

Fachhochschule Köln
University of Applied Science Cologne
Campus Gummersbach

Fakultät für Informatik und
Ingenieurwissenschaften

Diplomarbeit
(Drei-Monats-Arbeit)

zur Erlangung
des Diplomgrades
Diplom-Informatiker (FH)
Im Studiengang Wirtschaftsinformatik

Aufbau eines Hochschul-EIP's (Enterprise Information Portals)
mit Oracle Portal 3.0

Erstprüfer:	Prof. Dr. Heide Faeskorn - Woyke
Zweitprüfer:	Prof. Dr. Frank Victor
vorgelegt am:	02.12.2002
von cand.:	Darius Langner

Inhaltsverzeichnis

1	Einführung.....	1
1.1	Vorwort	1
1.2	Motivation und Ziel	4
1.3	Kapitelübersicht	6
2	Enterprise Information Portals	8
2.1	Allgemeine Definition.....	8
2.2	Anforderungen an EIP's	9
2.3	Eigenschaften von EIP's.....	13
2.4	Arten von Portalen.....	17
2.4.1	Business-To-Consumer Portale (B2C).....	17
2.4.2	Business-To-Busines Portale (B2B).....	18
2.4.3	Business-To-Emplyee Portale (B2E)	18
2.5	Aufbau und Inhalt von EIP's.....	18
2.6	Nutzen von EIP's.....	21
2.7	Beobachtungen des EIP Marktes.....	21
2.7.1	Stand von EIP's im Mai 2002	22
2.7.1.1	Wachstum des Portalmarktes	22
2.7.1.2	Portal Software Markt.....	22
2.7.1.3	Zeitraum bis zum Einsatz eines Portals	23
2.7.1.4	Portalpublikum.....	23
2.7.1.5	Gründe für den Einsatz eines EIP's im Unternehmen	25
2.7.1.6	Ermittlung des Portalerfolges.....	26
2.7.1.7	Communities innerhalb des Portals.....	27
2.7.1.8	Inhalte im Portal.....	28
2.7.2	Zukünftige Entwicklungstendenzen.....	30
2.7.2.1	Marktvolumen für Portale.....	30
2.7.2.2	Portaltrends bei den Herstellern.....	31
2.8	Web Services.....	34
2.8.1	Standards und deren Zusammenspiel.....	35
2.8.1.1	Standards innerhalb von Web Services.....	35
2.8.1.2	Zusammenspiel innerhalb der Web Services	36

2.8.1.3	Web Services - Standard für Schnittstellen.....	37
2.8.1.4	Beispiel von Web Services.....	37
2.8.2	Was sagen die Analysten	39
2.8.2.1	Anwendungsgebiete von Web Services.....	39
2.8.2.2	Was versprechen sich die Unternehmen	40
2.8.2.3	Was für Vorteile bringen Web Services?.....	40
2.8.3	Web Services in Portalen	42
2.8.3.1	WSRP als Web Service Erweiterung.....	43
2.8.3.2	WSRP und WSIA.....	44
2.8.3.3	Vergleich herkömmliche Web Services und WSRP.....	44
2.9	Portlet Standards	47
2.9.1	Portale und Portlets	47
2.9.2	Die Notwendigkeit von Standards	50
2.9.3	Java Portlet API	54
2.9.4	Web Services for Remote Portals (WSRP).....	54
2.9.5	Zusammenhang zwischen Portlet-API und WSRP	55
2.9.6	Schlussfolgerung.....	56
2.9.7	Einsatz der Standards bei Portalsoftware Herstellern	57
3	Konzept für ein EIP am Bsp. der FH-Gummersbach	58
3.1	Definition der FH-Köln, Campus Gummersbach-Homepage	58
3.1.1	Aufgaben der bisherigen FH- Homepage	58
3.1.2	Benutzer	59
3.2	Inhalte und Rollen des FH-EIP's.....	60
3.2.1	Rollen im FH-Portal	60
3.2.2	Inhalte im FH-Portal	61
4	Oracle Portal.....	64
4.1	Abstrakt	64
4.2	Portal Framework.....	65
4.2.1	Portlets	66
4.2.2	Portalseiten (Pages).....	69
4.2.2.1	Seitenlayout.....	69
4.2.2.2	Seitenstil.....	71

4.2.2.3	Seitenbanner	71
4.2.2.4	Seitensicherheit	72
4.2.2.5	Sprachanpassung	74
4.2.3	Navigation	74
4.3	Portaldienste	75
4.3.1	Content Areas	75
4.3.1.1	Ordner	75
4.3.1.2	Inhalt	77
4.3.1.3	Inhaltsklassifikation	80
4.3.1.4	Navigationsleiste	81
4.3.1.5	Stil	81
4.3.1.6	Content Area Suchdienst	83
4.3.2	Komponenten (Components)	83
4.3.2.1	Formulare	84
4.3.2.2	Berichte	86
4.3.2.3	Diagramme	87
4.3.2.4	Dynamische Seiten	88
4.3.2.5	Kalender	88
4.3.2.6	Hierarchie	89
4.3.2.7	Frame Driver	90
4.3.2.8	Menü	90
4.3.2.9	Gemeinsam benutzte Komponenten	91
4.4	Architektur und Sicherheit	92
4.4.1	Architektur	92
4.4.2	Sicherheit	96
4.5	Administration	97
4.5.1	Anwenderverwaltung	98
4.5.2	Gruppenverwaltung	98
4.5.3	Datenbankverwaltung	99
4.5.4	Überwachung	100
4.5.5	Zusätzliche Portaladministrations-Dienste	101
4.6	Java Portlet Development Kit (JPDK)	101

4.6.1	JPDK	102
4.6.2	provider.xml	102
4.6.3	Provider Adapter	105
4.6.4	Provider Interfaces	105
4.6.5	Provider Runtime	106
4.6.6	Provider Utilities	107
5	Praktische Umsetzung des FH-Portals.....	108
5.1	Konzept	108
5.1.1	Design der einzelnen Portalseiten	108
5.1.2	Benutzer und Gruppen	110
5.1.3	Inhalte.....	111
5.2	Erstellung von Seiten und Benutzern.....	112
5.2.1	Erstellung der Startseite des FH-Portals	112
5.2.1.1	Layoutvorlagen erstellen.....	112
5.2.1.2	Seite erstellen	113
5.2.2	Homepage der Informatik Fakultät	115
5.2.2.1	Layoutvorlagen erstellen.....	115
5.2.2.2	Seite erstellen	116
5.2.3	Homepage des FH-Personals	116
5.2.3.1	Layoutvorlagen erstellen.....	116
5.2.3.2	Seite erstellen	117
5.2.4	Benutzergruppen erstellen.....	118
5.2.5	Benutzer erstellen.....	119
5.2.5.1	Benutzer mit Professorenrolle erstellen	119
5.2.5.2	Benutzer mit Studentenrolle erstellen	120
5.3	Content Area Integration	121
5.3.1	Neue Content Area erstellen	121
5.3.2	Inhalte in Content Area einspielen.....	123
5.3.3	Content Area als Portlet verfügbar machen	124
5.3.4	Content Area ins Portal integrieren.....	124
5.4	Erstellung von Anwendungen für das FH-Portal	125
5.4.1	Anwendungsentwicklung.....	126

5.4.1.1	Anwendung: Eingabemaske.....	127
5.4.1.2	Anwendung: Studentenauswertung.....	129
5.4.1.3	Anwendung: Studentenauswertungs-Report.....	130
5.4.1.4	Anwendung: XML-Datenbankabfrage.....	132
5.4.1.5	Anwendung: XML-News.....	134
5.5	Ergebnisse der praktischen Umsetzung.....	136
6	Analyse.....	137
6.1	Einfacher Gebrauch.....	137
6.2	Universell einsetzbar.....	141
6.3	Suche und Navigation.....	144
6.4	Erweiterbarkeit.....	146
6.5	Kollaborativ.....	150
6.6	Administrierbar.....	151
6.7	Sicherheit.....	152
6.8	Skalierbar.....	155
6.9	Verfügbar.....	158
6.10	Personalisierbar.....	158
6.11	Oracle Portal und die Portlet-Standards.....	160
6.11.1	WSRP Standard von OASIS.....	160
6.11.2	Verwandtschaft von WSIA mit WSRP.....	161
6.11.3	JSR168 eine neue Java Portlet Spezifikation.....	162
6.11.4	WSRP und Oracle 9iAS Portal.....	163
6.11.5	JSR168 und Oracle 9iAS Portal.....	163
6.11.6	Wann sollen diese Standards verabschiedet werden?.....	164
6.11.7	WSRP und JSR168 im Oracle 9iAS Portal.....	165
7	Oracle WebDB und Portal 3.0.....	166
7.1	Verbesserung des Portal User Interface.....	166
7.2	Erweiterung der Content Bereiche.....	166
7.3	Verbesserung der Portlet Suche und Kategorisierung.....	167
7.4	Neue Tools und Administrationsfunktionen.....	167
7.5	Erweiterung der Anwendungsentwicklungsfunktion.....	167
7.6	Partnerprogramm von Oracle.....	167

7.7	Änderungen in der Architektur.....	168
7.7.1	Login Server Integration	169
7.7.2	Parallel servlet.....	169
7.7.3	Caching	170
8	Zusammenfassung.....	171
9	Anhang	175
9.1	Screenshots	175
9.2	Glossar.....	177
9.3	Abkürzungsverzeichnis	179
9.4	Literaturverzeichnis.....	181

Abbildungsverzeichnis

Abbildung 1: Yahoo! Internetpräsents im Jahr 1996	2
Abbildung 2: Consors Brokingportal Startseite	17
Abbildung 3: Aufgaben eines EIP's.....	20
Abbildung 4: Butlerumfrage vom November 2001	22
Abbildung 5: Forresterumfrage vom August 2001	23
Abbildung 6: Zeitraum bis zum Einsatz eines Portals	23
Abbildung 7: Anzahl von Benutzern	24
Abbildung 8: Vorteile für das Unternehmen (a)	25
Abbildung 9: Vorteile für das Unternehmen (b).....	26
Abbildung 10: Messbarkeit das Portabenefits.....	26
Abbildung 11: Anzahl an Communities im EIP.	27
Abbildung 12: Welche Bereiche sind an Communities interessiert.....	28
Abbildung 13: Kommerzielle Anwendungen im Portal.....	29
Abbildung 14: Selbsterstellte Web Services	29
Abbildung 15: Zugriff auf einen Web Service.....	36
Abbildung 16: Web-Services-Einsatz bei Simplicity	38
Abbildung 17: Anwendungsgebiete von web Services.....	39
Abbildung 18: Erwartungen der Unternehmen an Web Services	40
Abbildung 19: Erwartungen an Geschäftsprozesse.....	41
Abbildung 20: Erwartungen an den Marktprozess	42
Abbildung 21: Web Services Aufruf mitspezifischen Proxies	43
Abbildung 22: WSRP Aufruf mit generischen Proxies	44
Abbildung 23: Web-Services-Aufruf über lokale Portlets.....	45
Abbildung 24: WSRP-Portlet mit generischen Proxies	46
Abbildung 25: Beispiel eines Portals	48
Abbildung 26: Beispiel eines Portlets	48
Abbildung 27: Aufruf von Portlets	49
Abbildung 28: Existierende (grün) und fehlende (rot) Portalstandards	52
Abbildung 29: Vielfalt von Schnittstellen für Portal Komponenten.....	52
Abbildung 30: Gemeinsame Portlet API für Portal Komponenten.....	53
Abbildung 31: Portlets als WSRP Dienste in Portalen einbinden	55

Abbildung 32: Architektur und Dienste in vereinfachter Darstellung.....	64
Abbildung 33: Beispiel eines Portlets, zum abrufen von Fluginformationen.....	66
Abbildung 34: Beispiel Portalseite	66
Abbildung 35: Portlet für Benutzerverwaltung innerhalb des Portals	67
Abbildung 36: Defaultansicht für den Administrator	69
Abbildung 37: Menü zur Erstellung von Regionen	70
Abbildung 38: Menü zum editieren des gesamten Portalseite.....	71
Abbildung 39: Menü zur Editierung des Banners.....	72
Abbildung 40: Bsp. Spracheinstellungen beim „Verwalten“ Tab	74
Abbildung 41: Darstellung des Content Area Editors.....	77
Abbildung 42: Darstellung des Contentarea-Styl-Editors]	82
Abbildung 43: Container für Applikationen	84
Abbildung 44: Erstellung eines Formulars	85
Abbildung 45: Darstellung eines Formulars als Portlet im Browser	86
Abbildung 46: Mit Hilfe des Wizards erstelltes Diagramm	87
Abbildung 47: Kalender mit Zugriff auf Daten aus der DB	89
Abbildung 48: Frame-Driver gesteuerten Abfragemaske.....	90
Abbildung 49: Oracle Portal three-tier Architektur(vereinfacht).....	92
Abbildung 50: Oracle Portal 3.0 Architektur.....	93
Abbildung 51: Administrations-Seiten im Oracle Portal.....	97
Abbildung 52:Ausschnitt aus der provider.xml	104
Abbildung 53: Design der FH-Startseite.....	108
Abbildung 54: Design der Informatik Homepage.....	109
Abbildung 55: Design der Professoren Homepage.....	109
Abbildung 56: Menü zur Erstellen von Regionen	113
Abbildung 57: Layouts und Stiele zuordnen.....	114
Abbildung 58: Banner-Edit Menü.....	114
Abbildung 59: Linker- und rechter Teil des Logos der FH-Startseite	115
Abbildung 60: Teil des Logos der Informatik-Homepage.....	116
Abbildung 61: Teil des Logos der Professoren Homepage	117
Abbildung 62: Vergabe von Zugriffsrechten für die Gruppen.....	118
Abbildung 63: Neu erstellte Content Area.....	123

Abbildung 64: Auswahlmenü der verschiedenen Item Typen.....	123
Abbildung 65: Content Area Verzeichnis mit Edit-Möglichkeit.....	124
Abbildung 66: Menü zum Einfügen von Portlets	125
Abbildung 67: Anwendung: Eingabemaske zur Erfassung von Studentendaten	128
Abbildung 68: Inhalt der „Studenten“ Tabelle	128
Abbildung 69: Ausschnitt aus der FH-Portal Startseite.....	139
Abbildung 70: Seiten Edit Menü mit den einzelnen Funktionen.....	140
Abbildung 71: Auswertungsportlet mit Anzahl der Studenten.....	143
Abbildung 72: Navigationsmenü im FH-Portal.....	145
Abbildung 73: Navigator Linkbutton.....	145
Abbildung 74: Requests auf das FH-Portal im Monat 11-2002.....	152
Abbildung 75: Konfigurationsmöglichkeiten des Apache Servers	155
Abbildung 76: : Verteilte Oracle Portal Architektur.....	156
Abbildung 77: Sprachauswahl Portlet.....	159
Abbildung 78: Wirkungsbereiches von WSRP und JSR168	164
Abbildung 79: Oracle WebDB 2.x Architektur	168
Abbildung 80: Oracle Portal 3.0 Architektur.....	168
Abbildung 81: Fertige FH-Startpage.....	175
Abbildung 82: Fertige Informatik Homepage.....	176
Abbildung 83: Fertige Professoren Homepage	176

1 Einführung

1.1 Vorwort

Seitdem das Internet für jeden frei zugänglich geworden ist, hat es sich in kurzer Zeit zum Massenmedium entwickelt. In dieser neuen Onlinewelt wächst die Menge an Informationen ständig an. Auch die Unternehmen haben längst erkannt, sich die hier verfügbaren Informationen zu ihren Gunsten zunutze machen zu können. Sie begannen das Medium „Internet“ für ihre Geschäftsprozesse und Anwendungen zu verwenden und sie damit auch zu verknüpfen.

