpenSocial Mitglied Keine Applikationen	3	Einige Emailkonten Textdatei Import
Bebo	5	OpenSocial Mitglied Eigene Applikationen	3	Einige Emailkonten MSN IM Import / AIM
Hi5	5	OpenSocial Mitglied Eigene Applikationen	2	Einige Emailkonten
Mittelwert	4		3	

Tab. A.4: API Funktionalitäten von Sozialen Netzwerken

A.6.3 Blog

	Blog führen		Blog abonnieren	
studivZ	0	Nicht möglich	0	Nicht möglich
Facebook	0	Nicht möglich	0	Nicht möglich
Xing	0	Nicht möglich	0	Nicht möglich
MySpace	3	Funktion vorhanden	3	Funktion vorhanden
Orkut	0	Nicht möglich	0	Nicht möglich
LinkedIn	0	Nicht möglich	0	Nicht möglich
Bebo	2	Funktion vorhanden - sehr spartanisch	3	Funktion vorhanden
Hi5	4	(Journal) Funktion vorhanden Fotos einbinden möglich	0	Nicht möglich
Mittelwert	3		3	

Tab. A.5: Blogfunktionalitäten von Sozialen Netzwerken

A.6.4 Datenfreigabe

	Granularität der Einstellungen bezogen auf Elemente	Genauigkeit bei der Einschränkung auf Personengruppen
studivZ	3 Nur grob, z.B. Kontaktdaten zusammengefasst, aber beispielsweise kann die Pinnwand versteckt werden	4 Nur grob (Freunde, Freunde von Freunden und Public, alle an meiner Hochschule)
Facebook	5 Sehr detailliert - für jede einzelne Emailadresse	5 Freunde, Freunde von Freunden, spezifisches Netzwerk, niemand, einige Kontakte, alle außer eine bestimmte Person
Xing	4 Allgemeine Daten nur sehr grob, aber Kontaktdaten für jeden einzelnen Kontakt	4 Direkter Kontakt, Kontakte meiner Kontakte, 3+4 Grad, jeder
MySpace	1 z.B. für Geburtsdatum aber für andere Sachen wieder nicht	2 Sehr grob; (älter als 18 Jahre und Freunde)
Orkut	2 Nur für einige Elemente wie z.B.: Geschlecht	3 Kontakte von Kontakten, alle Kontakte, Netzwerk
LinkedIn	1 Kontaktmöglichkeiten können eingeschränkt werden, aber ansonsten eher keine Möglichkeiten	2 Nur Unterscheidung zwischen Netzwerk und außerhalb des Netzwerks
Bebo	1 Geburtsdatum kann versteckt werden	2 Nicht-Öffentlich und Öffentlich
Hi5	3 Detaillierte Einstellungen möglich	3 Alle, niemand, Freunde
Mittelwert	3	3

Tab. A.6: Privacyfunktionalitäten von Sozialen Netzwerken

A.6.5 Fotos

	Fotos hochladen	Personen markieren	Fotos herunterladen
studiVZ	4 Drehen möglich, Titel vergeben, Reihenfolge ändern	3 Per Mausklick statisches Quadrat vergeben; nur für Freunde	0 Rechts-klick wird blockiert
Facebook	4 Gleiches Applet wie bei studiVZ Albumcover kann gesetzt werden	3 Per Mausklick statisches Quadrat vergeben; beliebige Namen	2 Über Rechtsklick möglich
Xing	0 Nicht möglich	0 Nicht möglich	0 Nicht möglich
MySpace	4 Kein Drehen möglich	4 Dynamisches Quadrat Beliebige Namen	2 Über Rechtsklick möglich
Orkut	2 Jedes Foto muss separat selektiert werden Kein Drehen möglich	0 Nicht möglich	2 Über Rechtsklick möglich
LinkedIn	0 Nicht möglich	0 Nicht möglich	0 Nicht möglich
Bebo	3 gGleicher Uploadmanager wie bei Facebook und studiVZ kein Umsortieren möglich	4 Dynamisches Quadrat Beliebige Namen	2 Über Rechtsklick möglich
Hi5	3 Mehre Fotos möglich Keine Umsortierung, kein Drehen	4 Halb-Dynamisches Quadrat Beliebige Namen	3 Über Rechtsklick möglich Dezidierter Fotolink abrufbar
Mittelwert	3	4	2

Tab. A.7: Fotofunktionalität von Sozialen Netzwerken

A.6.6 Gästebuch

	Möglichkeit der Erstellung		Einbinden von Medien	
studivZ	3	Pinnwand - Einträge von einem selber und von anderen möglich	0	Nicht möglich
Facebook	3	Wall - Einträge von einem selber und anderen möglich	3	Einbettung von Photo, Video über Link möglich
Xing	3	Gästebuch	0	Nicht möglich
MySpace	0	Nicht vorhanden	0	Nicht möglich
Orkut	3	Scrapbook	1	Foto kann eingebunden werden
LinkedIn	0	Nicht vorhanden	0	Nicht möglich
Bebo	4	White Board - auch Zeichnungen möglich	0	Nicht möglich
Hi5	3	Scrapbook - dieses fungiert auch als Art Newsfeed	0	Nicht möglich
Mittelwert	3		2	

Tab. A.8: Pinnwandfunktionalitäten von Sozialen Netzwerken

A.6.7 Gruppen

	Gruppen	Netzwerke	Event zentrierte Gruppen
studiVZ	3 Funktion vorhanden Bild kann zugeordnet werden geschlossene Gruppen möglich	2 Jede Uni ist ein Netzwerk - statische festgelegt	3 Als Vorlesungen
Facebook	3 Funktion vorhanden Sichtbarkeit auf Netzwerke beschränkbar Bild kann hinzugefügt werden	3 Netzwerk ergibt sich vorrangig aus Arbeitsbeziehungen und Abschlussklasse etc. Zusätzlich auch noch regionale Netzwerke	3 Für Termine werden automatisch auch umfangreiche Gruppenfunktionen bereitgestellt
Xing	3 Gruppen können manuell angelegt werden - müssen genehmigt werden Forumsfunktion verfügbar	2 Indirekt über Arbeitgeber oder Uni, aber ohne weitere Funktion	2 Über Termine können Listen eingesehen werden
MySpace	3 Funktion vorhanden Bulletin Board und Fotofunktion verfügbar	0 Nicht direkt implementiert	0 Keine Ausgeprägten Funktionen verfügbar
Orkut	3 Communities mit Forum, Umfragen, Events	0 Nicht implementiert	0 Nicht direkt verfügbar
LinkedIn	0 Gruppenfunktion vorgesehen, aber noch nicht wirklich implementiert	2 Über Schule/Arbeitgeber Netzwerke verfügbar, aber ohne Funktion	0 Nicht verfügbar
Bebo	3 Gruppen mit Photos, White Board, Blog, Forum und Poll	3 Über Schule/Uni verfügbar mit Forum und Wall	0 Nicht verfügbar

Hi5	3 Gruppenfunktion vorhanden Message Board Email Gruppe	2 Netzwerke über Schule etc.	0 Nicht verfügbar
Mittelwert	3	2	3

Tab. A.9: Gruppen- und Netzwerkfunktionalitäten von Sozialen Netzwerken

A.6.8 Kalender

	Events einstellen		Export/Import Möglichkeiten	
studiVZ	0	Nicht möglich	0	Nicht möglich
Facebook	5	Möglich, mit Begrenzungen für Teilnahmen und Möglichkeiten zu zusagen Fotos etc. können zu einem Event hinzugefügt werden	2	Export im ICS Format möglich Kein Import möglich
Xing	5	Verschiedene Arten wie Kalendernotiz, Telefonkonferenz, Privater Termin, Öffentlicher Termin Gästeliste möglich und Möglichkeit Teilnahme zu zusagen	2	Export im ICS Format möglich Kein Import möglich
MySpace	3	Termin erstellen möglich Drei Gruppen für Öffentlichkeit anwählbar	0	Nicht möglich
Orkut	2	Für Netzwerke möglich Sehr wenige Optionen	0	Nicht möglich
LinkedIn	0	Nicht möglich	0	Nicht möglich
Bebo	0	Nicht möglich	0	Nicht möglich
Hi5	0	Nicht möglich	0	Nicht möglich
Mittelwert	4		2	

Tab. A.10: Kalenderfunktionalitäten von Sozialen Netzwerken

A.6.9 Nachrichten

	Nachrichten verschicken		Nachrichten mit Multimedialen Inhalten anreichern	
studiVZ	3	An Personen möglich Posteingang und Chronik Verwaltung	0	Nicht möglich
Facebook	5	An Personen und eigene Adressgruppen Direkt an Emailadressen von Nicht-Mitgliedern	3	Einbindung per Link mit Vorschaubild möglich
Xing	3	Als zahlendes Mitglied möglich Versand an Emailadresse	0	Nicht möglich
MySpace	3	An Personen möglich	0	Nicht möglich
Orkut	4	An Personen und Adressbuchgruppen möglich	0	Nicht möglich
LinkedIn	3	Verschiedene Nachrichtentypen Auch direkt an Emailadresse von Nutzern	0	Nicht möglich
Bebo	3	Freunde und Liste von Freunden	4	Audio kann aufgenommen und direkt mit verschickt werden Video, Fotos und eigene Zeichnungen Link auf Profile
Hi5	3	an Personen und Gruppen möglich	3	Video, Musik und Photo möglich aus der eigenen Sammlung
Mittelwert	3		3	

Tab. A.11: Nachrichtenfunktionalitäten von Sozialen Netzwerken

A.6.10 Profil

	Modifikationsmöglichkeiten Profil		Aktivitätsanzeige	
studiVZ	5	Informationen können verändert werden Art der grafischen Anzeige dagegen nicht	0	Funktion nicht vorhanden
Facebook	5	Grafische Anzeige kann fast beliebig verändert werden	5	Feeds informieren je nach Wunsche darüber was die anderen machen
Xing	0	Keine grafischen Veränderungen möglich	4	Feed informiert über neue Stellenangebote und Aktivitäten von Freunden
MySpace	5	Profilseite kann mit beliebigem HTML verändert werden Grafische Reihenfolge der Elemente kann auch verändert werden	4	Status Update darüber was Kontakte machen Update von den Objekten, die der Nutzer abonniert hat.
Orkut	0	Keine grafischen Veränderungen möglich	2	Feed mit Fotokomentaren
LinkedIn	0	Keine grafischen Veränderungen möglich	2	Feeds für Forum und Jobs - weitere Module können hinzugefügt werden
Bebo	4	Skins können hinzugefügt werden Grafische Veränderung der Module	3	Änderungsnachrichten über eigene und fremde Aktivitäten
Hi5	3	Skins, CSS/HTML Customizing	3	Aktuelle Nachrichten über Freunde und einen selber
Mittelwert	4		3	