Schon seit längerer Zeit setzen die Unternehmen im alltäglichen „business“ verschiedene organisatorische und geschäftsprozessorientierte Softwaremodelle ein, um den Anwendern (Mitarbeitern und Geschäftspartnern) die Unternehmens- sowie Geschäftsdaten aus dem Intra- und Internet in einfacher Form zugänglich zu machen. Ziel dieses Informationsaustausches ist es die Geschäftsabläufe zu optimieren und somit ihre Effizienz zu steigern. Die immensen Investitionen in die unterschiedlichen Softwarelösungen, wie z.Bsp. Data Warehouse, Data Mart, Knowledge Management, *OLAP*, etc. mussten sich ja letztendlich amortisieren.

Diese Lösung brachte anfangs den gewünschten Erfolg mit sich, was aber mit der steigenden Zahl der Anwendungen und wachsenden Flut an Informationen und Funktionen sich zum Gegenteil kehrte. Denn der Anwender stand nun einer unüberschaubaren Informationsmenge und Funktionsvielfalt orientierungslos gegenüber. Er verbrachte zu viel Zeit mit der Suche nach den gewünschten Informationen, mit deren Hilfe er ja eine Entscheidung treffen musste. Ein weiteres Problem war die Handhabung der Anwendungen die ihm zur Verfügung standen. Ohne eine individuelle Einarbeitung mit dem jeweiligen Tool war der einzelne Mitarbeiter teilweise überfordert. Viele spezialisierten sich im Laufe der Zeit auf wenige Tools, mit denen sie dann arbeiteten, ohne sich die Funktionalitäten, der vielen anderen zur Verfügung stehenden Anwendungen, anzunehmen. Beispielsweise eine Mitarbeiterin in der Buchhaltung: Sie ist mit ihrem *FIBU*-Programm sehr vertraut, wenn es aber um die Suche nach einer neuen Steuervorschrift geht, bittet sie immer den Auszubildenden, damit er im Internet für sie nachschaut.

Um eine Lösung für solche Probleme anbieten zu können, haben sich einige IT-Software-Hersteller eines bis dahin bereits bekannten Konzeptes bedient:

dem Portal

Der Begriff „Portal“ war mit Aufkommen der ersten Suchmaschinen im Internet (z.Bsp. Yahoo!, Lycos) bekannt geworden. Diese Suchmaschinen dienten primär dazu, das WWW nach den vom Benutzer gesuchten Begriff zu durchforsten und ihm die Seiten aufzuzeigen, in denen dieser Begriff vorkam. Der Erfolg der Suche war oft von der „richtigen“ Eingabe des Begriffes durch den User abhängig.

Um die Suche für den User zu vereinfachen, wurden Seiten nach denen oft gesucht wurde (z.Bsp. Reisen, Finanzen), nach ihren entsprechenden Themen von den Suchmaschinenbetreibern zusammengefasst und auf der Startseite zur Verfügung gestellt. Das könnte man damals als die ersten Informationsportale im Netz bezeichnen.



Abbildung 1: Yahoo! Internetpräsents im Jahr 1996

Ein „echtes“ Unternehmensportal muss allerdings mehr als eine einfache Suchmaschine mit einer mehr oder weniger gut strukturierten Auflistung von URL's als Ergebnis einer Suchanfrage und eine thematisierte Linksammlung bieten könnten.

Im November 1998 wurde das Unternehmensportal von Christopher C. Shilakes und Julie Tylman von dem Finanzberatungsunternehmen Merrill Lynch als „Enterprise Information Portal“ (im folgenden *EIP*) definiert.

“Enterprise Information Portals are applications that enable companies to unlock internally and externally stored information, and provide users a single gateway to personalized information needed to make informed business decisions. They are: ... an amalgamation of software applications that consolidate, manage, analyze and distribute information across and outside of an enterprise (including Business Intelligence, Content Management, Data Warehouse & Mart and Data Management applications.)” [JMF199]

Im Jahr 2001 führte die Gartner Reserarch Group in ihrer Portalmarkt Studie [GPGA01] von den zahlreichen Anbietern die 32 vielversprechendsten Hersteller für EIP Lösungen auf dem internationalen Portalmarkt auf. Von denen sind inzwischen wieder einige verschwunden oder von großen Konzernen aufgekauft worden. In der aktuellsten Gartner Marktstudie vom Mai 2002 [GPRV02], sind zu diesem Zeitpunkt noch 25 namhafte Anbieter aufgeführt.

Dieser heiß umkämpfte Markt erlaubt es nur den innovativsten und anpassungsfähigsten Anbietern sich mit ihren EIP Lösungen beim Kunden zu profilieren. Als großes Kriterium zählt hier die Interoperabilität mit der beim Unternehmen bereits bestehenden Systemlandschaft und ihre Integration ins EIP. Dazu wird in Zukunft die Einhaltung verschiedener Standards Sorge tragen.

Einer der vielen Anbieter ist der Softwareriese Oracle, der mit seinem Produkt „Oracle Portal“ die Kundenanforderungen durch den Einsatz von Portlets und Content Areas zu erfüllen versucht

1.2 Motivation und Ziel

Enterprise Information Portals sind seit einiger Zeit zu einem Begriff geworden, der aus dem IT-Bereich eines Unternehmens nicht mehr weg zu denken ist. Entweder haben die Unternehmen bereits damit begonnen oder sie planen mit der Integration eines EIP's.

Doch was genau ist ein EIP eigentlich und wieso erwarten Analysten, wie die der Giga Information Group, ein Gesamtvolumen bei Lizenzeinnahmen für EIP-Software von rund zwei Milliarden US-Dollar? Auch die Unternehmen versprechen sich von ihnen einen erhöhten Informationszugriff. Doch wie genau kann das erreicht werden und mit welcher Technologie? Dabei fallen unter den IT-Verantwortlichen oft Begriffe wie „Portlets“, „Content Management“ oder auch in letzter Zeit immer öfters „Web Services“.

Seit der Begriff „EIP“ erstmals von den Analytikern Shilakes und Tulman von Merrill Lynch definiert wurde, sind viele weitere Definitionen zum EIP entstanden oder verändert worden.

Es wird oft vom Business Portal, Business Intelligence Portal, Knowledge Portal oder auch Collaborative Processing Portal gesprochen. Das Grundkonzept all dieser Definitionen bleibt zwar gleich, jedoch hat jeder Portal-Software-Hersteller den Begriff nach eigenen Interessen oder Stärken ausgelegt um sich am Softwaremarkt profilieren zu können.

Auch der US-Konzern Oracle, der sich mit seiner überzeugender Datenbanktechnologie seit Jahren zu den Marktführern im Datenbankbereich zählen darf, wollte mit dem Produkt „WebDB“ ein Stück vom EIP-Kuchen für sich gewinnen. Da diese Softwarelösung sich aber nicht so recht etablieren konnte, was sicherlich an der eingeschränkten Integrationsmöglichkeit von Anwendungen lag, will man nun bei Oracle mit dem „neuen“ Produkt „Oracle Portal“ den EIP-Markt für sich gewinnen. Es soll laut Angaben des Herstellers allen Anforderungen an ein EIP gewachsen sein.

Mit dieser Software soll es Unternehmen möglich sein, ihre gesamte Softwarelandschaft über den Webbrowser aufrufen zu können. Doch nicht nur Unternehmen wollen mit EIP's ihren Benutzern einen einheitlichen Zugangspunkt zu Informationen und Anwendungen bieten. Auch Gemeinden und öffentliche Einrichtungen versuchen diesen Schritt zu wagen. [URLS]

Genau hier setzt diese Diplomarbeit auf, mit der dem Leser die theoretische und praktische Umsetzung eines Hochschul-Portals für die Fachhochschule Köln, Campus Gummersbach erläutert werden soll. Doch bevor mit der Portalentwicklung begonnen werden kann, muss ein allgemeiner Überblick über EIP's, die verschiedenen Arten davon und ihre Zukunftsaussichten gegeben werden. Auch die Anforderungen, die ein EIP erfüllen sollte werden angesprochen. Die in dem Portal zu tragen kommende Technologie und Webstandards müssen erläutert werden, damit der Leser besser mit der zur Zeit vorherrschenden Problematik, bei der Anwendungsintegration zwischen verschiedenen Portalen, konfrontiert werden kann. Es werden Beispielsanwendungen genannt, die in einem FH-Portal zum Einsatz kommen könnten.

Anschließend wird der Leser den Aufbau und die Funktionalitäten des Oracle Produktes „Portal 3.0“ kennen lernen, damit ein Verständnis für Tools und Anwendungen, die bei der praktischen Umsetzung eines Beispielportals verwendet werden, vorhanden ist.

Abschließend wird das EIP-Produkt von Oracle auf Grund der erwähnten Anforderungen analysiert und kritisch auf die Frage geprüft: „Eignet sich Oracle Portal für die Erstellung und zukünftige Verwaltung und Pflege eines Portals für die Fachhochschule Köln, Campus Gummersbach?“

1.3 Kapitelübersicht

Im zweiten Kapitel werden Enterprise Information Portals (EIP's) definiert und die an sie gestellten Anforderungen sowie die daraus resultierenden Eigenschaften näher beschrieben. Ein Beitrag zu ihren verschiedenen Arten und Inhalten und Nutzen ist ebenfalls in diesem Kapitel zu finden. Aktuelle Auszüge aus verschiedenen Marktstudien, sollen über den aktuellen Stand und die zukünftige Entwicklung am EIP-Markt informieren. Der Begriff „Web Services“ und die damit zusammenhängenden Standards, sowie ihre Einsatzmöglichkeiten in EIP's schließen das Kapitel 2 ab.

Im dritten Kapitel wird ein Beispiel-Konzept anhand des Webauftrittes der Fachhochschule Köln, Campus Gummersbach beschreiben. Dies soll zeigen, wie ein Portal für die Fachhochschule effektiv eingesetzt werden könnte. Die hier genannten Beispiele sind theoretischer Natur, da eine praktische Umsetzung den Rahmen und die Möglichkeiten dieser Arbeit sprengen würde. Es dient als ein zusätzlicher Leitfaden, der die Funktion eines EIP's verdeutlichen soll.

Da die Umsetzung mit dem Produkt Oracle Portal erfolgen soll, wird im Kapitel 4 das Produkt (Version 3.0) genauer vorgestellt. Das Produkt wird beschrieben, indem die Portal Frameworks, die Portaldienste zur Informationsveröffentlichung, die Architektur sowie die Administration und Sicherheit genauer vorgestellt werden.

Kapitel 5 besteht aus der Erläuterung der Vorgehensweise bei der praktischen Umsetzung eines Beispielportals mit einigen Anwendungen und Content Areas für die Fachhochschule Köln, Campus Gummersbach. Dabei werden drei unterschiedliche Portalseiten erstellt, deren Endergebnis im Anhang als Abbildung 81, 82 und 83 begutachtet werden oder live unter dem Link [FHPOR] angesehen werden kann.

Kapitel 6 befasst sich mit der Analyse des Produktes Oracle Portal anhand der aus Kapitel 2 abgeleiteten Anforderungen an ein EIP. Hier werden kritisch Stärken und Schwachpunkte von Oracle Portal beleuchtet und Erfahrungen, aus dem praktischen

Einsatz wiedergegeben. Im Zusammenhang mit den Standards für Web Anwendungen trägt hier die Stellungnahme eines Portalentwicklers bei Oracle einen Beitrag zur Produktanalyse bei.

Kapitel 7 beschäftigt sich ebenfalls mit der Analyse des Produktes Oracle Portal, da es sich jedoch hauptsächlich auf Neuerungen im Vergleich zum Vorgängerprodukt WebDB bezieht, wurde dieser kurzen Analyse ein eigenes Kapitel gewidmet.

Im letzten Kapitel 8 werden im Fazit die wichtigsten Merkmale von Oracle Portal in Verbindung mit EIP's zusammenfassend dargestellt. In Bezug auf die vorangehenden Kapitel wird verdeutlicht, wie das Portal für den Aufbau und Einsatz des Hochschulportals verwendet werden könnte. Vorschläge zur weiterführenden Arbeiten und Projekten werden ebenfalls in diesem Kapitel gegeben.

Die Arbeit findet ihren Abschluss mit dem Anhang, in dem Screenshots, das Glossar, ein Abkürzungs- und Literaturverzeichnis enthalten sind.

In der Arbeit sind typografische Konventionen verwendet worden, die im Folgenden näher erläutert werden:

Typografische Konventionen

Kursiv wird verwendet für:

- Abbildungsbeschriftungen
- Tabellenbeschriftungen
- Fachbegriffe und Abkürzungen, die im Glossar oder im Abkürzungsverzeichnis (Kap. 9.3 und 9.4) näher erläutert werden.

Fett wird zur Hervorhebung von Gliederungselementen verwendet.

2 Enterprise Information Portals

2.1 Allgemeine Definition

Aus der von Merrill Lynch definierten Anforderungen, lässt sich die allgemeine Definition für EIP's ableiten:

„Bei EIP's handelt es sich um Applikationen, die es dem Anwender durch einen personalisierten Zugang ermöglichen, an alle Unternehmensinternen und -externen Informationen zu gelangen um Entscheidungen treffen zu können. Des Weiteren bietet ein EIP eine Verknüpfung verschiedener Applikationen die innerhalb und außerhalb des Unternehmens zur Verfügung stehen, um die täglichen Geschäftsabläufe durchführen zu können.“

Der Zugang erfolgt über den *Webbrowser* (Netscape, Internet Explorer, Mozilla, etc.) und unterscheidet sich von herkömmlichen Webseiten durch die Möglichkeit, Daten und Informationen aus verschiedenen Quellen (Datenbank, Internet, etc.) und in verschiedenen Formaten (*HTML, XML, Java, etc.*) in einer dennoch konsistenten Weise im Browserfenster darzustellen.

Der Browser dient dem Benutzer nicht nur zur Darstellung der Informationen, sondern auch zur ihrer Personalisierung. Jeder Anwender kann ganz individuell eine persönliche Startseite für seine Ansprüche zusammenstellen. Sofern er entsprechende Zugriffsrechte besitzt, kann der Benutzer sogar Informationen selbst publizieren und Anwendungen oder Dienste innerhalb seines Wirkungsbereichs im Portal integrieren. Diese Funktionalitäten sind einfach und selbsterklärend über den Browser aufrufbar, ohne dass der Benutzer spezielle Kenntnisse über Datenbanken oder andere Portaltechnologien besitzen muss.

Die häufigsten vom Unternehmen im Portal integrierten Anwendungen sind:

Content Management

Business Intelligence

Data Warehouse / Data Mart

Data Management Systeme

2.2 Anforderungen an EIP's

Bereits im Jahr 1999 stellte der Analyst Wayne Eckerson [WAEC99] die 15 bedeutendsten Charakteristiken auf, die ein Unternehmens Portal aufweisen muss. Diese sollten von den jeweiligen Portalsoftware-Anbietern erfüllt werden, um mit ihrem Produkt am Markt bestehen und den Anforderungen des Kunden genüge tun zu können.

Fünfzehn Regeln nach Wayne Eckerson

Regel 1: An die gelegentlichen Nutzer richten

Unternehmen sollen ihre Portale hauptsächlich an den Bedarf der Informationssuchenden ausrichten, die einen einfachen und schnellen Zugang zu Daten und Informationen benötigen. Um zusätzlichen Einarbeitung zu vermeiden, sollte es intuitiv erlernbar sein und über den *Webbrowser* und *HTML* als Werkzeug zugänglich sein.

Regel 2: Intuitive Suche und Klassifizierung ermöglichen

Ein Unternehmensportal muss im Hinblick auf Klassifizierung von Objekten selbsterklärend sein. Über ausreichende Erläuterungen der Objekteigenschaften und eine mächtige Suchmaschine verfügen.

Regel 3: Zugriff auf Veröffentlichungs- und Abonentendienste

Das Portal sollte es Benutzern ermöglichen, Objekte in ein Repositorium (Datenbehälter) abzulegen und bestimmen zu können, welche Benutzer, Benutzergruppen oder Kanäle Zugriffsberechtigungen darauf haben sollen.

Weiterhin sollen andere Benutzer diese Objekte oder vordefinierte Kanäle abonnieren können.

Diese Abonnements (z. Bsp. eines Reports) soll es möglich sein, den Zeitplan, das Format, die Zustellungsart und die bevorzugte Methode der Benachrichtigungen festzulegen. Agententools, die vom Benutzer Datenquellen zugewiesen bekommen, sollen permanent nach Informationen zu bestimmten Themen oder Stichworten suchen und den jeweiligen Benutzer bei erfolgreicher Suche benachrichtigen.

Regel 4: Universelle Verknüpfung zu Informationsquellen ermöglichen

Das Unternehmensportal muss eine Vielzahl von verschiedenartigen Datenquellen, wie: *RDBS*, Multidimensionale Datenbanken, Dokumenten-management- und Email Systeme, *Webserver*, Nachrichtenzustellungsdienste, etc. in sich vereinen können.

Regel 5: Für dynamischen Zugang zu den Informationsquellen sorgen

Das Portal sollte mehr als nur ein Darstellungstool statischer Objekte sein. Es soll dynamisch auf die verschiedenen, durch Business Intelligence- und Dokumenten-Manangement-Tools erstellten Berichte und Auswertungen zugreifen und diese nach Bedarf aufbereiten können.

Regel 6: Einsatz von intelligenten Routingsystemen

Als Teil eines speziell definierten Workflow, soll das Portal im Rahmen der „erweiterten Veröffentlichungs- und Abonnementfunktion“ durch intelligente Verteilertools die abonnierten Dokumente an die individuellen Gruppen zustellen.

Regel 7: Integration von Business-Intelligence Toolset

Ein EIP muss eine ganze Reihe von Anfragen, Reports und Analysefähigkeiten im höchsten Stil unterstützen und *BI*-Anfragen in *HTML*- sowie in herkömmlichen Formaten ausgeben können.

Regel 8: Verwendung einer serverbasierenden Architektur

Um gleichzeitige Anfragen von mehreren tausend Benutzern bearbeiten zu können, muss ein EIP mindestens eine three- oder n-tier Architektur unterstützen. Bei der die Mehrzahl der Prozesse auf der Mittelschicht bearbeitet wird. Der dazu notwendige *Applikation Server* sollte auf UNIX oder einem Microsoft Windows NT-System basieren.

Regel 9: Verteilte Multithread Dienste erschaffen

Das EIP sollte aus „Multithread Applikation Services“ bestehen, welche auf mehreren Servern zur optimalen Lastenverteilung aufgesetzt werden können.

Regel 10: Eine flexible Zugriffsrechte Verwaltung ermöglichen

Administratoren soll es möglich sein, Zugriffsrechte für einzelne Benutzer, Gruppen oder diverse Rollen innerhalb des Unternehmens zu vergeben. Diese Rechte definieren das Thema oder die Kategorie die von den Berechtigten gesehen werden kann. Ebenfalls werden dadurch die Kanäle, die der Benutzer abonnieren darf, sowie die Funktionalitäten auf die er zugreifen kann bestimmt. Somit wird gewährleistet, dass ein Interaktivitätslevel auf Daten und Reports vorherrscht (wie ansehen, editieren, aktualisieren, erstellen).

Regel 11: Anhängen von externe Schnittstellen ermöglichen

Viele Unternehmen haben bereits eine enorme Anzahl an Benutzerverzeichnissen, Informationspools, portalähnlichen Diensten in ihrem bestehenden Intranet. Deshalb ist es außerordentlich wichtig, dass ein Portal Schnittstellen zu diesen Quellen bereitstellt.

Regel 12: Gängige Schnittstellen liefern

Die Portaldienste sollten durch eine eigene *API*¹ über anderen Applikationen abrufbar sein. Über diese *API* sollen auch die Entwickler das Portal gestalten und administrieren können.

Regel 13: Internet Sicherheit aufbauen

Das EIP muss über Netzwerkelevel-Sicherheit, Verschlüsselung, Sessionsmanagement verfügen. Auch ein zusätzlicher Authentifizierungsdienst zu Unternehmensinformationen, die besonderen Schutzmechanismen bedürfen soll verfügbar sein, um unautorisierte Zugriffe auf diese Daten zu vermeiden.

Dieser Dienst sollen auf Servern außerhalb der Unternehmens-Firewall platziert sein, um auch externen Benutzer die Portaldienste zur Verfügung stellen zu können.

¹ Application-Programming Interface

Regel 14: Kosteneffektiven Einsatz ermöglichen

Ein Portal bedarf einer *Thin-Client* Architektur um den immensen Aufwand zu vermeiden, auf Tausenden einzelner Arbeitsplätze die Software aufzuspielen und zu betreuen. Es muss einfach sein die Software zu installieren, zu konfigurieren und ihre Aktualität und Funktionalität aufrecht zu erhalten.

Regel 15: Persönliche Anpassung und Gestaltung sicherstellen

Entwickler und Administratoren sollten in der Lage sein das Portal dem Unternehmenslook anzupassen, um ein ganzheitliches „look and feel“ wie Grafiken, Banner und Kanäle getrennt voneinander zu erstellen.

Administratoren sollen Kontrolle über die Funktions- und Grafikinterfaces durch Benutzer und Gruppenprofile ermöglichen. Zusätzlich soll jeder Benutzer individuell das Portal seinen Ansprüchen und Vorlieben entsprechend personalisieren können.

2.3 Eigenschaften von EIP's

Trotz der abweichenden Konzepte von EIP's, lassen sich bestimmte Eigenschaften aus den oben aufgeführten Portalcharakteristiken ableiten.

Im Folgenden werden diese im Einzelnen aufgezeigt und kurz erläutert.

a) Einfacher Gebrauch

Ein Portal sollte an die Fähigkeiten der meisten, nicht versierten, Anwender angepasst sein. Es sollte die Browsertechnologie verwenden, da der größte Teil der Benutzer bereits mit dieser Technologie vertraut ist. Des Weiteren sollte an bereits bekannte grafische Darstellungen angelehnt sein (wie bei Yahoo!). Damit es dem Benutzer leichter fällt, sich daran zu Recht zu finden.

b) Universell einsetzbar

Ein EIP muss unabhängig von bereits bestehender Softwarelandschaft im Unternehmen, einen globalen Zugang zu diesen strukturierten und unstrukturierten Informationsquellen verschaffen. Wie das Intranet, Internet und Extranet.

c) Suche und Navigation

Einen erheblichen Anteil zu der Akzeptanz eines EIP's beim Anwender trägt eine einfache aber effiziente, textbasierende Suchmaschine bei.

Dem Anwender soll möglich sein, nach verschiedenen Wörtern und Begriffen suchen zu können und somit zu den entsprechenden Informationsquellen geleitet zu werden. Die Suchergebnisse sollen nur aus Ergebnissen bestehen, die der voreingestellten Benutzerrolle entsprechen. D.h. es sollen dem "Suchenden" nur die Ergebnisse angezeigt werden, die seinem Interessen- oder Aufgabenbereich entsprechen.

Ebenfalls von Bedeutung ist ein Übersichtliches Navigationsmenü. Es sollte dem Benutzer vertraute Verzeichnisstrukturen bieten (Yahoo! ähnlich), durch die er navigieren kann. Das aktuelle Verzeichnis sollte stets sichtbar sein, sowie die Möglichkeit immer zum vorherigen Verzeichnis zurück zu gelangen sollte gegeben sein. Die Rückkehr zum Home-Verzeichnis sollte für

den Benutzer mit einem einzigen Mouseclick durchführbar sein, wobei der Homelink übersichtlich im EIP platziert sein sollte.

d) Erweiterbar

Um es den Entwicklern oder Administratoren zu ermöglichen neue Anwendungen und Datenquellen ins Portal integrieren zu können, sollte es über offene Standard-Schnittstellen verfügen (z.Bsp. *Java API*). Dadurch kann ein ständiger Ausbau und Anpassung der Funktionalität des EIP's garantiert werden.

e) Kollaborativ

Damit Unternehmen und Anwender den größten Nutzen vom Portal haben, müssen die dort enthaltenen Informationen geteilt werden können. Dem Anwender sollte ermöglicht werden sich über, von seinen Interessen abhängige Themen, Projekte und Ereignisse informieren zu können oder sogar informiert zu werden. Dazu sollte sich das Portal verschiedener Kanäle bedienen. Wie Diskussionsforen, Email-Verteilerlisten und Gruppenverzeichnissen.

Dem Anwender sollte es ermöglicht werden diese Kanäle einzurichten und zu verwalten, sich an diesen anmelden können und eigene Beiträge und Informationen dort zu veröffentlichen.

Diesen selbst erstellten Kanälen können Anwender Zugriffsrechte vergeben und somit beeinflussen welche Benutzer, Gruppen oder andere Kanäle Zugang zu ihren Informationen haben sollen und was sie mit diesen Informationen machen können. Wie zum Beispiel lesen, ändern und speichern.

f) Administrierbar

Das EIP muss leicht handhabbar sein. Einfache grafische Werkzeuge, die es den Administratoren ermöglichen Sicherheitsschemata und Zugangsberechtigungen festzulegen, Benutzer- und Systemschnittstellen zu erstellen sollten verfügbar sein. Mit diesen Tools sind benutzerfreundliche

und übersichtliche Anwendungen und Assistenten gemeint, mit deren Hilfe die Administratoren neben den bereits erwähnten Funktionen auch die Datenbanken verwalten können und die Möglichkeit haben das Layout und das „look and feel“ anzupassen. Ziel dieser Assistenten-Tools ist es administrative Arbeiten zu unterstützen oder einzusparen.

g) Sicher

Die Sicherheit ist eines der Schwerpunkte des EIP's, weil hier eine sehr große Anzahl von Informationen und Daten gespeichert werden. Oftmals handelt es sich dabei um unternehmenskritische Daten, auf die nicht jeder Anwender zugreifen darf. Die berechtigten Benutzer hingegen sollen mittels einer Authentifizierung Zugriff erhalten können. Dadurch erkennt das System den Anwender, seine Gruppenzugehörigkeit und somit auch seine Rechte auf Zugriffe oder Manipulationsmöglichkeiten der Daten. Für Portalzugriffe die außerhalb des Unternehmensnetzes stattfinden, soll die Möglichkeit eines verschlüsselten Datenaustausches vorhanden sein.

h) Skalierbar

Mit der Erwartung ständig steigender Benutzerzahlen am Portal, liegt der Fokus auf der Fähigkeit eine hohe Anzahl (mehrere Tausend) von gleichzeitigen Suchanfragen, Zugriffen auf Datenbanken, dynamischen Browserdarstellung, etc. ohne Zeitverlust zu bewältigen. Hierbei steht die besonders kurze Antwortzeit an die Benutzer im Vordergrund. Mit folgenden Techniken sollte das EIP den kurzen Antwortzeiten genüge tun: Es soll aus einer (mindestens) 3-Schichtigen (three- or n-tier architecture) Architektur bestehen. Die meisten Anfragen werden dabei in der Mittelschicht verarbeitet. Es sollten Dienste wie *load-balancing* (Lastenausgleich) oder *multithreaded request broker* (Aufteilung von Anfragen) einsetzbar sein. Intelligente Caching-Mechanismen sollten für sich ständig wiederholende Anfragen und Applikationen zum Einsatz kommen.

i) Verfügbar

Da ein EIP als zentrale Informationsquelle eines Unternehmens dient und seinen Anwender die Informationsbeschaffung, -bearbeitung und -verwaltung ermöglicht, muss der Zugriff auf das EIP und seine Inhalte jederzeit gewährleistet werden. Durch globalen Einsatz ist oft notwendig Kunden, Lieferanten und Angestellten weltweit zu unterschiedlichen Zeiten einen Zugriff zu gewährleisten. Um diese notwendige Ausfallsicherheit und damit die Verfügbarkeit zu garantieren, sollten mindestens 2 parallel arbeitende Server mit gespiegelten Datenbanken für das EIP eingesetzt werden.

j) Personalisierbar

Der Anwender sollte die Möglichkeit haben, entsprechend seinen individuellen Informationsbedarf eine persönliche Startseite und Umgebung zusammenstellen zu können. D.h. nach der Login-Prozedur sollte er automatisch zu seiner persönlichen Startseite gelangen.

Eine weitere wichtige Funktion eines EIP's ist die Mehrsprachigkeit Unterstützung. Dem Anwender sollte auch die Sprachauswahl gemäß seinen Vorlieben ermöglicht werden. So sollte er das Portalumfeld (Menüs, Auswahlfelder, Hilfefunktion) in der von ihm definierten Sprache sehen können.

2.4 Arten von Portalen

Der Begriff Portal wird oft im Zusammenhang mit verschiedenen Begriffen verwendet, deshalb werden hier die wichtigsten erläutert

2.4.1 Business-To-Consumer Portale (B2C)

Die B2C-Portale bieten dem Nutzer die Möglichkeit über eine fertig zusammengestellte Startseite, mit vordefinierten Links zu unterschiedlichen Dienstleistern, Suchmaschinen und allgemeinen Informationen zu gelangen.

Die populärsten B2C-Portale wären AOL, Yahoo! und MSN.

Verdeutlichung des Nutzens eines B2C Portals am Beispiel:

Bevor ein Anleger sein Geld investiert, loggt er sich bei seinem onlinebanking Portal ein (hier: Consors). Dort studiert er die zusammengestellten Expertennews über Aktien und Fonds. Liest dazu die Wirtschaftsnachrichten und unterhält sich im Forum mit anderen Anlegern über etwaige Erfahrungen mit bestimmten Anlageformen. Nachdem er seine Entscheidung gefällt hat, klickt er auf das Onlinebanking (hier: Broking) und gibt die Order für den Kauf des Wertpapiers ein, über deren Ausführung er in wenigen Sekunden informiert wird. Auf Wunsch auch per Email.

The screenshot shows the Consors Brokingportal homepage. Key elements include:

- Navigation Bar:** Home, Broking, Finanzinfos, Research, Planung & Beratung, Community, Meine Dienste.
- Search Bar:** "Suche zu..."
- Kursabfrage:** A box for entering a stock symbol (WKN) and clicking "suchen". A callout box points to it with the text "Auskünfte über bestimmte Wertpapiere einholen".
- Chat und Foren:** A callout box points to the "Community" link in the navigation bar.
- Aktuelle Marktübersicht:** A callout box points to the "MARKTKURS SÄULE" table on the right.
- MARKTKURS SÄULE Table:**

Index	Wert	Änderung	Änderung %
DAX	3.503,29	-49,37	-1,39
Nikkei 50	5.191,29	0	0
Dow Jones	1.280,23	0	0
EURUSD	1,00	+0,01	+0,95
- Neues von Consors:** A section with a headline "Aktuelles zum Übernahmeangebot an die Continental AG" and a sub-headline "Böhrer hält die BNP als präsent 91,55% der Consors-Aktie".

Abbildung 2: Consors Brokingportal Startseite

Dieses Beispiel soll verdeutlichen, wie einfach es solche Portale machen, ihren Benutzer zum gewünschten Ziel zu führen ohne sich durch etliche Webseiten arbeiten zu müssen. Diese Vorteile führen oft zu zunehmend steigenden Besucherzahlen solcher *B2C* Portalanbieter.

2.4.2 Business-To-Business Portale (*B2B*)

Laut einer Studie von Forrester sind im Jahre 2001 mehr als 1,3 Mio. Unternehmen vernetzt. Daraus sind aus diesen firmenübergreifenden Netzwerken Tausende von *B2B*-Portalen entstanden. Der Unterschied zu *B2C*-Portalen ist hier, dass Wissenswertes für die Geschäftswelt gebündelt und zusammengestellt wird. Spezialisiert nach Industriezweig (Banken, Versicherungen, Autoindustrie, etc.), Geschäftsbereich (Personalwesen, Einkauf, Verkauf, etc.) oder Tätigkeitsfeld (Finanzwesen, Informationstechnologie, Physik, etc.) ist hier alles an neuesten Nachrichten und Errungenschaften aus dem jeweiligen Bereich zu finden.