Tab. A.12: Profilfunktionen von Sozialen Netzwerken

A.6.11 Suche

Funktionen der Suche	
studiVZ	3 Personensuche Gruppensuche „Super Suche“ mit umfangreichen Suchoptionen
Facebook	4 Personen Netzwerke Pages Gruppen Events (Alles über ein Suchfeld erreichbar) Umfangreiche Optionen um Suche einzuschränken
Xing	4 Mitglieder Firma Umfangreiche Suchoptionen Vordefinierte Suchen Umfangreiche Optionen um Suche einzuschränken
MySpace	3 Personen Schule Einige Optionen um Suche einzuschränken
Orkut	3 Users Communities Foren Umfangreiche Optionen für Personensuche
LinkedIn	3 Leute Jobs Antworten Einige Optionen für Personensuche
Bebo	2 People Musik Applikationen Groups Keine Optionen fürs Einschränken der Suche
Hi5	3 People Gruppen Videos Sehr wenige Optionen fürs Einschränken der Suche
Mittelwert	3

Tab. A.13: Suchfunktion von Sozialen Netzwerken

A.6.12 Tagging

Welche Arten von Informationen können getaggt werden?		
studiVZ	3	Fotos
Facebook	4	Fotos Videos Notizen
Xing	2	Kontakte
MySpace	3	Fotos Videos
Orkut	3	Fotos
LinkedIn	3	Personenbeschreibungen Jobbeschreibungen
Bebo	3	Fotos
Hi5	3	Fotos
Mittelwert	3	

Tab. A.14: Taggingfunktionalitäten von Sozialen Netzwerken

A.6.13 Videos

	Videos hoch laden		Videos einbinden		Videos herunterladen	
studiVZ	0	Nicht möglich	0	Nicht möglich	0	Nicht möglich
Facebook	0	Nicht möglich		Per Link mit Vorschaubild z.B. bei Nachrichten und Wall	0	Nicht möglich
Xing	0	Nicht möglich	0	Nicht möglich	0	Nicht möglich
MySpace	4	Möglich Sehr ausgeprägtes eigenes Videoangebot	0	Nicht möglich	2	Nicht möglich, aber Video URL wird direkt angegeben
Orkut	0	Nicht möglich	4	Für Google Video + YouTube möglich	0	Nicht möglich
LinkedIn	0	Nicht möglich	0	Nicht möglich	0	Nicht möglich
Bebo	3	Über VideoEgg möglich	3	Von YouTube möglich	0	Nicht möglich
Hi5	0	Wird deaktiviert (bisher auch über VideoEgg realisiert)	0	Nicht möglich	0	Nicht möglich
Mittelwert	4		4		2	

Tab. A.15: Videofunktionalitäten von Sozialen Netzwerken

A.7 FRAMEWORKS FÜR SOZIALE NETZWERKE

A.7.1 Übersicht über Frameworks

Name	Grund für Ablehnung	URI
Acquia	Eher CMS	acquia.com
Affinity Circles	Branding kann nicht entfernt werden	www.affinitycircles.com
Alfresco	Eher ECM	www.alfresco.com
AlstraSoft	Kein Tagging, nur eine Sprache	www.alstrasoft.com
Awareness	Hostinglösung	www.awarenessnetworks.com
Blogtronix	.NET	www.blogtronix.com
BoonEx	Support nur über expertzzz.com	www.boonex.com
Cclonesumating	Code lange nicht aktualisiert	code.google.com/p/clonesumating/
Community Engine	Sehr kleine Lösung	missingmethod.brunobornszstein.com/projects/community_engine/
Converdge	Hostinglösung	www.converdge.com
Crowd Factory	Sehr kleine Lösung	www.crowdfactory.com
Dzoic Handshake	Zu kleine Lösung	www.dzoic.com
ektron	Eher CMS Lösung	www.ektron.com
Elgg	Neue Version nicht verfügbar	www.elgg.org
GroupSwim	Hostinglösung	www.groupswim.com
Golightly	Hostinglösung	www.golightly.com
Igloosoftware	Eher Kollaboration	www.igloosoftware.com
Insoshi Portal	Ruby on Rails; sehr klein	portal.insoshi.com
Inquire	Eher Forum Add On	www.inquire.com
introNetworks	Copyright auf jeder Seite	www.intronetworks.com
iScripts SocialWare	Normale Community Lösung	www.iscripts.com/socialware

KickApss	Hostinglösung	www.kickapps.com
Kwiqq	Kein Blogging, keine Wikis	www.kwiqq.com
Leverage Software	Hostinglösung	www.leveragesoftware.com
LiquidPlanner	Hauptsächlich Projekt Management	liquidplanner.com
Lithium Technologies	Hostinglösung und Branding	www.lithium.com
LiveWorld	Hostinglösung	www.liveworld.com
LovdByLess	Ruby on Rails; sehr klein	lovdbyless.com
mzinga	Eher Community als SocialNetwork	www.mzinga.com
MyWorklight	Hängt sich eher an andere Web 2.0 Dienste dran	www.myworklight.com
Neighborhood America	Hostinglösung und Branding	www.neighborhoodamerica.com
nNGenera	Eher Kollaboration	www.ngenera.com
Omnifuse	Sehr klein, .NET	www.omnifuse.com
OneSite	Hostinglösung	www.onesite.com
OpenLink Data Spaces (ODS)	Kein einheitliches Framework	virtuoso.openlinksw.com/wiki/main/Main/Ods
PHPizabi	Eher CMS	www.phpizabi.net
PixPulse	Hostinglösung	www.pixpulseteam.com
Pringo Networks	Hostinglösung	www.pringo.com
Prospero Technologies	Hostinglösung	www.prospero.com
Sharepoint	Bietet momentan nur sehr wenige Social Funktionen	www.microsoft.com/sharePoint/default.msp
SelectMinds	Hostinglösung	www.selectminds.com
Small World Labs	Hostinglösung	www.smallworldlabs.com
Sparte Social Network	Keine Testinstallation möglich; nur sehr wage Produktbeschreibung	www.spartasocialnetworks.com
Social Platform	Hostinglösung	www.socialplatform.com
SocialEngine	PHP; eher kleine Lösung	www.socialengine.net

SocialText	Hostinglösung, aber sehr mächtig	www.socialtext.com
Spigit	Eher Ideenmanagement	www.spigit.com
SuiteTwo	Binden eher andere Produkte mit ein	www.spikesource.com/suitedwo/
TheSchoolHall	Nicht erreichbar	www.theschoolhall.com
Trampolinesystems	Eher Kollaboraion - nicht wirklich Community	www.trampolinesystems.com
VMIX Media	Hostinglösung	www.vmix.com
Web Crossing	Hostinglösung	www.webcrossing.com
Web Scribble Solutions	Hostinglösung, keine Statusnachrichten	www.webscribble.com
Webligo	Keine Belastungstest, kein Tagging	www.webligo.com

Tab. A.16: Auflistung von Frameworks für Soziale Netzwerke, die nicht untersucht wurden

A.8 VERGLEICH UND BEWERTUNG DER FUNKTIONALITÄTEN

A.8.1 Accountfunktionalitäten

	People Aggregator	Clearspace Community	Ning	Community Server	Lotus Connections
Registrierungs- informationen	Login Name Vor- und Nachname Passwort E-Mail Foto	Login Name Vor- und Nachname Passwort E-Mail	Verwendung der Ning ID Name Geburtstag E-Mail Passwort	Benutzername Passwort E-Mail Zeitzone	Vor- und Nachname Emailadresse Firma
Login	Login Name sxore.com openID flickr	Login Name LDAP Active Directory	NingID (E-Mail)	OpenID Login Name Windows AD, ASP.NET Forms, Cookies	LDAP Email Single-Sign-on mit Websphere

	People Aggregator	Clearspace Community	Ning	Community Server	Lotus Connections
Hinterlegbare Daten	Geschlecht Geburtstag Adressdaten Usertags Ethnie Geschäftliche Daten Ausbildungsdaten	Mitgliedsdatum Adressdaten Erfahrung Biographie Foto Beruf	Geschlecht Ort Land Weitere freie Felder anlegbar	Name Ort, Geburtstag Geschlecht Webseite Blogadresse Galerie Sprache Öffentliche Emailadresse Lieblings Beiträge Gute RSS Feeds	Gebäude Stockwerk Büro Bürotelefonnummer IP-Telefonnummer Handynummer Pagernummer Faxnummer Alternative E-Mail-Adresse Link zum Blog Jobbezeichnung Assistent Über mich Hintergrund Aussprache
Reputation	Keine speziellen Funktionalitäten	Status Punkte anhand von Aktivität Nutzer können auch manuell auf einen bestimmten Status gesetzt werden	Nutzer können hervorgehoben werden	Verschiedene automatische Erfassungsmöglichkeiten	Hierarchie kann abgebildet werden, aber keine spezielle Funktionalität

Tab. A.17: Accountfunktionalitäten

A.8.2 Applikationen und APIs

	People Aggregator	Clearspace Community	Ning	Community Server	Lotus Connections
API (Import)	3 RSS Widgets FOAF XFN (XHTML Friends Network) (nächst Version)	5 RSS Widgets Blogimport von Movable Type, TypePad, Wordpress und Roller Anbindung an NNTP Newsgruppe Import von Emailpostfachdatei	4 RSS Widgets Flicker Fotos	2 Share Membership	0
API (Export)	3 RSS (RSD) Feed für Nutzerseiten	5 RSS Export für Alle Inhalte, Blog, Dokumente, Diskussionen vcf Card von Personen	4 RSS für Inhalte; Netzwerkinhalt kann exportiert werden (als Admin); Nutzerliste kann als CSV exportiert werden	4 RSS für Aktivitäten, Blogs, Ankündigungen Kommentare, Forum	4 RSS für Aktivitäten, Blogs, Kommentare, Forum
Widgets / Applikationen	3 Implementierung eigener API Methoden möglich; erstellen von Blockmodulen ist möglich; Web Services APIs; PHP Methoden	5 Widgets (HTML, CSS, JavaScript); Umfangreiche Web Services Client (SOAP; REST; XML_RPC) und Core API	5 PHP, REST XML, and REST JSON APIs; Ning Web Widget Framework ; Ning Javascript APIs	3 Community Server Module; REST API; Umfangreiche Theme API	3 Anbindung an Sametime, Notes, Microsoft Office, OutlookExplorer; Widgets auf eigener Seite können angezeigt werden