2.4.3 Business-To-Employee Portale (*B2E*)

Diese Unternehmensportale sind die Schnittstelle zwischen den Intranet- und Extranet-Netzwerken der Unternehmen. Sie kombinieren interne und externe Unternehmensanwendungen und -ressourcen um ihren Mitarbeitern oder Kunden Zugriff auf alle für sie verfügbaren und personalisierten Dienste und Informationen über den Browser zu ermöglichen.

Laut einer Studie der Meta Group [BCME01] wollen bis Ende 2002 bereits 85% der Unternehmen ein eigenes Intranetportal eingerichtet haben.

2.5 Aufbau und Inhalt von EIP's

Es gibt (bisher) keine einheitliche Norm, wie EIP's aufgebaut werden sollen oder welche Informationen und Tools in ihnen vorhanden sein müssen. Jeder EIP-Software-Hersteller hat eigenen Vorstellungen und Standards. Es lässt sich jedoch feststellen, dass sich viele EIP's in ihrem Aufbau ähneln und auch dieselben Arten von Tools und Informationsobjekten zur Verfügung stellen.

Das Hauptmerkmal ist die Mischform des Intra-, Extra- und Internets. Aus mehreren Objekten aufgebaut, werden diese Seiten im Portal vereint. Der Zugriff darauf, wird durch Anwender- und Sicherheitsschemata bestimmt.

Auch das „Öffentlich – Privat“ Konzept, scheinen viele EIP-Software-Anbieter in ihren Lösungen integriert zu haben. Dieses Konzept ermöglicht es den nicht registrierten Anwendern (z.Bsp. Neukunden) sich in dem öffentlichen Bereich über Produkte, Dienstleistungen, Pressemitteilungen oder allgemeine Ausschreibungen zu informieren. Dieser öffentliche Bereich ist im privaten Bereich eingeschlossen, jedoch stehen dem angemeldeten Benutzer zusätzliche Informationen und Tools zur Verfügung. Dabei kann es sich um private Quellen (wie Email, Telefon- und Adressenverzeichnis, Kalender, etc.) handeln oder aber auch um Nachrichten, Terminverzeichnis, Neuigkeiten, Geschäftsberichte, Tabellen, etc. die direkt mit seiner Position und Aufgabe im Unternehmen zu tun haben. Zudem kann er hier seine Startseite personalisieren und seinen Ansprüchen nach anpassen. Darüber hinaus sind die diversen Unternehmenssoftwarelösungen wie Projektplanung, Content- und Datamanagement, Berichtswesen oder E-Commerce für den Benutzer aufrufbar.

Die folgende Darstellung soll die Aufgaben des EIP's verdeutlichen. [GFTH99]

Enterprise Information Portals

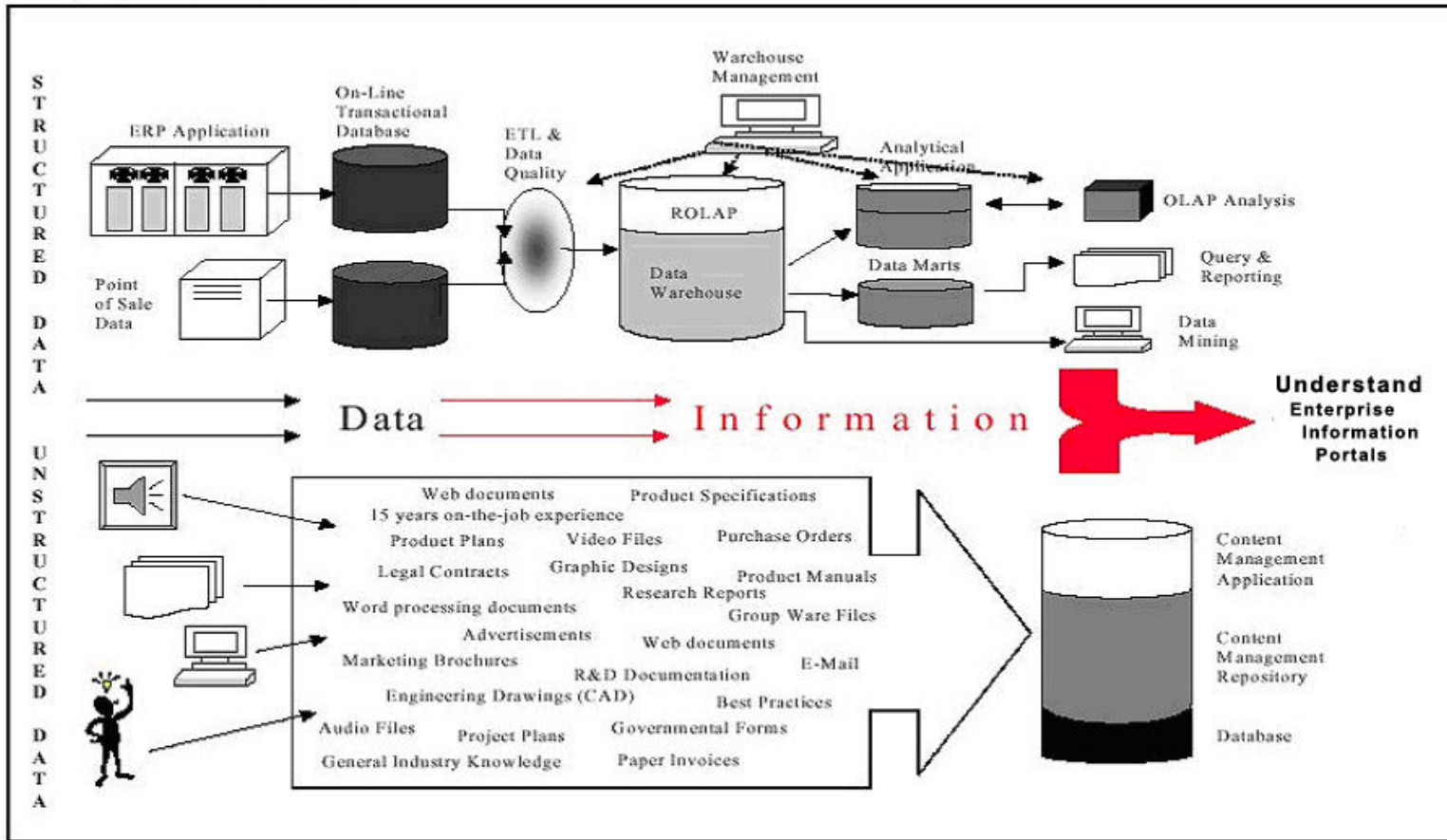


Abbildung 3: Aufgaben eines EIP's

2.6 Nutzen von EIP's

Der Nutzen, den die Unternehmen durch das EIP erwerben, liegt in der Steigerung der Produktivität der Anwender. Dieser Nutzen, der durch eine Verringerung der Informationsbeschaffungszeit jedes einzelnen Benutzers entsteht, ist der Zeitgewinn bei der Informations- und Ressourcensuche im täglichen Arbeitsumfeld.

Weil der Benutzer nun nicht mehr in unterschiedlichen Datenquellen, Webseiten, Systemen und Programmen suchen muss, um an die benötigten Informationen zu kommen.

Ein weiteres Kriterium ist die Informationsqualität die durch das EIP entsteht. Da dem Benutzer je nach Rolle und eigenen Definitionen genau die Informationen zur Verfügung stehen, die er auch benötigt.

Die durch das EIP erhöhte Effektivität und Produktivität der Benutzer erhöht auch diese für das Unternehmen. Es profitiert davon, dass die Aufgaben schneller erledigt werden können.

Die Schaffung von Wettbewerbsvorteilen liegt auf der Hand, wenn das Unternehmen durch den Einsatz von E-Commerce Tools im EIP zusätzlichen Informationen über ihre Kunden, Partner oder Konkurrenten gewinnen kann.

Ein weniger bedeutender Nutzen entsteht aus der Möglichkeit, Daten und Informationen mit andern Benutzern zu teilen. Das führt im Unternehmen zu geringeren Speichermedien Kosten. Denn nun steht z.Bsp. eine Text-Datei dem ganzen Team zur Verfügung und muss nicht an jeden einzelnen als Kopie per Email geschickt und gespeichert werden.

2.7 Beobachtungen des EIP Marktes

Der inzwischen vier Jahre alte Portalprodukt Markt wird seit 1999 von verschiedenen Beratungs- und Marktforschungsunternehmen analysiert und auf seine Entwicklungstendenzen hin beobachtet. Zu den Beobachtern gehören große Consulting Unternehmen wie META Group und ebenso bedeutende Marktforschungsunternehmen die sich auf Marktstudien spezialisiert haben wie Gartner Inc., Forrester Research, Delphi Group und Giga Information Group. Fast wöchentlich werden neue Berichte und Tendenzen, die den Portalmarkt betreffen, veröffentlicht. Es ist schwer für den Interessenten festzustellen, welche Werte und Trends nun die Aktuellen und damit auch vermutlich die zutreffendsten sind. Die folgende Zusammenstellung aus verschiedenen Marktstudien soll für etwas mehr Transparenz in der aktuellen und zukünftigen EIP Entwicklung sorgen.

2.7.1 Stand von EIP's im Mai 2002

2.7.1.1 Wachstum des Portalmarktes

Brian McDonough von der IDC sieht in seiner aktuellen Analyse, [BDID01] eine interessante Entwicklung auf dem Portalmarkt: Er erwartet eine Wachstumsrate des Marktes von 55 %. Das bedeutet eine Steigerung der Portalinvestitionen von 455 Mio. US-\$ in 2001 auf 705 Mio. US-\$ im Jahr 2002. Dabei stellte er fest, dass die üblichen Streichungen von IT-Projektausgaben im letzten Jahr bei Portalprojekten am geringsten waren.

Eine Gartner Studie hingegen schätzt den aktuellen Portalmarkt auf eine Milliarde US-\$ ein, erwartet aber einen viel geringeren Wachstum als IDC.

2.7.1.2 Portal Software Markt

Auf dem Software Markt sind inzwischen mehr als 100 Anbieter vertreten. Die meisten Portallösungen sind jedoch nur bei speziellen Anforderungen einsetzbar. Die großen Software-Anbieter jedoch wollen ihre EIP Lösungen als universelles Portalprodukt anbieten und haben begonnen, ihre Softwareangebote um entsprechende Portallösungen zu erweitern. In zwei unabhängigen Studien von Butler und Forrester wurden die bekanntesten Anbieter und deren Marktbekanntheit ermittelt. Dabei wurden die Kunden gefragt: „Wenn Sie ein Portalprodukt kaufen würden, welches würden sie dabei erwägen?“ Mit großer Mehrheit wurde dabei das Plumtree Produkt genannt. Dieses Ergebnis ist nicht überraschend, wenn beachtet wird, dass diese Umfrage in den USA erhoben wurde. Denn in den USA ist Plumtree bereits seit vielen Jahren als reiner Portalanbieter (Pure Play) am Markt vertreten und hat somit eine breite Marktdurchdringung erlangt. Weitere genannte Unternehmen haben dagegen ihre Bekanntheit mit anderen Soft- und Hardwarelösungen erlangt.



Abbildung 4: Butlerumfrage vom November 2001

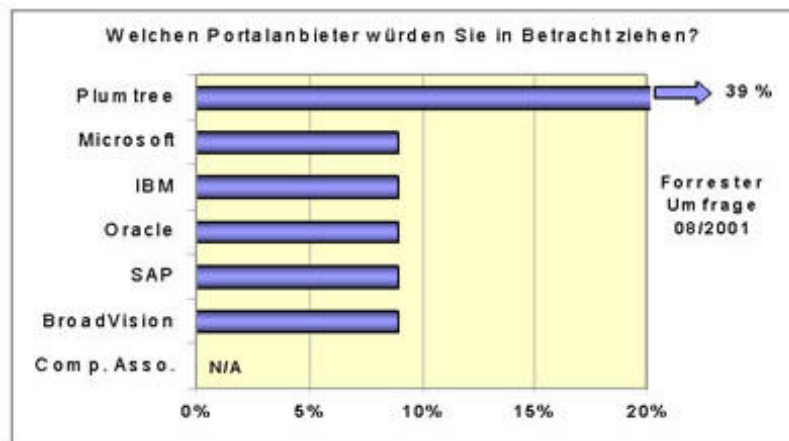


Abbildung 5: Forresterumfrage vom August 2001

2.7.1.3 Zeitraum bis zum Einsatz eines Portals

Ein Grund, wieso Portalprojekte nicht so schnell verworfen werden wie andere IT-Projekte, ist die kurze Umsetzungszeit von der Planung bis zum ersten Einsatz. In einer vom Portalsoftwarehersteller Plumtree in Auftrag gegebenen Studie [PLUM02] hat folgendes ergeben. Auf die Frage: „Wie lange haben Sie zum Einsatz ihres Portals benötigt?“ antworteten 34 % der Kunden, sie hätten weniger als 3 Monate gebraucht. Und weiter 33 % weniger als 6 Monate. Diese Ergebnisse bestätigten eine früher durchgeführte Umfrage der Delphi Group.

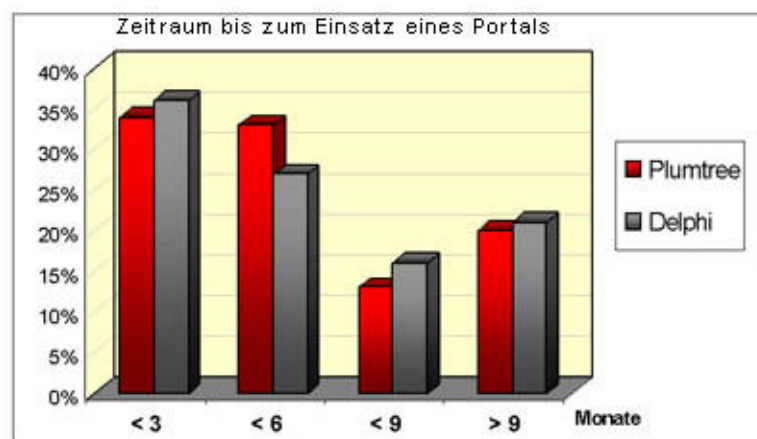


Abbildung 6: Zeitraum bis zum Einsatz eines Portals

2.7.1.4 Portalpublikum

Eine andere hervorragende Eigenschaft von Portalen ist die große Anzahl von Benutzern, die Unternehmen mit ihnen erreichen können. Damit sind nicht nur die Mitarbeiter sondern

auch die Kunden und Partner der Unternehmen gemeint. Wobei das Portal explizit zu diesem Zweck eingesetzt wird, um seine Benutzer zu den gewünschten Informationsquellen zu führen. Nur die Email übertrumpft das Portal als Lieferant von Informationen an eine große Anzahl von Mitarbeitern. Diese Tatsache führte dazu, dass viele Analysten das Portal als „den“ Lieferanten von Mitarbeiter Services und anderen Anwendungen innerhalb des Unternehmens ansehen.

Die Frage nach der Anzahl der Portalbenutzer, die bei portaleinsetzenden Unternehmen herrschen, lässt sich nur schwer beantworten, da viele Portallösungen für eine große Anzahl von Benutzern zur Verfügung stehen, einige von ihnen keine eigenen Accounts im Portal besitzen oder es nicht nur sporadisch besuchen. Deshalb wurde bei der Plumtree Umfrage die Frage gestellt: „Wie viele Benutzer in Ihrem Unternehmen besuchen das Portal mindestens einmal pro Woche?“. Das Ergebnis besagt, dass 34 % der Portale von unter 100 Anwendern regelmäßig verwendet wird und 30% von bis zu 1000 Anwendern. Nur 22 % gaben an ihr Portal wird von mehr als 10.000 Benutzern mindestens einmal pro Wochen besucht.