	People Aggregator	Clearspace Community	Ning	Community Server	Lotus Connections
Import-möglichkeiten von Adressen	4 hCard Facebook (Beta) AIM (Beta) Flickr (Beta)	0 Keine Möglichkeiten	4 AOL, GMail, Hotmail, MSN, Yahoo, Outlook, CSV	0 Keine Möglichkeiten	0 Keine Möglichkeiten

Tab. A.18: Applikationen und APIs

A.8.3 Blog

	People Aggregator	Clearspace Community	Ning	Community Server	Lotus Connections
Blog führen	3 TinyMCE als Editor Anbindung an Gruppenblogs	5 Möglich, RichText, Plaintext, Vorschau, Bilder können eingefügt werden; Einstellungen für Trackbacks und Kommentare; Ansicht auch als PDF	4 Möglich, RichText Editor, Dateien und Bilder können angefügt werden; Privacyeinstellungen für Blog	3 Möglich, aber werden vom Admin angelegt Blogs können in Gruppen zusammengefasst werden	5 Möglich für jeden Benutzer; Gruppenblogs möglich; Rich-Content-Editor mit Dateianhängen
Blog abonnieren	3 Ja (RSS)	4 Ja (RSS); Blog und Kommentare; Versand per Email;	3 Keine Möglichkeit (nur bei öffentlichen Netzwerken RSS Möglichkeit)	4 Ja (RSS)	4 Per (RSS/Atom); Kommentare als auch für Einträge

Tab. A.19: Blog

A.8.4 Datenfreigabe

	People Aggregator	Clearspace Community	Ning	Community Server	Lotus Connections
Granularität der Einstellungen auf Elementebene	4 Privacy-einstellungen sind für jedes Datum separat möglich	4 Nur für Email und Name möglich; allerdings vom Admin noch weitergehend konfigurierbar	4 Für jedes Feld kann die Sichtbarkeit vom Admin festgelegt werden; Für Fotos, Videos und Blogbeiträge können generelle Einstellungen gemacht werden	1 Anzeige in der Mitgliederliste Emailadresse verhindern	0 Keine Möglichkeiten
Granularität der Einstellungen auf Personenebene	3 Vier Personengruppen können unterschieden werden (Nutzer selber, Freunde, jeder, Familie)	0 Keine Möglichkeiten	3 Jeder, Freunde, nur ich	0 Keine Möglichkeiten	0 Keine Möglichkeiten
Security	2 CAPTCHA für die Registrierung;	4 CAPTCHA für Spampotabwehr; SSL für SOAP, REST, und XMLRPC	4 CAPTCHA für Spamabwehr; REST API ist SSL verschlüsselt	4 RSS über SSL; IP Aufzeichnung	4 SSL Zugriff auf REST

Tab. A.20: Datenfreigabe & Privacy

A.8.5 Gruppen (Communities)

	People Aggregator	Clearspace Community	Ning	Community Server	Lotus Connections
Gruppen	4 Verschiedene Gruppenarten; Funktionalitäten: Forum, Fotos, Videos, Termine, Themes, Widget, Mailingliste, und Playlist	3 Über Statusrollen könnten Gruppen dargestellt werden; Alternativ über die verschachtelten Communities	4 Gruppen können erstellt werden; Funktionalitäten: Forum, TextBox, RSS Reader, Privat ; Widgets können angezeigt werden	4 Gruppen mit Blog, Diskussionen, Medien	4 Öffentliche, private und auf Einladung; Funktionalitäten: Foren, Lesezeichen, Feeds Mail an Community
Netzwerke	4 Bilden das komplette System als Subdomain ab; Vollständige Funktionalität kann geklont werden; Netzwerk kann Öffentlich oder Privat sein	5 Communities können beliebig verschachtelt werden; Funktionen: Diskussionen, Wikis, Blog, Community Blog	0 Keine Möglichkeit	0 Keine Möglichkeiten	0 Keine Möglichkeit
Event zentrierte Gruppen	0 Keine Möglichkeit	0 keine Möglichkeit	2 Gästeliste	0 Keine Möglichkeit	4 Nutzer können zu Activities hinzugefügt werden

Tab. A.21: Datenfreigabe & Privacy

A.8.6 Implementierung

	People Aggregator	Clearspace Community	Ning	Community Server	Lotus Connections
Anforderungen	XSLT Support PHP 5.1.6 DOM Support Editierbare PHP.INI ModRewrite Unix Server Mysql (vgl. [22])	mind. 2 Dual Core CPUs (at least 2 GHz) + 2GB; 1GB Java heap; Unterstützte Datenbanksysteme (MySQL, Oracle, PostgreSQL, DB2, SQL Server); Server (Apache, Tomcat, JBoss, BEA, WebLogic Server, IBM, WebSphere, Application Server, Oracle Application Server (vgl. [87]))	SaaS, deshalb keine Anforderungen	Windows Server 2003, Microsoft Windows Server 2000 Microsoft Internet Information Services (IIS) Microsoft .NET 2.0; Datenbank (Microsoft SQL Server 2005 64-bit o Windows Server 2003 R2 64-bit, Microsoft SQL Server 2000 (vgl. [148]))	Zwei CPUs/per server, 2.6 GHz CPU; 4 GB RAM; Red Hat Enterprise Linux ES, Microsoft Windows Server 2003 Standard Edition (32-bit) (vgl. [76])
Hosting	Auf eigenem Server als SaaS Lösung	Auf eigenem Server oder als SaaS Lösung.	SaaS	Eigener Server	Eigener Server

	People Aggregator	Clearspace Community	Ning	Community Server	Lotus Connections
Lizenz & Kosten	Bis 250 Mitglieder (\$2,500.00) 50.000 - 100.000 Mitglieder (\$20,000) für den Nichtkommerziellen Gebrauch ist die Nutzung kostenlos	\$49,950.00 für die erste CPU - jede weitere \$19,500.00	Entfernen des Brandings (\$7.95/Jahr) Anzeigenkontrolle (\$19.95/Jahr) Eigener Domain (\$4.95/Jahr)	\$50,000/Jahr für den ersten Prozessor \$20,000/Jahr für weitere Prozessoren.	
Dokumentation	Wiki Dokumentation ist unvollständig	Die Java API und Web Services sind über eine vollständige Doc sehr gut zugänglich. Einführende Beispiele für Widgets Eigene Community für Entwickler	Community für Netzwerkowner + Developer Ausführliche Dokumentation mit einigen Beispielen Tutorials (auch für ohne Code)	Sehr ausführliche Dokumentation für jeden Teilbereich	Sehr ausführliche Dokumentation

Tab. A.22: Implementierung

A.8.7 Inhalte (Dokumente, Fotos, Videos)

	People Aggregator	Clearspace Community	Ning	Community Server	Lotus Connections
Import / Export von Inhalten	0 Keine Möglichkeit	3 Blogbeiträge	0 Keine Möglichkeit	0 Keine Möglichkeit	3 Bookmarks
Versioning / Backup	0 Keine Möglichkeit	4 Versionisierung für Dokumente	0 Keine Möglichkeit	0 Keine Möglichkeit	0 Keine Möglichkeit
Inhalts-Workflow	0 Keine Möglichkeit	4 Person für Genehmigung kann festgelegt werden	4 Beschränkung für Fotos und Videos	2 Nur pauschale Erlaubnis für Dateiupload beim eigenem Profil	2 für Blogeinträge möglich
Verschiedene Arten von Inhalten	3 Textbeiträge, Video, Audio, Forumsbeiträge	4 Blogbeiträge Dokumente Diskussionen Kommentare Fragen Forumsbeiträge	4 Blogbeiträge, Fotos, Kommentare, Videos, Musik, Widgets, Forumsbeiträge	4 Blogbeiträge, Medien, Foren, Dokumente, Kommentare, Announcements	4 Blogeinträge Bookmarks Forenbeiträge Kommentare Aktivitäten ToDos
Zugriffskontrolle	2 Lizenz kann z.B. für Fotos angegeben werden	4 Dokumente können zum Bearbeiten mit Rechten versehen werden	3 Gruppenerstellung kann beschränkt werden; Musikdownload-beschränkung	5 Ausgeprägtes Rollensystem	4 LDAP Server Anbindung Teamblogs und Community-administratoren möglich

Tab. A.23: Inhalte (Dokumente, Fotos, Videos)

A.8.8 Kalender & Aktionen

	People Aggregator	Clearspace Community	Ning	Community Server	Lotus Connections
Events einstellen	2 Events können mit Tags, Adresse, Beschreibung und Link versehen werden; es stehen drei Eventtypen zur Verfügung: Allgemein, Konferenz und Konzert	0 Keine Möglichkeit (nur in der Enterpriseversion)	4 Events können erstellt werden Events können kommentiert werden	0 Keine Möglichkeit	2 Keine Möglichkeit, aber bedingte Abbildung über Aktivitäten
Export/Import	0 Keine Möglichkeit	0 Keine Möglichkeit	3 Export zu Outlook oder iCal (.ics)	0 Keine Möglichkeit	0 Keine Möglichkeit

Tab. A.24: Kalender & Aktionen

A.8.9 Kollaboration

	People Aggregator	Clearspace Community	Ning	Community Server	Lotus Connections
Inhalte mit anderen zusammen verfassen	0 Nicht möglich	4 Dokumente können zwischen Personengruppen geteilt werden	0 Nicht möglich	0 Nicht möglich	5 Blogs können zusammen verfasst werden; Aktivitäten können zusammen verwaltet werden
Kommentare & Bewertungsmöglichkeiten	4 Inhalte können mit 0-5 Sternchen bewertet oder kommentiert werden.	4 Alle Inhalte können kommentiert werden; Inhalte können mit fünf Sternen bewertet werden	3 Fotos können bewertet werden. Inhalte können kommentiert werden	4 Blogeinträge und Objekte in der Mediengalerie können kommentiert werden; Objekte in der Mediengalerie können ebenfalls mit fünf Sternen bewertet werden	4 Blogeinträge können kommentiert und bewertet werden

Tab. A.25: Kollaboration

A.8.10 Nachrichten

	People Aggregator	Clearspace Community	Ning	Community Server	Lotus Connections
Nachrichten verschicken	4 Zu Kontakten möglich; Konversationsansicht eigene Ordner möglich	4 RSS Feed ; Eigene Ordnerverwaltung; Adressbuch	4 Möglich, auch an externe Emailadressen	3 Nachrichten an Freunde	0 Keine Möglichkeit
Nachrichten mit Multi-medialen Inhalten anreichern	0 Keine Möglichkeit	0 Keine Möglichkeit	0 Keine Möglichkeit	5 Objekte aus der Media Galerie	0 Keine Möglichkeit