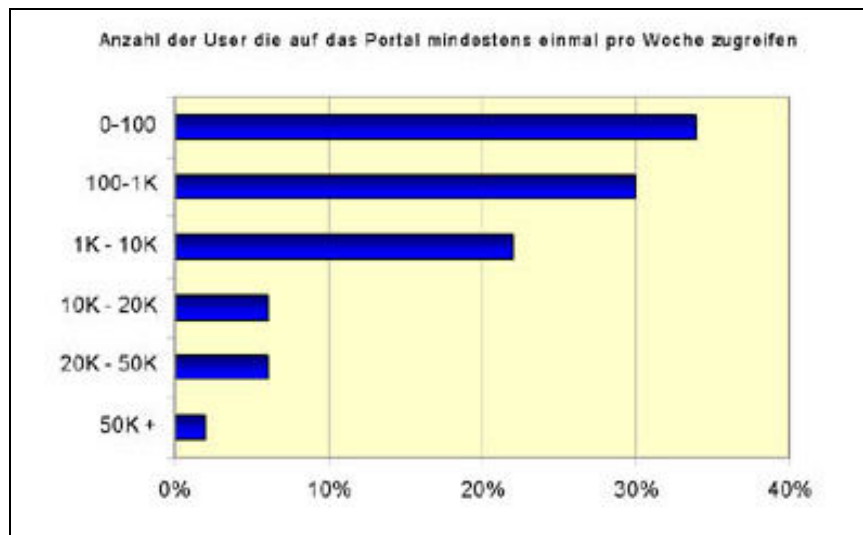


Abbildung 7: Anzahl von Benutzern²

Das zeigt die Tendenz mit der Unternehmen Portaleinsätze planen. Sie wollen entweder nur die Mitarbeiter mit den Unternehmensanwendungen und Informationsquellen versorgen oder zuerst sehen, ob sich das Portal innerhalb der Belegschaft bewährt und es dann später auch den Kunden und Partnern zugänglich machen. Diesen Trend bestätigt eine weitere

² Die mind. einmal pro Woche auf das Portal zugreifen

Umfrage von Plumtree, bei der hervorging, dass 59 % das Portal für Benutzer außerhalb des Unternehmens einsetzen möchten oder es bereits tun. Nur 37% wollen es als Mitarbeiterportal verwenden.

2.7.1.5 Gründe für den Einsatz eines EIP's im Unternehmen

Analysten und verschiedene Umfragen haben bestätigt, dass die meisten Unternehmen ein Portal primär dazu einsetzen um einen einfachen Zugang zu Informationen für ihre Mitarbeiter zu gewähren. So einfach der Grund auch ist, so bedeutend ist er für die Unternehmen, denn nur so ist der Zugang zu Unternehmensinformationen und Anwendungen für die Mitarbeiter möglich, ohne dass ihnen spezielle Kenntnisse oder Ausbildung abgefordert wird.

Die Unternehmensportale gewannen noch mehr an Popularität, als die Unternehmen aus Kostengründen eine Vielzahl ihrer weitgefächerten e-business Anwendungen abschafften und anfangen Personal zu entlassen. Bei der Plumtree Umfrage kristallisierten sich noch weitere Vorteile eines Portals für das Unternehmen heraus. Neben der mit 61 % angegebenen Erhöhung der Mitarbeiterproduktivität, sehen 18 % der Unternehmen das Portal als Hilfe zur Senkung der Webkosten und weitere 10 % zur Verbesserung des Kunden Self-Services und Supports.

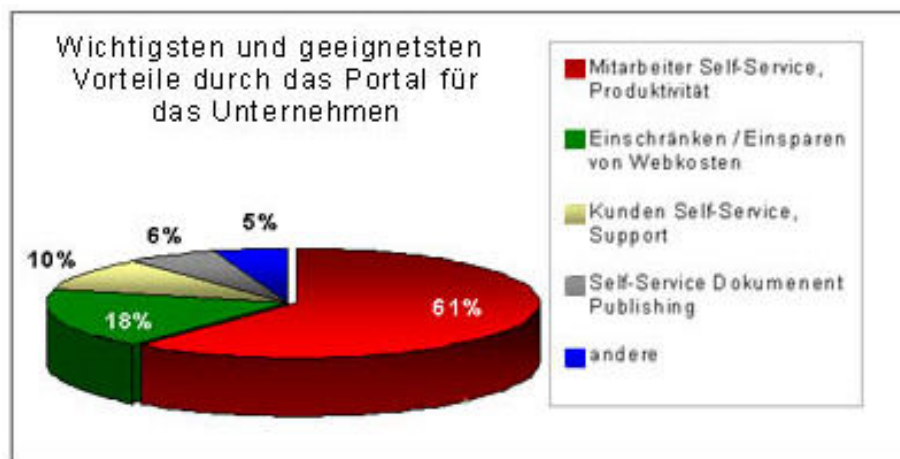


Abbildung 8: Vorteile für das Unternehmen (a)

Dieses Ergebnis wird von Studien, wie der von Forrester Research im August 2001 [FGFO01] durchgeführten, bestätigt. Es wurden 3500 Unternehmen nach dem messbaren Vorteil durch das Portal befragt. Dabei wurde „Zugang auf Informationen“ mit 53 % am

häufigsten genannt. Darin sehen die befragten Unternehmen den größten Vorteil, obwohl es sich als schwierig erweist ihn in Geldeinheiten zu bewerten.

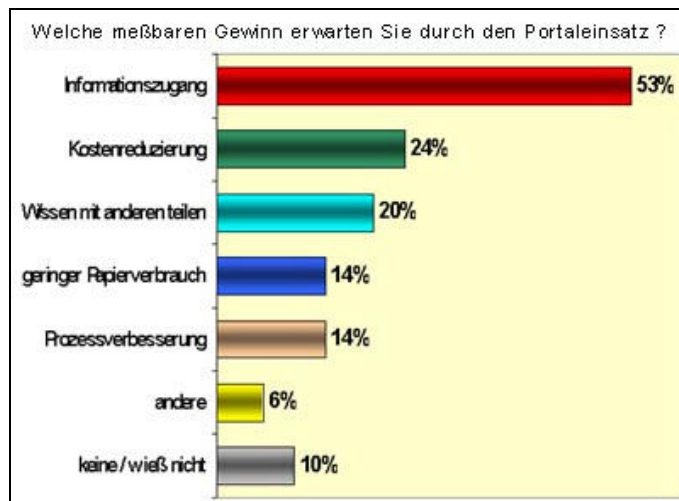


Abbildung 9: Vorteile für das Unternehmen (b)

2.7.1.6 Ermittlung des Portalerfolges

Viele Unternehmen sind nicht in der Lage ihren Erfolg, den sie sich durch den Portaleinsatz versprechen in Zahlen zu fassen. Dies belegt die Forrester Studie [FGFO01], in der es heißt, dass 41 % der Unternehmen ihren Erfolg gar nicht ermitteln und 20 % überhaupt nicht wissen, wie sie es anstellen sollen.

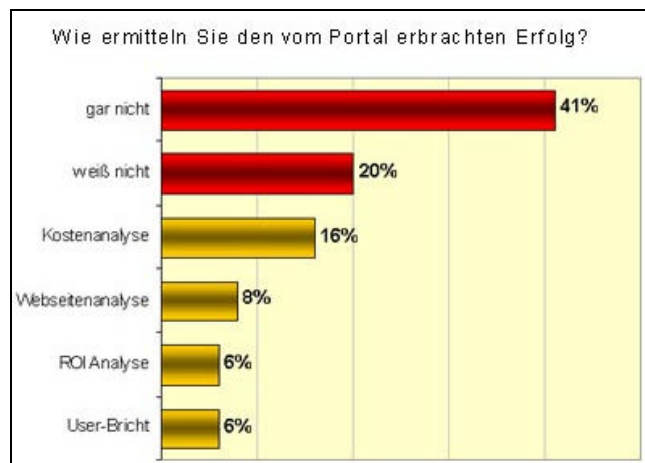


Abbildung 10: Messbarkeit des Portabenefits

Nach Aussagen einiger an der Studie teilnehmender Unternehmen, versuchen diese durch User-Feedback oder einfache Zählung von Pagehits an Werte zu gelangen, obwohl diese keine Tiefenanalyse ermöglichen. Weitere Aussagen sind, dass ein Portal durch den

einfachen Zugang zur Informationen dem Benutzer, also auch dem Unternehmen, Zeit einspart. Das alleine sehen viele Unternehmen bereits als langfristigen Erfolg an. Für einige der Befragten ist die Erreichbarkeit des Unternehmens durch mehrere Tausend User der hauptsächliche Vorteil den sie in dem EIP sehen.

2.7.1.7 Communities innerhalb des Portals

Plumtree definiert in seiner Studie eine Community wie folgt:

“A community is a group portal page, which can act as a resource-center for a workgroup or a business unit, and may host collaborative services.”

Damit ist ein gemeinsamer Arbeitsplatz gemeint, auf dem Inhalte direkt aus verschiedenen Businessbereichen veröffentlicht sind.

Für den Portalkunden ist diese Funktion, die ein Portal erfüllen muss, weitaus wichtiger als die ermöglichte Personalisierung der eigenen Portalseiten. Das liegt daran, dass innerhalb der Communities die Anwender ihre gemeinsamen Arbeitsbereiche kreieren können, in denen sie Foren gründen, Geschäftseinheiten zusammenfassen, Inhalte erfassen, Businessprozesse automatisieren und damit die Zusammenarbeit innerhalb des Unternehmens fördern. Diese Zusammenarbeit wird in fast allen Unternehmensbereichen praktiziert. Wie die Studie herausstellte, entstehen immer mehr Communities innerhalb der EIP's, wobei über die Hälfte der Kunden mehr als 10 Communities in ihrem Portal erzeugt haben. Besonders starken Einsatz finden Communities in den Personal- und Verkaufsabteilungen der Unternehmen

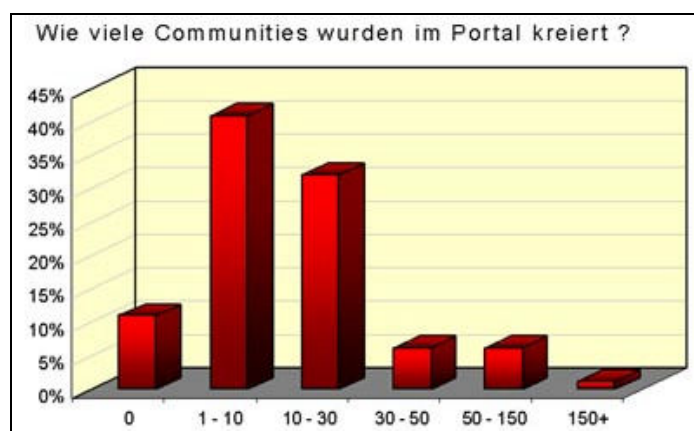


Abbildung 11: Anzahl an Communities im EIP.

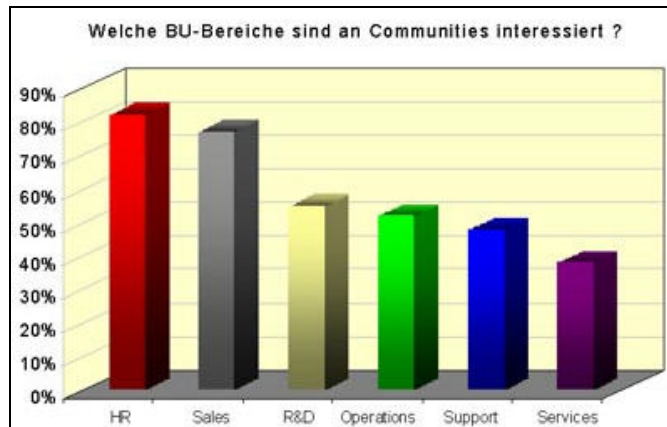


Abbildung 12: Welche Bereiche sind an Communities interessiert

2.7.1.8 Inhalte im Portal

Bei den Inhalten, die in ein Portal integriert werden, handelt es sich i.d.R. um Dokumente oder um Anwendungen, die in verschiedenen Formen im Portal verfügbar sind. Zu den bei den Portalnutzern beliebtesten Anwendungen gehören laut der Plumtree Studie kommerzielle Systeme wie SAP und Siebel oder auf eigenen Systemen basierende Tools. Dabei muss es sich nicht immer um Anwendungen handeln, die direkt mit dem Geschäftsgeschehen zu tun haben, denn wie die Umfrage zeigt, sind beim Anwender auch die „simplen“ Anwendungen beliebt. Dazu gehören:

- Unternehmens Nachrichten
- Mitarbeiterverzeichnis
- Verkaufszahlen
- Speisekarte
- Zugang zum gemeinsamen Plattennutzung
- Aktienticker
- Text Pager
- Urlaubstage Rechner

Diese Anwendungen werden im Portal über serverbasierte Connectoren implementiert, die je nach Portal-Software-Anbieter jeweils einen anderen Namen tragen, wie z.B. Portlets, Iviews, Gadgets, Web parts, e-clips, Omlets, Eclips. Diese Anwendungen können vom Kunden selbst programmiert oder von Fremdanbieter solcher Dienste bezogen werden. (Stichwort: Web Services. Siehe dazu Kap. 2.8 – 2.9)

Die Studie legt offen, dass 62 % der Plumtree Kunden in ihrem Portal ca. 1000 Dokumente indiziert haben aber nur 40 % haben mehr als 3 kommerzielle Anwendungen integriert. Darunter fallen Anwendungen, die im Unternehmen in Form von *ERP*-, *CRM*- und Dokumenten Management Software oder Email bereits bestanden haben. Bei den Web Services haben aber über die Hälfte der Umfrageteilnehmer bereits 10 oder mehr eigene Portlets entwickelt und ins Portal gestellt und mehr als 12 % sogar über 50 Anwendungen. Daraus lässt sich schließen, dass Inhalte und eigene Anwendungen bereitwilliger integriert werden als kommerzielle Software.

Bei der bereitwilligen Inhaltsintegration liegt es daran, dass es sich bei Dokumenten um die Inhalte handelt, mit denen die Geschäftswelt am häufigsten agiert und sie schon vor der Portalintegration vorhanden waren, aber in verschiedenen Quellen, Filesystemen und Darstellungsformen. Mit dem Portal kann endlich Ordnung geschaffen werden und alle Dokumente von einem einzigen Zugangspunkt erreicht werden.

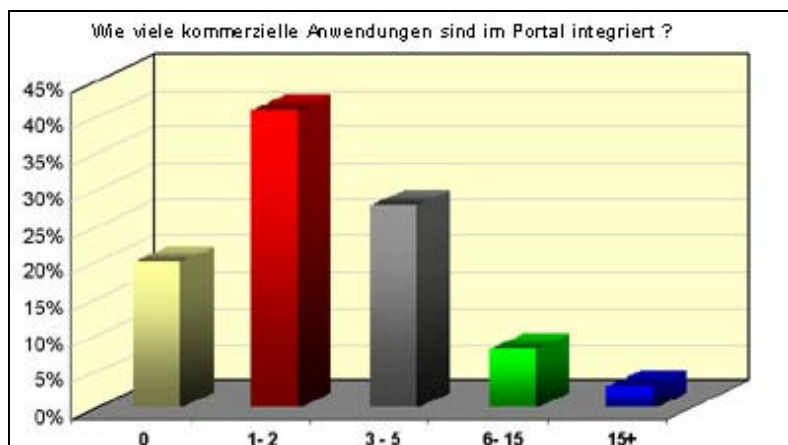


Abbildung 13: Kommerzielle Anwendungen im Portal



Abbildung 14: Selbsterstellte Web Services

Die geringe Anzahl von Applikationsintegration liegt vermutlich an den unterschiedlichen Plattformen, auf denen die Anwendungen entwickelt wurden. Zudem besitzen viele Kunden ein Portal das im Application Server (AS) integriert ist und mit der Software des jeweiligen Anbieters die beste Kompatibilität zeigt. Auszug aus einer Giga Studie [GIGA01] zeigt, dass Softwarehersteller wie Microsoft mit 22 %, Oracle mit 7 %, IBM mit 33 % und BEA mit 6% mit ihren eigenen Application Server bei den Unternehmen vertreten sind. Deshalb sind viele AS-Kunden bereit auf Anwendungen des jeweiligen Herstellers zu warten, bevor sie beginnen eigene Anwendungen zu entwickeln

2.7.2 Zukünftige Entwicklungstendenzen

2.7.2.1 Marktvolumen für Portale

Lt. Meta Group Deutschland GmbH, wird das Marktvolumen des europäischen Portalmarktes bis zum Jahr 2002 auf ca. 6,8 Mrd. € geschätzt. [META02] Hierbei entfallen auf den deutschen Markt 1,61 Mrd. €. META Group geht davon aus, dass ein Grossteil der Unternehmen in den kommenden 4 bis 5 Jahren in Portale investieren wird. Das mittlere Investitionsbudget liegt bei über 60 % der Anwender zwischen 125.000 und 250.000 €. Großunternehmen sollen allerdings ein Investitionsvolumen von 500.000 bis 2,5 Mio. € für ihre Portalprojekte verwenden. [META01]

Laut eines Artikels im E-Commerce Magazin sieht Giga Information Group folgende Entwicklung am EIP-Markt [ECM02]:

„Die Giga Information Group geht davon aus, dass sich Enterprise-Portale zur grundlegenden Zugangsschnittstelle zu IT-Systemen und Unternehmensapplikationen innerhalb von Unternehmen entwickeln werden. Bis zum Jahr 2005 erwarten die Analysten von Giga ein Gesamtvolumen an Lizenzeinnahmen für Enterprise-Portal-Software von rund zwei Milliarden US-Dollar und durchschnittliche jährliche Zuwachsraten von rund 35 Prozent.

In diesem Jahr lagen die Lizenzeinnahmen für Enterprise-Portal-Software bei 850 Millionen US-Dollar. Laura Ramos, Direktor der Giga Information Group, geht davon aus, dass Enterprise-Portal-Lösungen nicht als separates Marktsegment bestehen bleiben werden, sondern zusammen mit den Bereichen Enterprise-Content-Management, Applikationsintegration und Suche/Klassifizierung zu Technologien für Softwareinfrastruktur-Plattformen konvergieren werden. Die Analysten von „Giga Group“

beurteilt als derzeit markttreibende Kräfte für den Wachstumsschub bei Enterprise-Portalen vor allem den Bereich Web-Service-Technologien, Standardisierungsinitiativen zur Vereinfachung von Applikationsintegrationen und das Bestreben, Inhalt und Applikationen in einer anwenderspezifischen, sicheren und regeldefinierten Umgebung zusammenzubringen.

Auch wenn sich im Marktsegment der Enterprise-Portale derzeit noch keine eindeutige Marktführerschaft herauskristallisiert habe, sei bis Ende 2002 damit zu rechnen, dass IBM einen bedeutenden Marktanteil gegenüber den wichtigsten Mitbewerbern Bea, Broadvision, Epicentric, iPlanet, Oracle, PeopleSoft, Plumtree, SAP, Sun und Tibco erreichen könne. Grundsätzlich müsse bei diesem breiten Spektrum jedoch zwischen Anbietern von Softwareinfrastruktur, Enterprise Resource Planning- sowie Enterprise Application, Content-Management-Lösungen sowie spezialisierten Portalanbietern unterschieden werden.“

2.7.2.2 Portaltrends bei den Herstellern

Der noch relativ junge Portalsoftware Markt hat bereits die dynamischste Entwicklung in der Geschichte von Enterprise Software erfahren. G. Phifer ein Analyst vom Marktforschungsunternehmen Gartner Research hat diese Entwicklung in 3 Phasen eingeteilt [GPGA01]:

1. Phase:

„Entstehung des Marktes und Wachstum von “Pure Play” Anbietern“

Der Markt wird von zahlreichen „Pure Play“³ Anbietern, die oft aus kleinen, in Privatbesitz gehaltenen, Venture Capital finanzierten Unternehmen bestehen, überflutet. Alle wollen auf der Portalwelle mitschwimmen und beginnen Softwarelösungen zu entwickeln. Jeder hat seine eigenen Standards und Plattformen, die er versucht beim Kunden durchzusetzen.

³ Pure-Play Hersteller vertreiben nur dieses einzige Portal Produkt. Sie haben sonst keine andere Software, mit der sie auf dem Markt vertreten sind. Zu dieser Schicht gehören die Meisten Portal Software Hersteller. (Bsp. Plumtree Software, Epicentric etc.)

2. Phase:

„Markteroberung durch die großen unabhängigen Softwarehersteller“

Die so genannten *ISVs* (Independent Software Vendors - Unabhängige Software-Hersteller) haben begriffen, dass die Nachfrage nach Portalprodukten größer als angenommen ist. Markteinschätzungen renommierter Institute bestätigen den enormen Wachstum des noch sehr jungen Marktes. Bsp.: Gartner Group, die ein Marktvolumen für das Jahr 2000 von 405 Mio. US-Dollar eingeschätzt hat. Verglichen den 197 Mio. US-Dollar im Jahr 1999.

Zudem gibt es noch viele Kunden, die sich nicht auf die Lösungen der „Pure Play“ Hersteller einlassen wollen. Sie warten lieber ab, bis Softwareriesen wie IBM oder Microsoft mit ihren eigenen Portallösungen auf den Markt kommen, damit sie diese auf ihre bereits im Unternehmen vorhandenen Softwareprodukte, wie Datenbanken, ERP-Lösung etc., des jeweiligen Anbieters aufsetzen können.

3. Phase:

„Konsolidierung des Marktes“

Diese dritte Phase in der sich der Portalmarkt bereits befindet, wird von Gartner Research noch mal mit verschiedenen Ereignissen charakterisiert.

„Pure-Play Exodus“: Damit ist gemeint, dass der überfüllte Markt von den „Pure Play“ Anbietern, die in der ersten Phase den Markt überschwemmt, gereinigt wird. Dies geschieht meist durch folgende Ereignisse:

- Ihnen gehen die Zahlungsmittel aus und sie müssen ihr Geschäft aufgeben. (Bsp.: Radnet oder Linq)
- Sie profilieren sich außerhalb des Portalmarktes oder bleiben innerhalb des Marktes als Nischenprodukthersteller. (Bsp.: iBelong oder Intraspect)
- Sie werden von großen Softwareanbietern aufgekauft. (Bsp.: Sagemaker von Divine - 01/2001, Sequoia Software von Citrix - 02/2001 und TopTier von SAP - 03/2001)

Die dritte Option war bisher die am meisten zutreffende, allerdings wird die erste Option im Jahr 2002 immer mehr die kleinen Anbieter treffen.

„Enter the Gladiators“: Dieses ebenfalls 2002 stattfindende Ereignis, wird von den *ISVs* dominiert und ihrer Bündelung von Lizenzen und Technologien. Zusammen mit der Entstehung des „Portal Application Servers“ (Mehr zum diesen Thema, im Gartner Bericht: „Next for Portal Products: The Portal Application Server“) werden hier die Anforderungen an die zukünftigen Portalprodukte bestimmt.

Das Ergebnis wird ein weiteres Aussieben einiger Hersteller bedeuten.

„Application Server Assimilation“: Die ersten Portal Application Server werden am Markt erscheinen. Diese werden aus einer Kombination bereits vorhandener Portaltechnologien und einem *J2EE*- (Java 2 Enterprise Edition) Application Server bestehen. Die großen *ISVs* und Middleware Anbieter, die sich auf Application Server spezialisiert haben können sich auf diesem Markt entwickeln.

„Generational Change“: Portalprodukte der so genannten „Generation 3“ werden voraussichtlich Ende 2002 oder Anfang 2003 zum „mainstream“ (Haupttendenz) auf den Portalmarkt werden.

„Major Paradigm Shift“: Webservices reifen heran und werden zum tatsächlichen Standard für Portlets. Portlethersteller werden ihre Produkte in Web-Service-fähigen „Hüllen“ ausliefern müssen. Damit wird der erste Schritt zum interoperabilitäts Standard auf dem Portal Markt vollzogen sein.

2.8 Web Services

Damit die Unternehmen nicht von dem im Kap 2.7.2.2 erwähnten „Major Paradigm Shift“ überrollt werden, begannen bereits einige von ihnen sich mit der Web-Services-Thematik zu beschäftigen. Dies zeigt zumindest eine von Info World veröffentlichte Umfrage unter den 100 wichtigsten Chief Technology Officers (CTO 100) [EC-WS].

Aus dieser, im späteren Abschnitt dieses Kapitels noch näher vorgestellten, Umfrage geht hervor, dass die Unternehmen große Erwartungen in die Web Services stecken. Auch Marktforscher wie die IDC sehen eine Zukunft für Web Services und sprechen von Umsatzerwartung in diesem Bereich von 1,6 Mrd. US-\$ für das Jahr 2004.

Doch was genau versteht man unter Web Services. Ist es etwa ein „cutting edge“ Produkt eines Softwareherstellers, ein Application Server oder etwa eine neue Programmiersprache?

Nein, denn bei Web Services handelt es sich viel mehr um einen neuen Standard des *W3C*. Auf diesen Standard haben sich alle Softwaregiganten wie Sun, IBM, Microsoft, SAP, Bea und Oracle geeinigt.

Als Bausteine von Web Services dienen eine Reihe weiterer wichtiger *W3C*-Standards die für Funktionen, Architektur und Distribution nötig sind.

Bei diesen Standards handelt es sich um *XML*, *SOAP*, *UDDI* und *WSDL*.

Um den Leser einen Einblick in das Thema „Web Services“ zu geben, werden im Kapitel 2.8.1 die einzelnen Standards kurz vorgestellt und deren Zusammenspiel innerhalb der Web Services aufgezeigt. Die dann folgenden Beispiele sollen nur eine kurze Vorstellung der Anwendungsbereiche geben. Sollte der Leser den Einsatz von Web Services selbst ausprobieren wollen, kann er unter folgender URL [AMAWS] life eine Bestellung über Web Services beim Onlinevertrieb „Amazon.com“ vollziehen.

Das Kapitel 2.8.2 stellt die Erwartungen der Unternehmen an die Web-Service-Technik und ihren geplanten Einsatz vor. Weitere Informationen zu diesem Thema sind unter der URL [EC-WS] zu finden.

Im letzten Unterkapitel 2.8.3 soll der Einsatz von Web Services innerhalb von EIP's und der damit notwendigen Standardisierung vorgestellt werden. Eine ausführliche Beschreibung der Standards erfolgt dann im Kapitel 2.9.

2.8.1 Standards und deren Zusammenspiel

2.8.1.1 Standards innerhalb von Web Services

XML - eXtensible Markup Language

Bei *XML* handelt es sich um ein ideales Format zum Datenaustausch zwischen verschiedenen Programmen und Plattformen. *XML* ist ein selbsterklärendes Dateiformat mit eigener Struktur. Wobei der Text durch bestimmte Feldbezeichnungen für fremde *XML*-konforme Systeme lesbar ist. Durch die so genannten Parser werden die Feldbezeichnungen vom Text getrennt und bei Bedarf hinzugefügt. So können die Daten problemlos weitergeleitet, gespeichert und angezeigt werden. Logik wird durch *XSLT* (*XML* basierende Programmiersprache) oder verwendete Werkzeuge wie Java, Perl, etc. realisiert. *XML* wird von allen Webservern verstanden und kann somit für alle Endgeräte, ohne die Basis des Systems verändern zu müssen, genutzt werden.

SOAP - Simple Object Access Protocol

Bei *SOAP* handelt es sich um einen Standard für das Versenden und Empfangen von Nachrichten über das Web. Dieses, für den Datenaustausch in einer dezentralisierten, verteilten Umgebung, verantwortliche Protokoll geht auf eine Entwicklung von Microsoft zurück, welche vom *W3C* als Standard eingestuft wurde. *SOAP* ermöglicht eine unkomplizierte Implementierung, da es selbst *XML*-basierend ist. Diese Tatsache führte bereits innerhalb von kürzester Zeit zu Implementierungen für *Java*, *Apache*, *WebSphere* und *Visual Basic*.

UDDI – Universal Description, Discovery and Integration

UDDI ist ein öffentliches Verzeichnis und dient der Lokalisierung der unterschiedlichen Webservices. Um jeden Web Service eindeutig zuzuordnen und vor allem auch auffinden zu können, wurde ein komplexes System zur Einordnung der Eigenschaften von Web Services entwickelt. Es definiert ein Geschäftsverzeichnis für Anbieter, damit diese ihre Dienste dort registrieren und Softwareentwickler sie von dort aus beziehen können. In dieser UDDI-Registry sind die Web Services immer von geschäftlichen Anwendungen abhängig. Daraus ergibt sich eine branchen- und unternehmensabhängige Kategorisierung. Das UDDI Verzeichnis kann sowohl manuell vom Entwickler, sowie auch als „runtime“ von Web Services durchsucht werden.

WDSL – Web Services Description Language

Unter *WDSL* versteht man die technische Beschreibung über die Funktionsweise der Services. Für den Aufruf eines Web Services seitens einer Applikation bedarf es einer genauen „Sprachregelung“. Somit ist die *WDSL*, selbst auf *XML*-basierend, für die verteilte Programmierung unabdingbar geworden. Die Beschreibung gibt genaue Auskunft über: das verwendete Protokoll, die Adresse, die Port Nummer, mögliche Prozeduren und Funktionen sowie die Formate für In- und Output.

2.8.1.2 Zusammenspiel innerhalb der Web Services

Application Server hosten die Web Services, wodurch sie für Web-Service-Protokolle zugänglich gemacht werden. Die Beschreibungssprache der Web Services heißt dabei *WDSL*. *WDSL* Dokumente wiederum beinhalten die Adresse und alle Befehle zum Aufruf des jeweiligen Web Service. Diese sind im *UDDI*-Business-Verzeichnis gespeichert und stehen abfragebereit zur Verfügung.

Der Ablauf bei einer Anfrage eines Web Services geschieht durch das Aufrufen eines *XML*-Request-Dokuments. Dieser Aufruf wird durch ein Response-Dokument beantwortet. Beide werden über das *SOAP*-Standardprotokoll im Internet weitergeleitet.

Die Abbildung 15 soll die einzelnen Schritte beim Zugriff auf ein Web Service verdeutlichen.

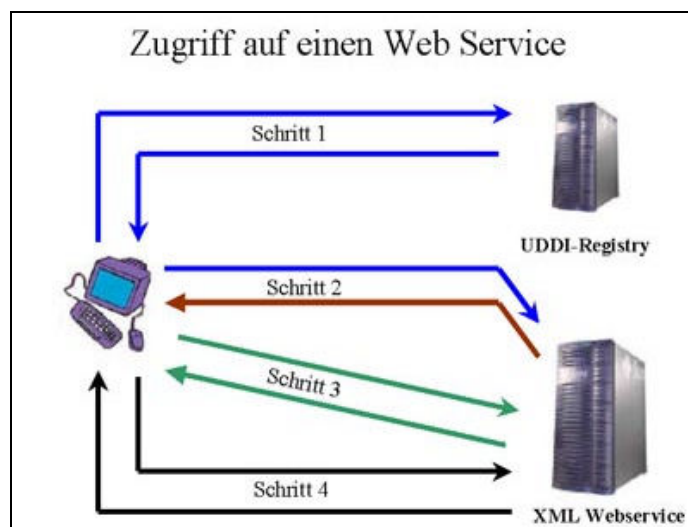


Abbildung 15: Zugriff auf einen Web Service

- Der Programmierer greift über die definierte *API* auf die *UDDI*-Registry zu. Er sucht dabei die Business-Entity-Informationen eines Partners, auf dessen Web Services er zugreifen möchte. Siehe Schritt 1.
- Im Schritt 2, lässt sich der Programmierer über die gefundene Business-Entity alle Binding Templates geben, welche die unterschiedlichen Web Services des Partners beschreiben. Dabei werden diese technischen Informationen auf der Clientseite gespeichert, was eine zukünftige Zugriffsminimierung auf die *UDDI* zufolge hat.
- Die erhaltenen technischen Informationen werden im Schritt 3 für die Ausgestaltung des Aufrufs des Web Services eingesetzt.
- Im vierten Schritt erfolgen der Aufruf und die Rückgabe des Web Service.