Tab. A.26: Nachrichten

A.8.11 Profil

	People Aggregator	Clearspace Community	Ning	Community Server	Lotus Connections
Modifikationsmöglichkeiten Profil	1 Außer über die Privacyeinstellungen können hier keine Modifikationen vorgenommen werden	0 Keine Möglichkeit	5 Blöcke auf der Startseite können verschoben werden; Eigene RSS Feeds können eingebunden werden	0 Keine Möglichkeit	4 Blöcke können verschoben werden; Eigene Widgets eingebunden werden
Aktivitätsanzeige	2 Funktionen für neuste Posts sind verfügbar; Keine Informationen über diese gesamten Aktivitäten	3 Zentrale Historyfunktion, die allerdings nicht nutzerfokussiert, sondern dokumentenfokussiert ist.	4 Auf seiner Homepage wird die Aktivität des einzelnen Nutzers aufgelistet, aber nicht über Kontakte	4 Auf der Profilseite ist jeweils eine Übersicht über die Aktivitäten des Nutzers	4 Update, wo etwas passiert

Tab. A.27: Profil

A.8.12 Suche

	People Aggregator	Clearspace Community	Ning	Community Server	Lotus Connections
Funktionalitäten der Suche	2 Suche mit grundlegender Funktionalität ist vorhanden Nur Einschränkung in Bezug auf das Datum möglich	4 Einfache Suche über Stichwort möglich Kann auf Bereiche (Blogs, Zeit, Communities etc.) eingeschränkt werden	2 Keine Möglichkeiten die Suche einzuschränken; durchsucht alle Inhaltstypen	4 Suche nach Tags, Stichwort, Datumsangaben, Autoren oder mit Beschränkung auf Freunde	5 Einfache Stichwortsuche; Kann auf die Bereiche beschränkt werden; Profile auch direkt nach Feldern durchsuchbar

Tab. A.28: Suche

A.8.13 Tagging

	People Aggregator	Clearspace Community	Ning	Community Server	Lotus Connections
Datentagging	3 Alle Arten von Inhalten können getaggt werden; Allerdings keine intelligente Ergänzung; Kaum Verwendungsmöglichkeiten auf Tags	4 Alle Inhalte (außer Ankündigungen) können getaggt werden; Anzeige von TagClouds; Keine Intelligente Ergänzung	3 Diskussionen, Videos, Blogs und Fotos können getaggt werden; Leute können nicht getaggt werden; Tags können zum Suchen verwendet werden	3 Elemente in der Mediengalerie und Blogeinträge; Auswahl von Tags, die bereits zuvor benutzt wurden; Such nach Tags Anzeige einer Tagcloud	5 Sehr zentrale Komponente; Tagging von Personen, Lesezeichen, Blogeinträge, ToDos,; Anzeige von Tagclouds als Liste oder Wolke; Keine Autoergänzung

Tab. A.29: Tagging

A.9 UNTERSUCHUNG UND BEWERTUNG DER DATENAGGREGATION

A.9.1 Kommunikationsstruktur des Plugins

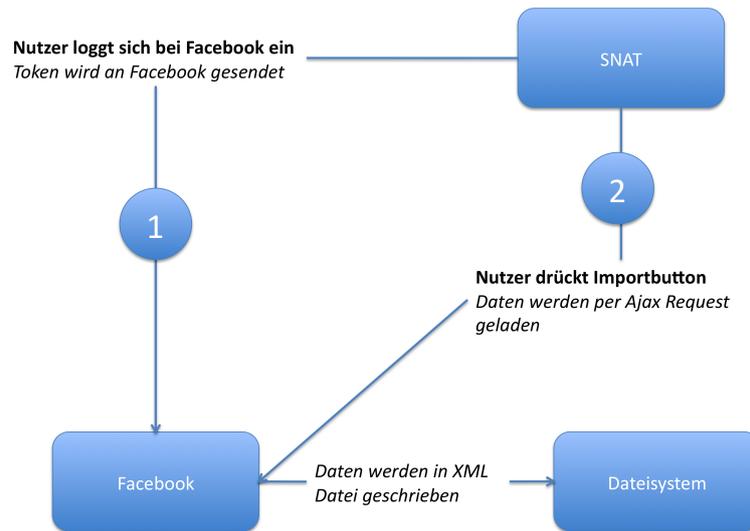


Abb. A.1: Kommunikationsdiagramm zum Import von Daten aus Facebook

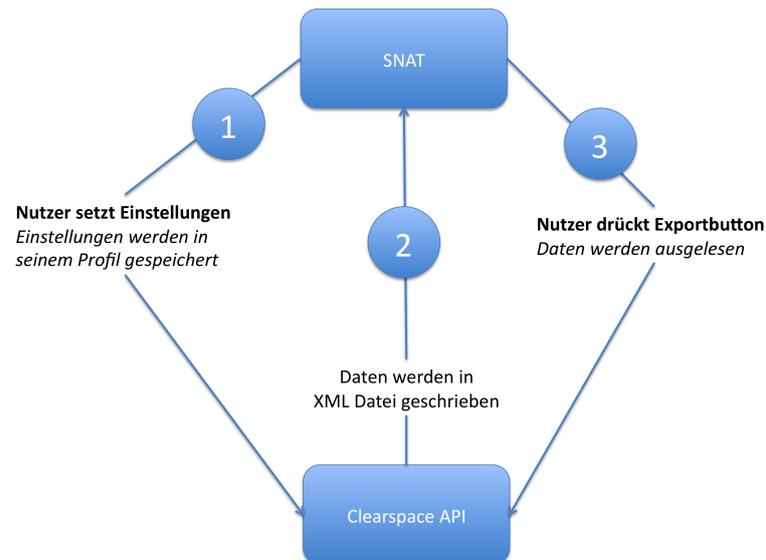


Abb. A.2: Kommunikationsdiagramm zum Export von Daten in eine Datei

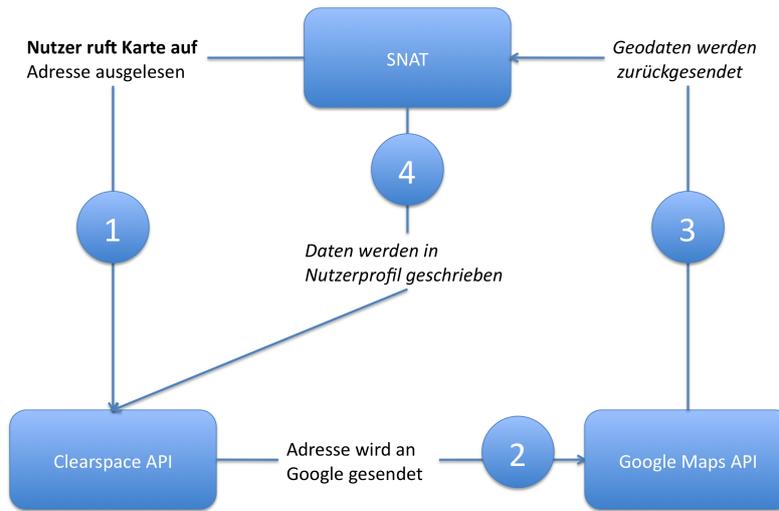


Abb. A.3: Kommunikationsdiagramm zur Geocoding von Adressen

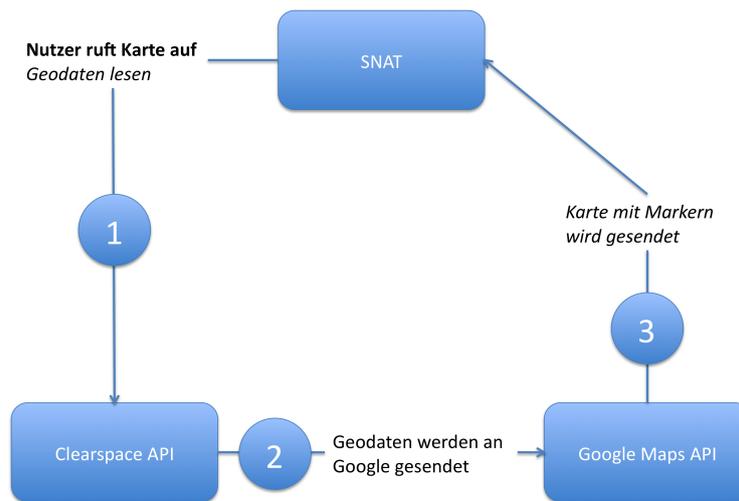


Abb. A.4: Kommunikationsdiagramm zur Kartendarstellung

LITERATUR- UND QUELLENVERZEICHNIS

- [1] AASE, Lee: Social Network Options for Organizations. (2007).
<http://social-media-university-global.org/2007/08/03/social-network-options-for-organizations/>, Abruf: 08.07.2008
- [2] ADAMIC, Lada ; ADAR, Eytan: How to search a social network. In: *Social Networks* Volume 27, Issue 3 (2005), S. 187–203
- [3] ANDREESSEN, Marc: Analyzing the Facebook Platform, three weeks in. (2007).
http://blog.pmarca.com/2007/06/analyzing_the_f.html, Abruf: 19.05.2008
- [4] ARRINGTON, Michael: The Social Network Wars Begin In Earnest: Facebook Bans Google Friend Connect. (2008). <http://www.techcrunch.com/2008/05/15/the-social-network-wars-begin-in-earnest-facebook-bans-google-friend-connect/>, Abruf: 03.06.2008
- [5] BARGH, John ; MCKENNA, Katelyn Y.: The Internet and Social Life. In: *Annu. Rev. Psychol* (2004)
- [6] BASIC, Robert: Social Network selber starten: MyFaible, Mixxt, Ning. (2007).
<http://www.basicthinking.de/blog/2007/12/03/social-network-selber-starten-myfaible-mixxt-ning/>, Abruf: 08.07.2008
- [7] BAYER, Thomas: REST Web Services. (2002).
http://www.thomas-bayer.com/resources/rest/rest_webservices.pdf, Abruf: 15.08.2008
- [8] BEAVER, Doug: Facebook Photos Infrastructure. (2007).
<http://blog.facebook.com/blog.php?post=2406207130>, Abruf: 22.05.2008
- [9] BEBO: Bebo Apps: Open for Business. (2008). <http://developer.bebo.com/blog/index.php/2008/01/10/bebo-apps-open-for-business/>, Abruf: 30.05.2008
- [10] BEBO: Documentation. (2008). <http://developer.bebo.com/documentation.html>, Abruf: 30.05.2008
- [11] BEBO: Leading online drama to film special episode at radio is big weekend featuring scouting for girls, djs and more! (2008).
<http://www.bebo.com/Press.jsp?PressPageId=6615101092>, Abruf: 09.05.2008
- [12] BEHME, Henning: Facebook-API demnächst Open Source? (2008). <http://www.heise.de/newsticker/Facebook-API-demnaechst-Open-Source--/meldung/108577/>, Abruf: 02.06.2008
- [13] BENITEZ, Luis: IBM wins round 1 against Microsoft! (2008). <http://socializeme.blogspot.com/2008/06/ibm-wins-round-1-against-microsoft.html>, Abruf: 09.07.2008
- [14] BIANCHINI, Gina: The Ning Platform: A Few Updates. (2008).
<http://blog.ning.com/2008/10/the-ning-platform-a-few-updates.html>, Abruf: 23.10.2008
- [15] BILLHARDT, Sonja ; HENKE, Ruth ; OEZLUE, Ercin: StudiVZ ist so gähn. 06 (2008). http://www.focus.de/digital/internet/internet-studivz-ist-so-gaehn_aid_236362.html, Abruf: 25.05.2008
- [16] BOYD, Danah ; ELLISON, Nicole: Social Network Sites: Definition, History, and Scholarship. In: *Journal of Computer-Mediated Communication* 13 (2008), S. 210–230.
<http://dx.doi.org/10.1111/j.1083-6101.2007.00393.x>, Abruf: 02.10.2008. – DOI 10.1111/j.1083-6101.2007.00393.x

- [17] BRAUN, Herbert: Google will Zugriff auf soziale Netzwerke erleichtern. (2007).
<http://www.heise.de/newsticker/Google-will-Zugriff-auf-soziale-Netzwerke-erleichtern--/meldung/98270/>,
 Abruf: 02.06.2008
- [18] BRICKLEY, Dan ; MILLER, Libby: FOAF Vocabulary Specification 0.91. (2007).
<http://xmlns.com/foaf/spec/>, Abruf: 13.08.2008
- [19] BRIEGLER, Volker: Holtzbrinck wechselt StudiVZ-Chef aus. (2008). <http://www.heise.de/newsticker/Holtzbrinck-wechselt-StudiVZ-Chef-aus--/meldung/117803>, Abruf: 22.10.2008
- [20] BRIEGLER, Volker: Windows Live tauscht Kontaktdaten mit Social Networks. (2008).
<http://www.heise.de/newsticker/Windows-Live-tauscht-Kontaktdaten-mit-Social-Networks--/meldung/105497/>,
 Abruf: 09.06.2008
- [21] BROADBAND MECHANICS: Description of Internal API. (2008).
http://wiki.peopleaggregator.org/Description_of_API, Abruf: 24.06.2008
- [22] BROADBAND MECHANICS: Installation guide. (2008).
http://wiki.peopleaggregator.org/Installation_guide, Abruf: 08.10.2008
- [23] BROADBAND MECHANICS: Who is Broadband Mechanics? (2008).
[http://www.broadbandmechanics.com/itstories/story\\$data=stories&num=3&sec=2](http://www.broadbandmechanics.com/itstories/story$data=stories&num=3&sec=2),
 Abruf: 24.06.2008
- [24] BROADBAND MECHANICS: WSAPI. (2008). <http://wiki.peopleaggregator.org/WSAPI>,
 Abruf: 24.06.2008
- [25] CARE2: Make a difference. (2008). <http://www.care2.com/aboutus/>, Abruf: 15.06.2008
- [26] CAVERLEE, James ; WEBB, Steve: A Large-Scale Study of MySpace: Observations and Implications for Online Social Networks. (2008).
<http://faculty.cs.tamu.edu/caverlee/pubs/caverlee08alarge.pdf>, Abruf: 02.10.2008
- [27] COMMUNITY SERVER: Showcase. (2008). <http://communityserver.com/showcases/>,
 Abruf: 26.09.2008
- [28] COMMUNITYCONNECT: Community Connect Inc. and Si TV announce partnership to launch channel on migente.com. (2007).
http://www.communityconnect.com/pr_20070228.html, Abruf: 09.05.2008
- [29] COMMUNITYCONNECT: Blackplanet.com is named the largest multicultural niche social networking site of 2007. (2008). http://www.communityconnect.com/pr_20080321.html,
 Abruf: 09.05.2008
- [30] COUCHSURFING: One Couch at a Time. (2008).
http://www.couchsurfing.org/mission_stats.html, Abruf: 09.05.2008
- [31] DE TELEGRAAF: Hyves heeft donderdag 5 miljoen leden. (2007).
http://www.hyves.net/index.php?l1=ut&l2=ab&l3=ns&pressmessage_id=1265907,
 Abruf: 13.05.2008
- [32] DING, Li ; ZHOU, Lina ; FININ, T. ; JOSHI, A.: How the Semantic Web is Being Used: An Analysis of FOAF Documents. In: *System Sciences, 2005. HICSS '05. Proceedings of the 38th Annual Hawaii International Conference on* (2005), Jan., S. 113c–113c.
<http://dx.doi.org/10.1109/HICSS.2005.299>. – DOI 10.1109/HICSS.2005.299. – ISSN 1530–1605
- [33] DODDS, Leigh: XFN. (2003). <http://www.ldodds.com/blog/archives/000105.html>,
 Abruf: 24.07.2008

- [34] DOGSTER: Dogster, Inc., makers of Dogster & Catster. (2008).
<http://www.dogster.com/about/>, Abruf: 13.05.2008
- [35] DUMBILL, Edd: XML Watch: Finding friends with XML and RDF. (2002).
<http://www.ibm.com/developerworks/xml/library/x-foaf.html>, Abruf: 07.07.2008
- [36] DWYER, Catherine ; HILTZ, Starr ; PASSERINI, Katia: Trust and privacy concern within social networking sites: A comparison of Facebook and MySpace. (2007)
- [37] EWERS, Justin: Cyworld: Bigger Than YouTube? (2006).
<http://www.usnews.com/usnews/biztech/articles/061109/9webstars.cyworld.htm>,
 Abruf: 09.05.2008
- [38] FACEBOOK: Developer Terms of Service. (2007).
<http://developer.facebook.com/terms.php>, Abruf: 03.10.2008
- [39] FACEBOOK: Facebook Platform Launches with 65 Developer Partners and Over 85 Applications for Facebook. (2007).
<http://www.new.facebook.com/press/releases.php?p=1319>, Abruf: 25.09.2008
- [40] FACEBOOK: FBMLspec. (2007).
<http://wiki.developers.facebook.com/index.php/FBMLspec>, Abruf: 29.05.2008
- [41] FACEBOOK: Documentation. (2008).
<http://developers.facebook.com/documentation.php>, Abruf: 29.05.2008
- [42] FACEBOOK: A Facebook API client implemented in Java, derived from the poorly-maintained official Facebook client. (2008).
<http://code.google.com/p/facebook-java-api/>, Abruf: 12.09.2008
- [43] FACEBOOK: Facebook Connect Launch Plans. (2008).
http://wiki.developers.facebook.com/index.php/Facebook_Connect_Launch_Plans,
 Abruf: 03.10.2008
- [44] FACEBOOK: Facebook Factsheet. (2008).
<http://www.facebook.com/press/info.php?factsheet>, Abruf: 13.05.2008
- [45] FACEBOOK: Facebook Timeline. (2008).
<http://www.facebook.com/press/info.php?timeline>, Abruf: 19.05.2008
- [46] FACEBOOK: Java. (2008). <http://wiki.developers.facebook.com/index.php/Java>,
 Abruf: 23.09.2008
- [47] FACEBOOK: Welcome to the Facebook Developer Wiki! (2008).
http://wiki.developers.facebook.com/index.php/Main_Page, Abruf: 19.05.2008
- [48] FIELDING, Roy: *Architectural Styles and the Design of Network-based Software Architectures*, University of California, Doctoral dissertation, 2000. http://www.ics.uci.edu/~fielding/pubs/dissertation/fielding_dissertation.pdf,
 Abruf: 15.08.2008
- [49] FIELDING, R.T. ; TAYLOR, Richard: Principled design of the modern Web architecture. In: *Software Engineering, 2000. Proceedings of the 2000 International Conference on* (2000), S. 407–416. <http://dx.doi.org/10.1109/ICSE.2000.870431>, Abruf: 24.09.2008. – DOI 10.1109/ICSE.2000.870431
- [50] FINK, Simon: Datenschutz zwischen Staat und Markt : die 'Safe-Harbor'-Lösung als Resultat einer strategischen Interaktion zwischen der EU, den USA und der IT-Industrie. (2002).
<http://www.ub.uni-konstanz.de/kops/volltexte/2003/1012/pdf/magarbsfink.pdf>,
 Abruf: 25.09.2008