2.8.1.3 Web Services - Standard für Schnittstellen

Mit den Web Services wird ein einheitlicher Standard für Schnittstellen zwischen Applikationen und Contentservices im Internet und im Unternehmen festgelegt. Bisher stellte sich beim Versuch ein Unternehmen mit einem anderen Unternehmen, Extranet oder Marktplatz zu verbinden immer die Frage nach der universellen Integrationssoftware. Damit ist die so genannte Middleware zwischen zwei proprietären Systemen (wie: SAP, Oracle, HP, Bea) gemeint.

Das grundlegend Neue an diesen Standard liegt darin, dass ein Web Service von einem anderen aufgerufen werden und dessen Funktionen nutzen kann, als wäre es ein systeminternes Modul. Da die Schnittstelle jedes Services nach einem bestimmten Standard gestaltet und definiert ist, kann ein dritter Service aufgerufen werden, ohne die dahinter liegende Software-Infrastruktur zu kennen.

2.8.1.4 Beispiel von Web Services

Am Beispiel von den üblichen Vorgängen bei einer Reisebuchung wird die Einsatzmöglichkeit von Web Services kurz aufgezeigt.

Bei Hotelgruppen und Autovermietungen, die auf Basis von Web Services arbeiten, können sämtliche Prozesse, die für Reiseplanung vorab entscheidend sind, über Server-Programmaufrufe gestartet werden. Dazu gehören unter Anderem:

- Einkauf und Buchung von Flugtickets
- Hotel- und Mietwagenreservierung

Damit würde eine Reiseplanungs-Anwendung theoretisch ohne Beratung durch einen Mitarbeiter eines Reisebüros auskommen. Des Weiteren könnte ein Web Service der Fluggesellschaft oder des Flughafens über Fluginformationen eine ganze Prozesskette optimieren. Wie die automatische Benachrichtigung des Hotels und der Autovermietung bei Verspätung eines Fluges, etc.

Diese Mechanismen würden zu Kosteneinsparung und Steigerung der Funktionalität und der Kundenzufriedenheit führen.

Ein anderes Beispiel in dem die Web Services zu tragen kommen, kann auf [AMAWS] ausprobiert werden. Siehe dazu Abb.16.

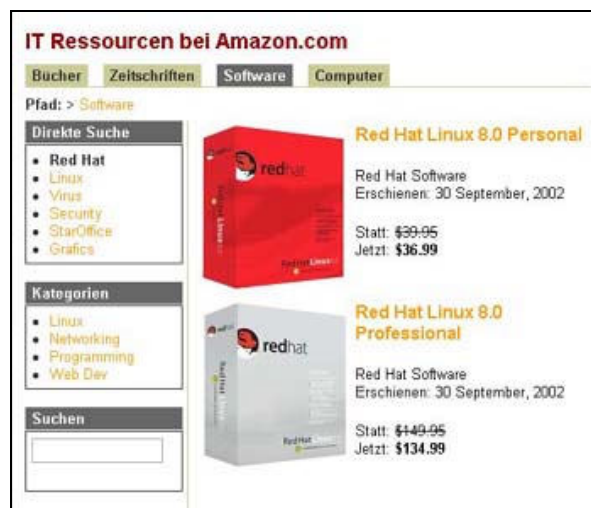


Abbildung 16: Web-Services-Einsatz bei Simplicity

Der Besucher der Homepage von Simplicity, einem Vertragspartner für Kommunikationslösungen, kann direkt mit Hilfe von Web Services diverse Artikel zum Thema „IT Ressourcen“ beim Versandhändler „Amazon.com“ bestellen, ohne direkt auf die

Amazon.com Homepage zu navigieren. Dazu wird über Web Services eine Sortimentsauswahl in den Bereichen: Bücher, Zeitschriften, Software und Computer direkt über Amazon.com zusammengestellt und mit Preis und einer kleinen Produktabbildung dargestellt.

2.8.2 Was sagen die Analysten

Die Bedeutung von Web Services scheint in den letzten Monaten enorm an Bedeutung gewonnen zu haben. Laut einer InfoWorld [EC-WS] Umfrage unter den CTO's 100 versprechen sich 84 % der Befragten enorme Veränderungen von dieser neuen Technologie. Besonders hebt sich dabei der e-Business Bereich ab, in dem 70 % der Befragten Web Services für andere Unternehmen zur Verfügung stellen wollen. Im Folgenden werden Ergebnisse aus Umfragen der Marktforschungsinstitute InfoWorld und Cap Gemini Ernst & Young zum Thema Web Services Ausschnittsweise vorgestellt. Weitere Auswertungen sind in der [WSCGE] nachzulesen.

2.8.2.1 Anwendungsgebiete von Web Services

Die befragten Technologie-Entscheider in den Unternehmen sehen den B2B (Business to Business) Markt mit 70 % als den Bereich mit dem größten Potential für den Einsatz von Web Services. An zweiter Stelle sehen sie den CRM- und Supply Chain-Geschäftsbereich dafür als geeignet an.

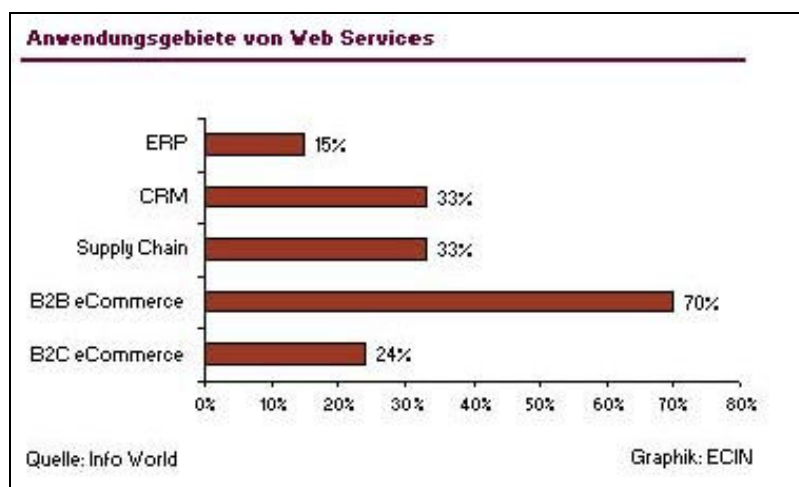


Abbildung 17: Anwendungsgebiete von web Services

2.8.2.2 Was versprechen sich die Unternehmen

Den erwarteten Erfolg von Web Services versprechen sich die Unternehmen nach der Implementierung besonders im Bereich der Kundenbeziehung (41%) und der Prozessoptimierung (22%). Wobei sich noch viele mit ihren Investitionen zurückhalten und lieber beobachtend abwarten. Denn wie aus weiterem Verlauf der Umfrage hervorgeht, planen 41 % der Befragten erst in mehr als 12 Monaten in diesem Bereich zu investieren. Das mag daran liegen, dass diese Standards frisch und noch nicht ausgereift sind.



Abbildung 18: Erwartungen der Unternehmen an Web Services

2.8.2.3 Was für Vorteile bringen Web Services?

In der von Cap Gemini Ernst & Young veröffentlichten Umfrage, bei der in deutschen Unternehmen die IT-Entscheidungsträger nach strategischen Aktivitäten im Web-Services-Bereich befragt wurden, stellte sich folgendes heraus:

Die meisten Unternehmen haben sich bereits mit dieser neuen Technologie befasst und besitzen in dem meisten Fällen bereits klare Vorstellungen über deren Einsatzgebiet, Erwartungen und die daraus resultierenden Vorteile für das eigene Unternehmen.

Im Geschäftsprozessbereich erwarten die Befragten eine Steigerung der Anwendungen, die Vereinfachung der Integration verschiedener IT-Systeme, eine modulare Gestaltung und Steigerung der Geschäftsprozesse. Weitere Erwartungen gehen aus Abb. 19 hervor.



Abbildung 19: Erwartungen an Geschäftsprozesse

Die Erwartungen an dem Marktprozess decken sich mit der InfoWorld Umfrage teilweise decken. Dazu gehört der erwartete Antrieb im e-Business Bereich sowie die Verbesserung der Dienstleistungen und deren Kombinationsmöglichkeiten. Weitere Umfrageergebnisse sind in Abb. 20 ersichtlich.

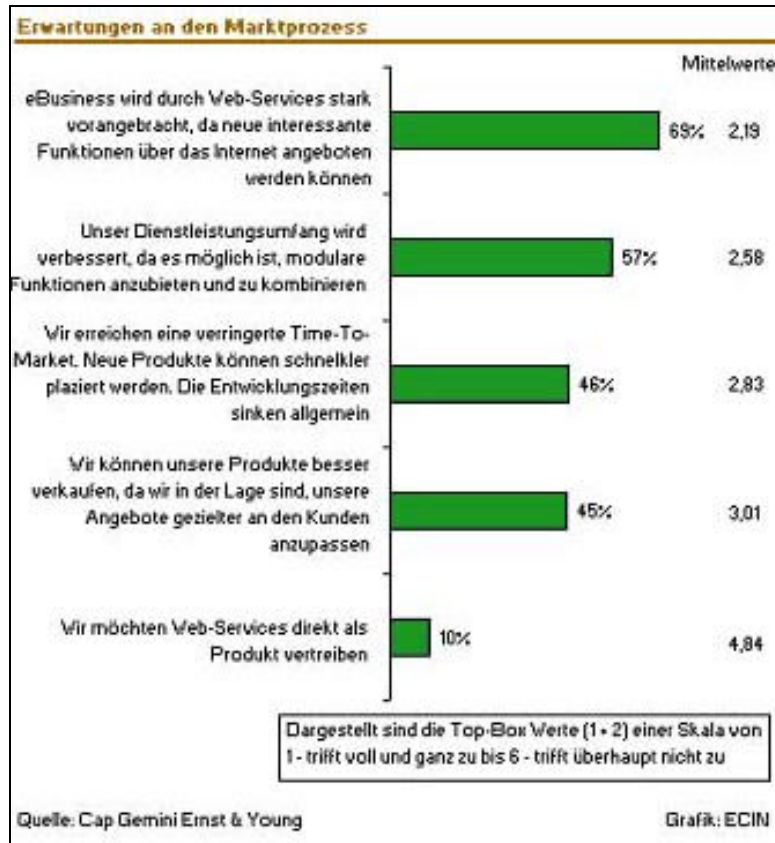


Abbildung 20: Erwartungen an den Marktprozess

2.8.3 Web Services in Portalen

Die Vision der Unternehmen Geschäftsprozesse rasch, effizient und kostengünstige mit den ihrer Partnern und Kunden zu integrieren, kann mit Einsatz von Web Services endlich realisiert werden. Unter Verwendung verschiedener, lose gekoppelter, wiederverwendbarer Softwarekomponenten, auf die mittels Standard-Internetprotokolle zugegriffen werden kann, ist dies möglich. Eine Zusammenstellung verschiedener eigener Dienste und Anwendungen kann dem Anwender zur Verfügung gestellt werden oder Dienste können einfach von entsprechenden Anbietern angemietet werden. Was bietet sich dafür besser an als die Integration dieser Web Services innerhalb eines Unternehmensportals mittels der Portlet-Technologie. Über diese Portlets kann der Portalbenutzer auf alle Services zugreifen oder von den automatisierten Abläufen profitieren.

Die Implementierung von portalkonformen Web Services bedarf es einer Erweiterung der im Kap 2.8.1 kennengelernten Web Services durch ein Benutzerinterface. Diese Erweiterung wurde von OASIS als *WSRP*⁵ definiert.

⁵ Web Services for Remote Portals

2.8.3.1 WSRP als Web Service Erweiterung

Der Bedarf dieses zusätzlichen Benutzerinterfaces entsteht aus der Tatsache, dass bei dem Aufruf von herkömmlichen Web Services die Clientapplikation eine *SOAP* Nachricht an einen Server sendet und die Antwort mittels *SOAP* wieder empfängt. Damit werden für jeden Service eigene Ein- und Ausgaben erwartet. In Abb. 21 ist schematisch dargestellt, wie ein Aktienkurs- und Wetterinformations- Web Service angesprochen wird.

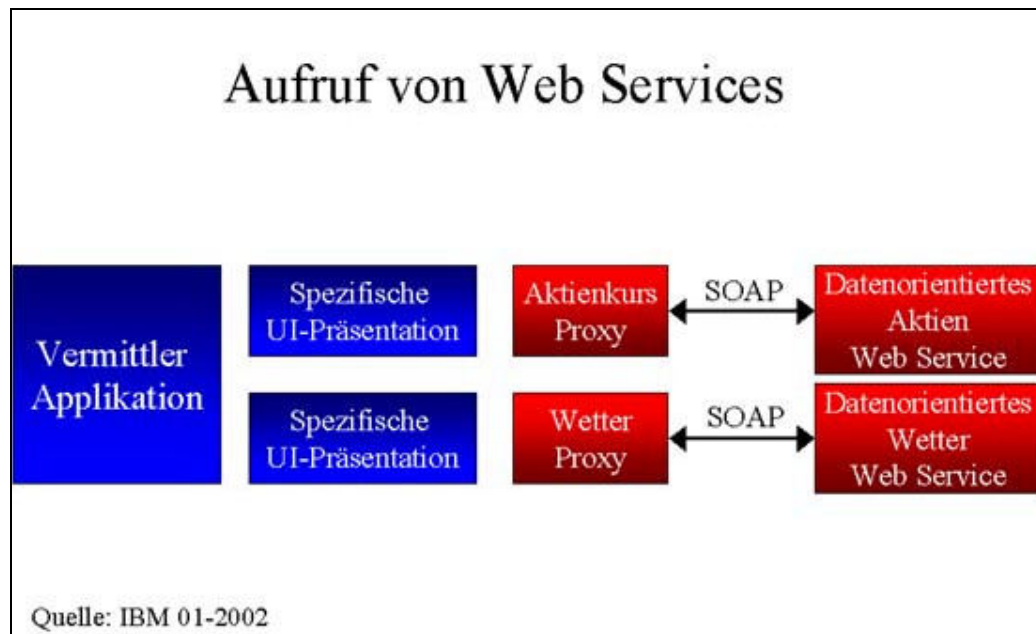


Abbildung 21: Web Services Aufruf mitspezifischen Proxies

Die Web Services für Aktienkurse und Wetterinformationen verwenden unterschiedliche *SOAP*-Nachrichten und eigene Schnittstellen. Somit werden für jeden Web Service individuelle Clientapplikationen benötigt.

Der Portaladministrator ist somit gezwungen, beim Hinzufügen eines neuen Dienstes ins EIP umfangreiche Codeänderungen vorzunehmen.

Eine vom Web Service Anbieter mitgeliefertes Interface würde dem Abhilfe schaffen.

WSRP sollen dabei helfen Web Services zu entwickeln, die ihre grafische Darstellung selbst bestimmen können und somit nur noch als Portlet an die Applikation eingefügt werden sollen, anstatt Darstellungscode um die Daten herum programmieren zu müssen.

Abbildung 22 stellt den Aufruf von Web Services mit WSRP durch die Erweiterung mit Benutzerschnittstellen und generischen Proxies dar.