- [51] FISCHER, Martin: Yahoo veröffentlicht Adressbuch-API. (2008). <http://www.heise.de/newsticker/Yahoo-veroeffentlicht-Adressbuch-API--/meldung/109013>, Abruf: 09.06.2008
- [52] FOAF PROJEKT: FOAF Developer Centre. (2008). <http://www.foaf-project.org/web/index.html>, Abruf: 13.08.2008
- [53] FRAPPR: About Us. (2008). <http://www.frappr.com/?a=aboutus>, Abruf: 09.05.2008
- [54] FREEMARKER: Overview. (2008). <http://freemarker.sourceforge.net/>, Abruf: 12.09.2008
- [55] FRIENDSTER: Friendster at a Glance. (2008). http://images.friendster.com/images/Friendster_At_A_Glance_April_2008.pdf, Abruf: 09.05.2008
- [56] GAVIN, Sarah: Bebo joins Googles OpenSocial, reveals plans to launch Facebook-compatible developers platform. (2008). <http://www.bebo.com/Press.jsp?PressPageId=4956881450>, Abruf: 30.05.2008
- [57] GMPG: XFN: Introduction and Examples. (2008). <http://www.gmpg.org/xfn/intro>, Abruf: 24.07.2008
- [58] GOLBECK, Jennifer: The Dynamics of Web-based Social Networks: Membership, Relationships, and Change. In: *Sunbelt XXVIII International Sunbelt Social Network Conference* (2008)
- [59] GOOGLE: Example sites. (2008). <http://www.google.com/friendconnect/home/examples>, Abruf: 09.06.2008
- [60] GOOGLE: Google Maps API. (2008). <http://code.google.com/apis/maps/>, Abruf: 12.09.2008
- [61] GOOGLE: OpenSocial Specification – Implementation Version 0.8. (2008). <http://sites.google.com/a/opensocial.org/opensocial/Technical-Resources/opensocial-specification---implementation-version-08#TOC-People>, Abruf: 02.06.2008
- [62] GOOGLE: People Data API Developer's Guide: Protocol. (2008). http://code.google.com/apis/opensocial/docs/gdata/people/developers_guide_protocol.html, Abruf: 09.06.2008
- [63] GOOGLE: Who's Using It? (2008). <http://code.google.com/apis/opensocial/partners.html>, Abruf: 27.05.2008
- [64] GOTTA, Mike: E2.0 Conference: Social Computing Platforms (IBM & Microsoft). (2008). *E2.0Conference:SocialComputingPlatforms(IBM&Microsoft)*, Abruf: 09.07.2008
- [65] GOTTA, Mike: SharePoint vs. Connections: The Battle Continues. (2008). <http://mikeg.typepad.com/perceptions/2008/05/sharepoint-vs-c.html>, Abruf: 26.09.2008
- [66] GRZONKOWSKI, Slawomir ; GZELLA, Adam ; KRAWCZYK, Henryk ; KRUK, Sebastian ; MOYANO, Francisco J. Martin-Recuerda ; WORONIECKI, Tomasz: D-FOAF - security aspects in distributed user management system. In: *TEHOSS* (2005)
- [67] HENDRICKSON, Mark: 34 More Ways to Build Your Own Social Network. (2007). <http://www.techcrunch.com/2007/08/14/34-more-ways-to-build-your-own-social-network/>, Abruf: 08.07.2008
- [68] HENDRICKSON, Mark: Nine Ways to Build Your Own Social Network. (2007). <http://www.techcrunch.com/2007/07/24/9-ways-to-build-your-own-social-network/>, Abruf: 08.07.2008

- [69] HENDRICKSON, Mark: White Label Social Networking Platforms. (2007).
http://www.techcrunch.com/wp-content/wlsn_comparison_chart.html, Abruf: 08.07.2008
- [70] HENDRICKSON, Mark: White Label Social Networking Solutions, Part II. (2007).
http://www.techcrunch.com/wp-content/white_label_social_networking_solutions_chart2.html, Abruf: 08.07.2008
- [71] HENSEN, Dirk: studiVZ kündigt Beitritt zur Initiative OpenSocial an. (2008).
<http://www.studivz.net/1/press/32>, Abruf: 27.05.2008
- [72] HI-MEDIA PUBLISHING NETWORK: About Fotolog. (2008).
<http://info.fotolog.com/aboutus>, Abruf: 09.05.2008
- [73] HI5: OpenSocial Extensions. (2008).
<http://www.hi5networks.com/platform/wiki/ExtendingOpenSocial>, Abruf: 04.06.2008
- [74] HI5: Press. (2008). <http://www.hi5networks.com/press.html>, Abruf: 13.05.2008
- [75] HOEGG, Roman ; MARTIGNONI, Robert ; MECKEL, Miriam ; STANOEVSKA-SLABEVA, Katarina: Overview of business models for Web 2.0 communities. (2006)
- [76] IBM: Detailed system requirements - Lotus Connections 2.0. (2008).
<http://www-01.ibm.com/support/docview.wss?rs=3265&uid=swg27011798>, Abruf: 10.09.2008
- [77] IBM: Drive your succes. (2008). ftp://ftp.software.ibm.com/software/lotus/lotusweb/product/connections/Lotus_Connections_flyer.pdf, Abruf: 10.09.2008
- [78] IBM: IBM Lotus Connections 2 Information Center. (2008).
http://publib.boulder.ibm.com/infocenter/ltscnct/v2r0/topic/com.ibm.help.developing.connections.doc/c_api_welcome.html, Abruf: 10.09.2008
- [79] IBM: IBM Lotus Connections R2 Reviewer's Guide. (2008). <http://download.boulder.ibm.com/ibmdl/pub/software/dw/lotus/lc2reviewersguide.pdf>, Abruf: 10.09.2008
- [80] JAKOBS, Hans-Jürgen: YouTube - Germany, wir kommen! In: *Süddeutsche Zeitung* (2007)
- [81] JDOM.ORG: JDOM. (2008). <http://www.jdom.org/>, Abruf: 29.09.2008
- [82] JENS - GOOGLE WATCH BLOG: Facebook blockiert Google Friend Connect. (2008). <http://www.googlewatchblog.de/2008/05/16/facebook-blockiert-google-friend-connect/>, Abruf: 22.10.2008
- [83] JIVE SOFTWARE: About Jive Software. (2008). <http://www.jivesoftware.com/company>, Abruf: 26.09.2008
- [84] JIVE SOFTWARE: About Jive Software. (2008). <http://www.jivesoftware.com/company>, Abruf: 03.07.2008
- [85] JIVE SOFTWARE: Example Plugin. (2008).
<http://www.jivesoftware.com/community/docs/DOC-1051>, Abruf: 12.09.2008
- [86] JIVE SOFTWARE: PHP Integration. (2008).
<http://www.jivesoftware.com/community/thread/27520>, Abruf: 09.10.2008
- [87] JIVE SOFTWARE: Recommended Environments for Clearspace Deployments. (2008).
<http://www.jivesoftware.com/builds/docs/clearspace/latest/RecommendedEnvironmentsforClearspaceDeployments.html>, Abruf: 08.10.2008
- [88] JIVE SOFTWARE: See How We've Helped Our Customers. (2008).
<http://www.jivesoftware.com/customers>, Abruf: 26.09.2008

- [89] KHARE, Rohit: Microformats. (2006).
<http://wiki.commerce.net/images/a/a9/20060109225345!CN-TR-06-01.pdf>, Abruf: 23.07.2008
- [90] KING, Ryan: Microformats. (2005).
<http://www.theryanking.com/presentations/2005/webzine/>, Abruf: 23.07.2008
- [91] KNOKE, Felix: Die Gefahren des sozialen Netzes. (2008).
<http://www.spiegel.de/netzwelt/web/0,1518,517584,00.html>, Abruf: 25.09.2008
- [92] KOCH, Michael ; RICHTER, Alexander: Social Software - Status quo und Zukunft. (2007), 48.
<http://www.kooperationssysteme.de/wp-content/uploads/RichterKoch2007.pdf>,
 Abruf: 06.05.2008
- [93] LANQUETIN, Nicolas: Virtual Reality Online Communities. (2006).
http://psbase.com/studies/uad.bak/wdd/ca1030a_essay/ca1030a_essay_rev20.pdf,
 Abruf: 05.05.2008
- [94] LI, J. ; BOLEY, H. ; BHAVSAR, V.C. ; MEI, J.: Expert Finding for eCollaboration Using FOAF with RuleML Rules. In: *Proceedings of the 2006 Conference on eTechnologies. (EDs.) L. Logrippo, H. Mili and A. Salah. Montreal, QC, Canada. NRC 49309.* (2006), S. 53–65
- [95] LINKEDIN: LinkedIn's 14 Million Members Can Now Include Professional Photo Within Their Online Profile. (2008).
http://www.linkedin.com/static?key=press_releases_092707, Abruf: 09.05.2008
- [96] LIVEJOURNAL: LiveJournal. (2008). <http://www.livejournal.com/>, Abruf: 09.05.2008
- [97] LOKALISTEN: In zahlen. (2008). <http://lokalisten.de/web/showHome.do;jsessionid=D44B32B85BCE1F1E08347ED3BA21D52E.s15n1s1>, Abruf: 10.05.2008
- [98] MASNICK, Mike: No Room In MySpace For YouTube. (2005).
http://techdirt.com/articles/20051223/1027248_F.shtml, Abruf: 23.05.2008
- [99] MCCARTHY, Caroline: What does CBS want with Last.fm? (2007). http://www.news.com/What-does-CBS-want-with-Last.fm/2100-1027_3-6187471.html,
 Abruf: 13.05.2008
- [100] MCCARTHY, Caroline: The Scoble scuffle: Facebook, Plaxo at odds over data portability. (2008). [ttp://news.cnet.com/8301-13577_3-9839474-36.html](http://news.cnet.com/8301-13577_3-9839474-36.html), Abruf: 09.06.2008
- [101] MCKINSEY: Harnessing the poewr of informal employee networks. In: *The McKinsey Quarterly* 4 (2007)
- [102] MEIER, Nicola: Sehen, was funktioniert und was nicht. In: *Stern* (2007).
<http://www.stern.de/wirtschaft/arbeit-karriere/karriere/Open-Business-Club-Sehen,/591430.html>, Abruf: 09.05.2008
- [103] MEYER, Eric: XFN and FOAF. (2008). <http://gmpg.org/xfn/and/foaf>, Abruf: 07.07.2008
- [104] MICROFORMATS: Compound Microformat. (2008).
<http://microformats.org/wiki/compound-microformat>, Abruf: 24.09.2008
- [105] MICROFORMATS: Elemental Microformat. (2008).
<http://microformats.org/wiki/elemental-microformat>, Abruf: 24.09.2008
- [106] MICROFORMATS: hCard. (2008). <http://microformats.org/wiki/hcard>, Abruf: 07.07.2008
- [107] MICROFORMATS: hCard and XFN. (2008).
http://microformats.org/wiki/hcard-examples#hCard_and_XFN, Abruf: 24.07.2008
- [108] MICROFORMATS: Microformats Implementations. (2008).
<http://microformats.org/wiki/implementations>, Abruf: 23.07.2008

- [109] MICROFORMATS: Überblick ueber Mikroformate. (2008).
<http://microformats.org/about/>, Abruf: 26.07.2008
- [110] MICROSOFT: Introduction to Windows Live ID. (2008).
<http://msdn.microsoft.com/en-us/library/bb288408.aspx>, Abruf: 24.07.2008
- [111] MISLOVE, Alan ; MARCON, Massimiliano ; GUMMADI, Krishna ; DRUSCHEL, Peter ; BHATTACHARJEE, Bobby: Measurement and analysis of online social networks. (2007), S. 29–42. <http://dx.doi.org/http://doi.acm.org/10.1145/1298306.1298311>, Abruf: 02.10.2008. – DOI <http://doi.acm.org/10.1145/1298306.1298311>. ISBN 978–1–59593–908–1
- [112] MITCHELL-WONG, J. ; KOWALCZYK, R. ; ROSHELOVA, A. ; JOY, B. ; TSAI, H.: OpenSocial: From Social Networks to Social Ecosystem. In: *Digital EcoSystems and Technologies Conference, 2007. DEST '07. Inaugural IEEE-IES (2007)*, Feb., S. 361–366.
<http://dx.doi.org/10.1109/DEST.2007.371999>. – DOI 10.1109/DEST.2007.371999
- [113] MOORE, Lou: hi5 Platform Launch 100(2008). <http://www.hi5networks.com/developer/2008/04/hi5-platform-launch-completed.html>, Abruf: 30.05.2008
- [114] MORIN, Dave: Announcing Facebook Connect. (2008).
<http://developers.facebook.com/news.php?blog=1&story=108>, Abruf: 09.06.2008
- [115] MULTIPLY.COM: Milestones further cement Multiply's position as a leading social media site. (2007). <http://multiply.com/info/press/fivemillion>, Abruf: 13.05.2008
- [116] MYCHURCH: MyChurch. (2008). <http://www.mychurch.org/>, Abruf: 16.05.2008
- [117] MYSPACE: Data Availability REST Resources. (2008).
<http://developer.myspace.com/community/myspace/da.aspx>, Abruf: 03.10.2008
- [118] MYSPACE: MySpace Developer Platform. (2008).
<http://developer.myspace.com/Community/>, Abruf: 29.05.2008
- [119] NICOLE, Kristen: Bebo Opens Developer Platform with Support for Facebook Applications. <http://mashable.com/2007/12/12/bebo-open-application-platform-facebook-support/>, Abruf: 19.05.2008
- [120] NING: Ning Architecture Basics. (2008).
http://developer.ning.com/notes/Ning_Architecture_Basics, Abruf: 09.10.2008
- [121] O'MURCHU, Ina ; BRESLIN, John ; DECKER, Stefan: Online Social and Business Networking Communities. (2004).
<http://www.sti-innsbruck.at/fileadmin/documents/DERI-TR-2004-08-11.pdf>, Abruf: 08.05.2008
- [122] OPENID: Libraries. (2008). <http://wiki.openid.net/Libraries>, Abruf: 26.07.2008
- [123] OPENID: Read the Specifications. (2008). <http://openid.net/developers/specs/>, Abruf: 26.07.2008
- [124] OSTROW, Adam: MySpace Announces "Data Availability"with Yahoo, eBay, Twitter. (2008).
<http://mashable.com/2008/05/08/breaking-myspace-announces-data-availability-with-yahoo-ebay-and-others/>, Abruf: 09.06.2008
- [125] OTTER, Thomas ; DRAKOS, Nikos: Case Study: Dow's Formula for Social Software. In: *Gartner RAS Core Research Note G00156018* (2008).
<http://mediaproducts.gartner.com/reprints/selectminds/156018.html>, Abruf: 19.06.2008

- [126] OWYANG, Jeremiah: List of "White Label" or "Private Label" (Applications you can Rebrand) Social Networking Platforms. (2007). <http://www.web-strategist.com/blog/2007/02/12/list-of-white-label-social-networking-platforms/>, Abruf: 08.07.2008
- [127] OWYANG, Jeremiah: Consolidation Needed in the White Label Social Networking Industry. (2008). <http://www.web-strategist.com/blog/2008/03/03/consolidation-needed-in-the-white-label-social-networking-industry/>, Abruf: 08.07.2008
- [128] OWYANG, Jeremiah: Forrester Underway to Catalog the White Label Social Networking Space. (2008). <http://www.web-strategist.com/blog/2008/05/28/forrester-underway-to-catalog-the-white-label-social-networking-space/>, Abruf: 08.07.2008
- [129] PARAMESWARAN, Manoj ; WHINSTON, Andrew: Social Computing: An Overview. 19 (2007)
- [130] PARSIA, Bijan: XFN vs. FOAF. (2007). <http://clarkparsia.com/weblog/2007/03/14/xfn-vs-foaf/>, Abruf: 07.07.2008
- [131] PAUTASSO, Cesare: SOAP vs. REST - Bringing the Web back into Web Services. (2007). http://www.iks.inf.ethz.ch/education/ss07/ws_soa/slides/SOAPvsREST_ETH.pdf, Abruf: 24.09.2008
- [132] PFEFFERLE, Matthias: Apple Store nutzt hReview. (2007). <http://notizblog.org/2007/10/04/apple-store-nutzt-hreview/>, Abruf: 23.07.2008
- [133] PICZO: Piczo Facts. (2008). <http://www.piczo.com/advertising.html>, Abruf: 13.05.2008
- [134] PRANTL, Heribert: Orwell und Orwellness. In: *Süddeutsche Zeitung* (2008). <http://www.sueddeutsche.de/deutschland/artikel/557/171056/>, Abruf: 25.09.2008
- [135] REISBERGER, Tobias ; REISBERGER, Philip ; SMOLNIK, Stefan: Entwicklung eines funktionalen Klassifikationsschemas für Social-Networking-Systeme. In: MEISSNER, Klaus (Hrsg.) ; ENGELIEN, Martin (Hrsg.) ; Workshop GeNeMe 2008 - Gemeinschaften in Neuen Medien (Veranst.): *Virtuelle Organisation und Neue Medien 2008* Workshop GeNeMe 2008 - Gemeinschaften in Neuen Medien, 2008, S. 43–55
- [136] REKHI, Manu: Apps are live in India. (2008). <http://en.blog.orkut.com/2008/04/apps-are-live-in-india.html>, Abruf: 30.05.2008
- [137] RICHARDS, John: Microsoft Partners with Top Social Networks to Put Users at the Center of their Data. (2008). <http://dev.live.com/blogs/devlive/archive/2008/03/25/237.aspx>, Abruf: 09.06.2008
- [138] ROHN, Julian ; SPETH, Charlotte: Warum die Wirtschaft die Netzwerke liebt. (2008). <http://www.tagesschau.de/wirtschaft/meldung7944.html>, Abruf: 25.09.2008
- [139] ROWE, Matthew: Facebook Foaf Generator. (2008). <http://www.dcs.shef.ac.uk/~mrowe/foafgenerator.html>, Abruf: 13.08.2008
- [140] SCHNEIDER, Professor Dr.: *IT und Computerrecht*. dtv, 2004
- [141] SCHONFELD, Erick: The fbOpen Initiative: Facebook Confirms Plans to Open-Source Its Platform. (2008). <http://www.techcrunch.com/2008/05/27/facebook-confirms-plans-to-open-source-its-platform/>, Abruf: 29.05.2008
- [142] SCHROEDER, Stan: 20 Ways To Aggregate Your Social Networking Profiles. (2007). <http://mashable.com/2007/07/17/social-network-aggregators/>, Abruf: 10.06.2008

- [143] SILVA, Enzo: Open Source Social Network Engines. (2008). <http://tweaklearning.wordpress.com/2008/05/10/open-source-social-network-engines/>, Abruf: 08.07.2008
- [144] STIFTUNG WARENTEST: Daten außer Kontrolle. In: *test* 8 (2008), S. 38–41
- [145] STUVIVZ: Ueber uns. (2008). <http://www.studivz.net/1/press>, Abruf: 09.05.2008
- [146] TECHCRUNCH: White Label Social Networking Solutions, Part II. (2007). http://www.techcrunch.com/wp-content/white_label_social_networking_solutions_chart2.html, Abruf: 25.06.2008
- [147] TECHWEB NETWORKS: Enterprise 2.0 Conference. (2008). <http://enterprise2conf.com/exhibition/demo-pavilion.php>, Abruf: 08.07.2008
- [148] TELLIGENT: System Requirements. (2008). <http://docs.communityserver.com/getting-started/installation/system-requirements/>, Abruf: 08.10.2008
- [149] TELLIGENT: Technology. (2008). <http://communityserver.com/learn-more/technology/>, Abruf: 09.10.2008
- [150] THE HOSPITALITY CLUB: Sample Article. (2005). <http://media.hospitalityclub.org/>, Abruf: 09.05.2008
- [151] TILKOV, Stefan: A Brief Introduction to REST. (2007). <http://www.infoq.com/articles/rest-introduction>, Abruf: 24.09.2008
- [152] TILKOV, Stefan: REST Anti-Patterns. (2008). <http://www.infoq.com/articles/rest-anti-patterns>, Abruf: 24.09.2008
- [153] VENKAT: Choosing a good font size variation algorithm for your tag cloud. (2007). <http://blogs.dekoh.com/dev/2007/10/29/choosing-a-good-font-size-variation-algorithm-for-your-tag-cloud/>, Abruf: 12.09.2008
- [154] VITIELLO, Eric: A module for defining relationships in FOAF. (2002). <http://www.perceive.net/schemas/20020729/relationship/>, Abruf: 07.07.2008
- [155] VIVALOGO: Top 40 Free Downloadable Open Source Social Networking Software. <http://vivalogo.com/v1-resources/open-source-social-networking-software.htm>, Abruf: 08.07.2008
- [156] VORA, Ami: Happy Anniversary (and Facebook Open Platform). (2008). <http://developers.facebook.com/news.php?blog=1&story=117>, Abruf: 03.06.2008
- [157] WATCH, CMS: The Enterprise Social Software Report 2008. (2008)
- [158] WIKIPEDIA: Tribe.net. (2006). <http://en.wikipedia.org/wiki/Tribe.net>, Abruf: 13.05.2008
- [159] WIKIPEDIA: AsianAve. (2007). <http://en.wikipedia.org/wiki/AsianAve>, Abruf: 09.05.2008
- [160] WIKIPEDIA: mixi. (2007). <http://en.wikipedia.org/wiki/Mixi>, Abruf: 13.05.2008
- [161] WIKIPEDIA: Comparison of social networking software. (2008). http://en.wikipedia.org/wiki/Comparison_of_social_networking_software, Abruf: 08.07.2008
- [162] WIKIPEDIA: Consumating. (2008). <http://en.wikipedia.org/wiki/Consumating>, Abruf: 09.05.2008
- [163] WIKIPEDIA: FOAF. (2008). <http://de.wikipedia.org/wiki/FOAF>, Abruf: 13.08.2008

- [164] WIKIPEDIA: FOAF (software). (2008). [http://en.wikipedia.org/wiki/FOAF_\(software\)](http://en.wikipedia.org/wiki/FOAF_(software)), Abruf: 13.08.2008
- [165] WIKIPEDIA: List of social networking websites. (2008). http://en.wikipedia.org/wiki/List_of_social_networking_websites, Abruf: 09.05.2008
- [166] WIKIPEDIA: Mikroformat. (2008). <http://de.wikipedia.org/wiki/Mikroformat>, Abruf: 23.07.2008
- [167] WIKIPEDIA: Orkut. (2008). <http://en.wikipedia.org/wiki/Orkut>, Abruf: 13.05.2008
- [168] WILKENS, Andreas: Facebook lizenziert seine Programmierschnittstelle an andere Sozialnetze. (2007). <http://www.heise.de/newsticker/Facebook-lizenziert-seine-Programmier-schnittstelle-an-andere-Sozialnetze--meldung/100532/>, Abruf: 02.06.2008
- [169] WILKENS, Andreas: Facebook reagiert auf Nutzerproteste gegen zielgruppenspezifische Werbung. (2007). <http://www.heise.de/newsticker/suche/ergebnis?rm=result;q=facebook;url=/newsticker/meldung/99827/;words=Facebook>, Abruf: 12.06.2008
- [170] WILKENS, Andreas: StudiVZ: Kein Verkauf von Nutzerdaten. (2007). <http://www.heise.de/newsticker/suche/ergebnis?rm=result;words=Facebook;q=facebook;url=/newsticker/meldung/100621/>, Abruf: 12.06.2008
- [171] WILKENS, Andreas: MySpace lässt die Entwickler ran. (2008). <http://www.heise.de/newsticker/suche/ergebnis?rm=result;words=MySpace%20myspace;q=myspace;url=/newsticker/meldung/102746/>, Abruf: 29.05.2008
- [172] WILKENS, Andreas: Yahoo will seine Sozialnetzfunktionen ausbauen Meldung vorlesen und MP3-Download. (2008). <http://www.heise.de/newsticker/Yahoo-will-seine-Sozialnetzfunktionen-ausbauen--meldung/117529>, Abruf: 22.10.2008
- [173] WU, Charles: Yahoo! Address Book API Delivered. (2008). http://developer.yahoo.net/blog/archives/2008/06/addressbook_api.html, Abruf: 09.06.2008
- [174] XING AG: OpenSocial: XING schließt sich offenem Programmierstandard an. (2007). http://corporate.xing.com/deutsch/presse/pressemitteilungen/pressmitteilungen-detailansicht/article/pressemitteilungbropensocial-xing-schliesst-sich-offenem-programmierstandard-an/7/896b754d21/?tx_ttnews%5Bpointer%5D=4, Abruf: 27.05.2008
- [175] XING AG: XING releases its 2007 Annual Report. (2007). http://corporate.xing.com/no_cache/english/press/welcome/news-detail/article/pressemitteilungbrxing-legt-geschaeftsbericht-2007-vor/6/d1d0b72b6f/, Abruf: 09.05.2008
- [176] YAHOO: Browser-Based Authentication (BBAuth). (2008). <http://developer.yahoo.com/auth/>, Abruf: 09.06.2008
- [177] YAKHNOV, Dmitry: The Friend of a Friend. (2008). <http://www.new.facebook.com/apps/application.php?id=6135226994>, Abruf: 13.08.2008

ABKÜRZUNGSVERZEICHNIS

A	All-in-One-Lösung
AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
ASP	Active Server Pages
ATOM	Atom Syndication Format
Blogs	Eine Webseite, die im Stil eines Tagebuchs geführt wird
CAPTCHA	Completely Automated Public Turing test to tell Computers and Humans Apart
CMS	Content Management System
CPAL	Common Public Attribution License
CSN	Create your own Social Network
CSV	Comma-separated values
DOM	Document Object Model
FAQ	Frequently Asked Question
FBJS	Facebook JavaScript
FBML	Facebook Markup Language
FOAF	Friend of a Friend
FQL	Facebook Query Language
Gadgets	Kleines Stück Software, welches eine sehr spezifische Aufgabe erledigt
GP	General Public
hCard	(HTML vCard) ist ein Mikroformat um Kontaktdaten im XML Format zu publizieren
HTML	Hyper Text Markup Language
HTTP	Hypertext Transfer Protocol
ICS	Dateiendung für iCalendar Dateien
IM	Instant Messaging
JSON	JavaScript Object Notation
LDAP	Lightweight Directory Access Protocol
MPL	Mozilla Public License
Multi-Framework	Ein System bei dem auf einer Plattform mehrere Netzwerke aufgesetzt werden können
MVC	Model-View-Controller
NNTP	Network News Transfer Protocol
OpenSocial	Framework von Google um Applikationen für Soziale Netzwerke zu erstellen
P2P	Peer-to-Peer
PHP	Hypertext Preprocessor

RDF	Resource Description Framework
REST	Representational State Transfer
RESTful	Bezeichnung aus der Praxis für Representational State Transfer
RSD	Really Simple Discovery
SaaS	Software as a Service
SFTP	SSH File Transfer Protocol
SNAT	Social Network Aggregation Tool
Snippet	Ein kleines Stück Code, welches die Integration einer bestimmten Funktion in anderen Webseiten erlaubt
SNML	Social Network Markup Language
SNQL	Social Network Query Language
SOAP	Simple Object Access Protocol
SQL	Structured Query Language
SSO	Single Sign-On
SZ	Spezielle Zielgruppe
T	Tool
TB	Blog
TF	Fotos
TL	Lesezeichen
TM	Musik
TR	Reise/Unterkunft
TT	Treffen/Meeting
TV	Video
Tagcloud	Tags werden dabei in verschiedenen Größen angezeigt (je nach Häufigkeit)
u.a.	unter anderem
URI	Uniform Resource Identifier
usw.	und so weiter
vcf	Elektronische Visitenkarte
WebDAV	Web-based Distributed Authoring and Versioning
White Label Network	Andere Bezeichnung für Frameworks, mit denen ein eigenes Soziales Netzwerke aufgesetzt werden kann.
Widgets	A component of a graphical user interface with which a user interacts
Wikis	Eine Webseite, deren Inhalte von den Benutzern gelesen und verändert werden können
WWF	Ning Web Widgets
XFN	XHTML Friends Network
XML	Extensible Markup Language
XMLRPC	XML-RPC (Extensible Markup Language Remote Procedure Call) ist eine Definition zum Methodenaufruf (oder auch Funktionsaufruf) durch verteilte Systeme.
XSLT	Extensible Stylesheet Language Transformations
z.B.	zum Beispiel

ABBILDUNGSVERZEICHNIS

2.1	Gründung von Sozialen Netzwerken (vgl. [16])	7
2.2	Überblick über Mikroformate (vgl. [109])	10
2.3	Funktionsweise OpenID	15
5.1	Übersicht über die Ning API (vgl. [120])	56
5.2	Übersicht über den Ning Content Store (vgl. [120])	56
5.3	Community Server Architektur (vgl. [149])	60
5.4	Beispielhomepage für einen Benutzer in Lotus Connections (vgl. [77])	63
5.5	Struktur von Lotus Connections (vgl. [79])	64
6.1	Kommunikationsdiagramm zum Import von Daten aus einer Datei	81
A.1	Kommunikationsdiagramm zum Import von Daten aus Facebook	138
A.2	Kommunikationsdiagramm zum Export von Daten in eine Datei	138
A.3	Kommunikationsdiagramm zur Geocoding von Adressen	139
A.4	Kommunikationsdiagramm zur Kartendarstellung	139

TABELLENVERZEICHNIS

2.1	Häufigkeit der Verwendung von FOAF Eigenschaften pro FOAF Datei (vgl. [32]) . . .	13
2.2	Übersicht über engere Auswahl von Sozialen Netzwerken mit Kriterien	17
3.1	Übersicht über die Analyses der Funktionalitäten von Sozialen Netzwerken	28
5.1	Evaluierungsmatrix der Frameworks	67
6.1	Bewertungsmatrix für eine Datenaggregation	89
A.1	Übersicht über Soziale Netzwerke ohne inhaltlichem Schwerpunkt	101
A.2	Übersicht über Soziale Netzwerke mit inhaltlichem Schwerpunkt	102
A.3	Accountfunktionaliten von Sozialen Netzwerken	105
A.4	API Funktionalitäten von Sozialen Netzwerken	106
A.5	Blogfunktionalitäten von Sozialen Netzwerken	107
A.6	Privacyfunktionalitäten von Sozialen Netzwerken	108
A.7	Fotofunktionalität von Sozialen Netzwerken	109
A.8	Pinwandfunktionalitäten von Sozialen Netzwerken	110
A.9	Gruppen- und Netzwerkfunktionalitäten von Sozialen Netzwerken	112
A.10	Kalenderfunktionalitäten von Sozialen Netzwerken	113
A.11	Nachrichtenfunktionalitäten von Sozialen Netzwerken	114
A.12	Profilfunktionen von Sozialen Netzwerken	115

A.13 Suchfunktion von Sozialen Netzwerken	116
A.14 Taggingfunktionalitäten von Sozialen Netzwerken	117
A.15 Videofunktionalitäten von Sozialen Netzwerken	118
A.16 Auflistung von Frameworks für Soziale Netzwerke, die nicht untersucht wurden . .	121
A.17 Accountfunktionalitäten	123
A.18 Applikationen und APIs	125
A.19 Blog	126
A.20 Datenfreigabe & Privacy	127
A.21 Datenfreigabe & Privacy	128
A.22 Implementierung	130
A.23 Inhalte (Dokumente, Fotos, Videos)	131
A.24 Kalender & Aktionen	132
A.25 Kollaboration	133
A.26 Nachrichten	134
A.27 Profil	135
A.28 Suche	136
A.29 Tagging	137

QUELLCODEVERZEICHNIS

2.1	Kontaktinformationen im HTML Format	11
2.2	Kontaktinformationen im hCard Format	11
2.3	XFN Codebeispiel[57]	11
2.4	Beispiel für eine FOAF Datei[163]	12
2.5	OpenID Aufbau	15
6.1	Zusammenstellung der Kontaktliste	77
6.2	Formel zur Berechnung der Schriftgröße in der Tagcloud (vgl. [153])	77
6.3	Facebook API Token generieren und Autorisierung anfordern	78
6.4	Facebook API Antwort auswerten	78
6.5	Zugriff per SOAP aus einer Facebook Applikation (vgl. [86])	79
6.6	Zugriff über die Facebook API auf Nutzerdaten	79
6.7	Anfrage an Google Maps API	79
6.8	Google Maps API Antwort in XML Datei schreiben	84
6.9	Vertauschen der Geokoordindaten aus Google Maps Antwort	85