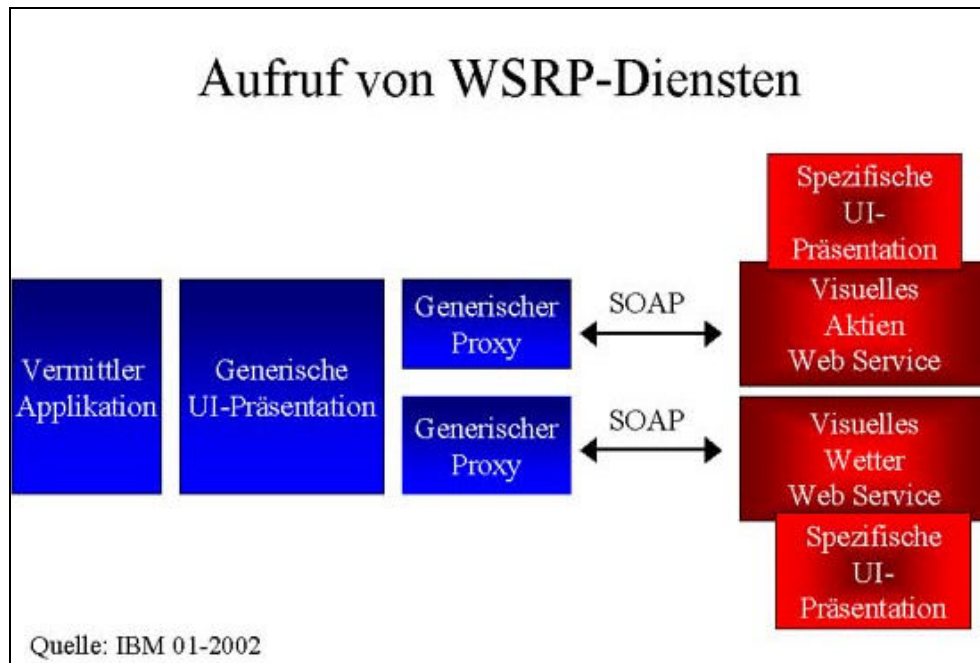


Abbildung 22: WSRP Aufruf mit generischen Proxies

2.8.3.2 WSRP und WSIA

Beim Einsatz von visuellen benutzerorientierten Web Services gibt es zwei Implementierungsarten. Zum einen ist es möglich selbst definierte Events und Ausgabeoperationen zu verwenden. Zum anderen die Reduzierung der Dienste zu einer konkreten, abgegrenzten Sammlung von Schnittstellen. Die *WSIA*⁵-Spezifikation bietet ein System zur Erstellung verschiedener Arten von benutzerorientierten Web Services.

Die *WSRP*-Dienste haben mit *WSIA* einige grundlegende Überlagerungen, bieten aber darüber hinaus noch weitere festgelegte Funktionen, die es ermöglichen *WSRP*-Dienste mit ihnen gefälligen *WSRP* Clients zusammenzuschließen. Es ist zudem möglich, *WSRP* Dienste zu nutzen, die Standard-*WSIA*-Operationen verwenden.

2.8.3.3 Vergleich herkömmliche Web Services und WSRP

Im Folgenden wird anhand eines Beispielen der Unterschied zwischen den herkömmlichen Web Services und den Web Services for Remote Portals (*WSRP*) dargestellt. [WSRP02]

Um ein Unternehmensportal mit einen HR-Dienst zur Gehaltsabrechnung und einem Wetterinformations-Dienst zu erweitern, gibt es für den Portaladministrator folgende Möglichkeiten: Die Entwicklung eines HR-Portlets und der eines Wetterinformations-

⁵ Web Services for Interactive Applications

Portlets, die lokal auf dem Portalserver laufen und ihre Daten über entfernte Web Services beziehen. Das Schema wird in Abb. 23 veranschaulicht.

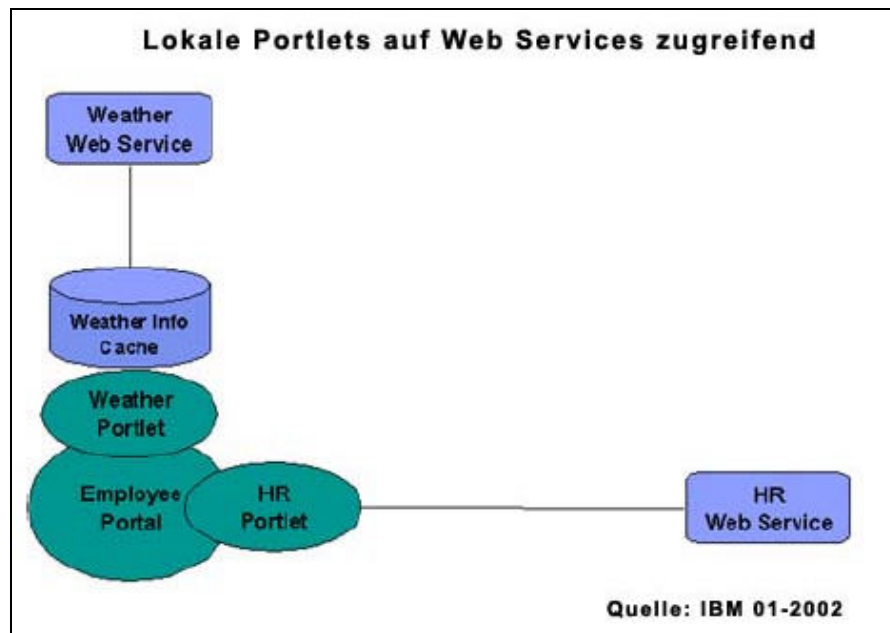


Abbildung 23: Web-Services-Aufruf über lokale Portlets

Dieses Verfahren funktioniert nur wenn alle Portlets physisch im Unternehmensportal installiert sind. Zudem ist der Prozess mühsam und kostspielig. Um das HR-Portlet ins EIP integrieren zu können, müsste entweder die HR-Abteilung das Portlet entwickeln und es dem Administrator zur Installation geben oder der Portalentwickler müsste das Portlet implementieren unter Berücksichtigung der HR-Web-Services Schnittstellen. Das gleiche gilt auch für den Wetterinformations-Dienst. In beiden Fällen ist ein immenser Aufwand und lokale Installation von zusätzlichen Programmcode notwendig um die beiden Portlets im EIP verfügbar zu machen.

Die andere Möglichkeit berücksichtigt die Tatsache, dass die Anbieter der Dienste das Benutzerinterface mitsamt Präsentations- und Anwendungslogik liefern sollen. Über generische Portlet-Proxies können standardisierte Web Services Schnittstellen einfach auf der Portalseite aufgerufen werden. Siehe dazu Abb. 24.

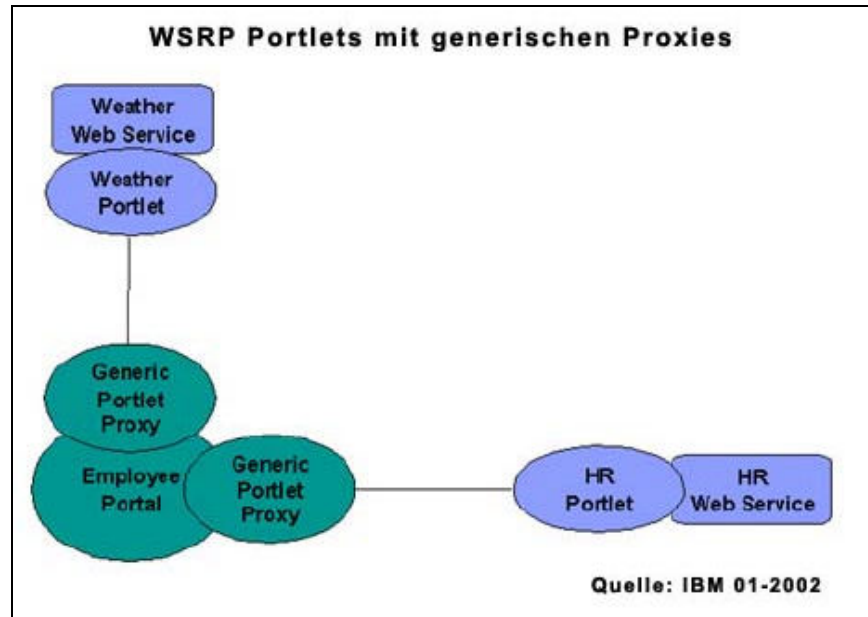


Abbildung 24: WSRP-Portlet mit generischen Proxies

Somit wird sich der Aufwand ein neues Portlet in ein EIP hinzuzufügen nur auf das Auffinden des Dienstes, und das Ankoppeln der neuen Instanz, der generischen Proxies, an den neuen Dienst beschränken.

Zum derzeitigen Zeitpunkt gibt es allerdings noch keinen einheitlichen Standard, mit dem das reibungslos funktioniert. Das soll sich lt. *OASIS*⁶ mit der Verabschiedung des *WSRP* Standards in Jahr 2003 ändern. [OASIS]

Viele namhafte Softwarehersteller haben sich bereits eigene Gedanken gemacht, wie eine solche standardisierte Architektur aussehen sollte. Im Kapitel 2.9 wird aus der Sicht vom Softwarehersteller IBM noch mal der Bedarf für einen Standard und die von IBM angestrebte Architektur für *WSRP* erläutert, die zwischen den Konsumenten und den Diensteanbieter eine Verbindung durch Protokolle ermöglicht.

Weitere Informationen zu *WSRP* sind aus dem Whitepaper von IBM zu entnehmen. [WSRP02]

⁶ Organization for the Advancement of Structured Information Standards

2.9 Portlet Standards

Mit der immens steigenden Anzahl von Enterprise Portals ist auch eine Vielfalt an verschiedenen *APIs* als Portalbestandteil (wie im Kap. 2.8.1 und 2.8.3 aufgezeigt), den sogenannten Portlets, von den unterschiedlichsten Anbietern entwickelt worden. Das hat zur Folge, dass unterschiedliche Mechanismen zum Aufruf der Visuellen Komponenten von den einzelnen Anbietern eingeführt werden. Diese Vielfalt an inkompatiblen Schnittstellen führt zu Problemen bei den Diensteanbietern selbst, den Portal Kunden, sowie bei den Portal-Software-Herstellern.

Um diesen Problemen beizukommen sollen zwei neue Standards für eine einheitliche Kommunikation zwischen den Portlets und dem Portal sorgen, sowie zwischen dem Portal und dem für den Benutzer sichtbaren Web Service.

Bei den Standards handelt es sich um die Java Portlet *API* und dem *WSRP*-Standard. Mit ihnen soll es für Diensteanbieter von Web Services und für Portalanwender möglich sein, Portlets und Web Services selber zu entwickeln und zu programmieren, unabhängig von dem jeweiligen EIP Produkt. Die Java Portlet *API* wird mit dem *WSRP*-Standard kompatibel sein und damit die Veröffentlichung von Portlets als Web Services ermöglichen.

2.9.1 Portale und Portlets

Portale sind Webseiten die einen effizienten Zugang zu personalisierten Informationen und Anwendungen aus unterschiedlichen Quellen ermöglichen.

Typische Portale basieren auf einer Ansammlung von visuellen Komponenten oder den so genannten Portlets.

Abbildung 25 zeigt ein typisches Beispiel eines EIP's, welches in einem HTML-Browser dargestellt wird.

In diesem Beispiel hat der Benutzer ein News-, Suche- und ein Marktreport Portlet ausgewählt. Diese sind in seiner personalisierten Seite vereinigt und stellen somit die relevanten Informationen auf einem Blick dar. Die Funktionalitäten von Portlets reichen von einfachen Darstellungen bis hin zu komplexen, interaktiven Anwendungen.

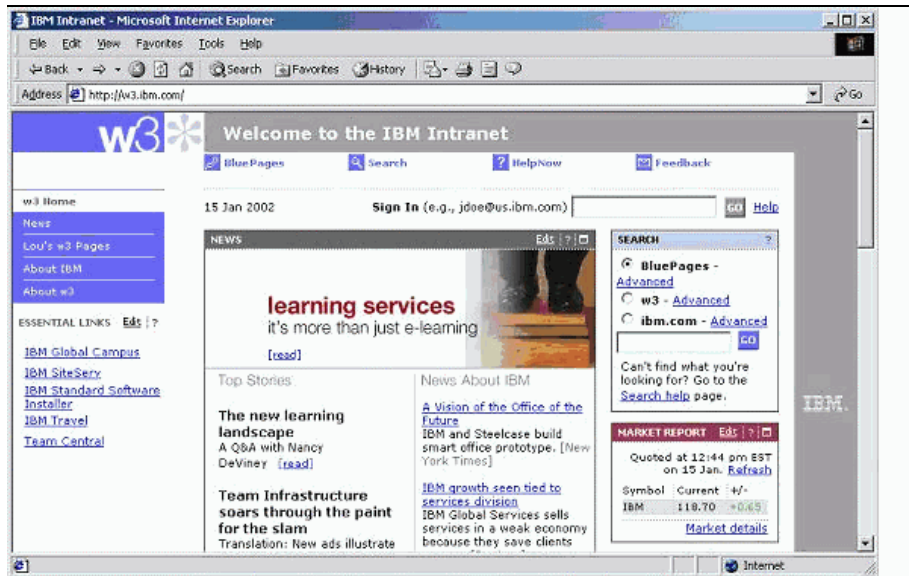


Abbildung 25: Beispiel eines Portals

Bei dem Marktreport-Portlet, welches in der Abb. 26 nochmals vergrößert dargestellt wird, handelt es sich um ein Portlet mittlerer Komplexität. Es hat einen Ansicht-Modus, welcher die Liste verschiedener Akteinkurse, in diesem Beispiel nur für die Aktie des Unternehmens, welches das EIP einsetzt, anzeigt.

Wenn der Portalbenutzer es wünscht, weitere Aktien in diesem Portlet dargestellt zu bekommen, muss er lediglich auf den „Edit“ Button in der Titelleiste über dem Portlet klicken um in den Editiermodus diese Portlets zu gelangen, wo er weitere Einstellungen vornehmen kann.

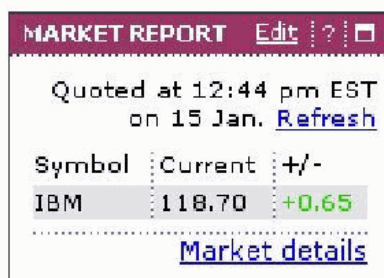


Abbildung 26: Beispiel eines Portlets

Um herauszufinden wie das Portlet funktioniert, muss er auf den „help“ Button klicken. Symbolisiert durch das „?“ . Um die Ansicht zu vergrößern muss er auf den „Vergrößerungs-Button“ klicken, der durch ein aus allen Windows Anwendungen bekanntem Symbol dargestellt wird.

Jedes dieser unterstützten Modi eines Portlets kann aus weiteren Ansichten bestehen. In unserem Beispiel liefert der Ansichtmodus zusätzlich zu dem aktuellen Aktienkurs noch Marktinformationen zu der jeweiligen Aktie.

Das hier in *HTML* angezeigte Portlet kann ebenso mit anderen Sichten verwendet werden. So kann ebenso mit diesem Portlet eine *WAP*, *PDA* oder Voice Ausgabe erzeugt werden. Typische Portal Server Implementationen beinhalten eine Request- (Anfrage) erhaltende und Personalisierung Komponente. In dieser Komponente laufen die Portlets und gleichzeitig liefert sie den für die Portlets benötigten Kontext.

Abb. 27 stellt dar, wie Portlets von einem J2EE basiertem Portal aufgerufen werden. Angenommen dass in dem in Abb. 25 dargestellten Beispiel der Benutzer auf den Marktinformations- (Market details) link innerhalb des Market-Report-Portlets klickt. Der Browser generiert daraufhin eine HTTP-Anfrage an den Portal Server.

Das Portal Servlet ruft Code auf, welcher die Portlets, die für den Benutzer dargestellt werden sollen, bestimmt. Weiterhin verschickt es die Anfrage zu der Zielaktion des Markt-Report Portlets um anschließend die nötigen Portlets und ihren Inhalt in einer einzelnen *HTML*-Seite darzustellen.

Das Portal Servlet ruft Code auf, welcher die Portlets, die für den Benutzer dargestellt werden sollen, bestimmt. Weiterhin verschickt es die Anfrage zu der Zielaktion des Markt-Report Portlets um anschließend die nötigen Portlets und ihren Inhalt in einer einzelnen *HTML*-Seite darzustellen.

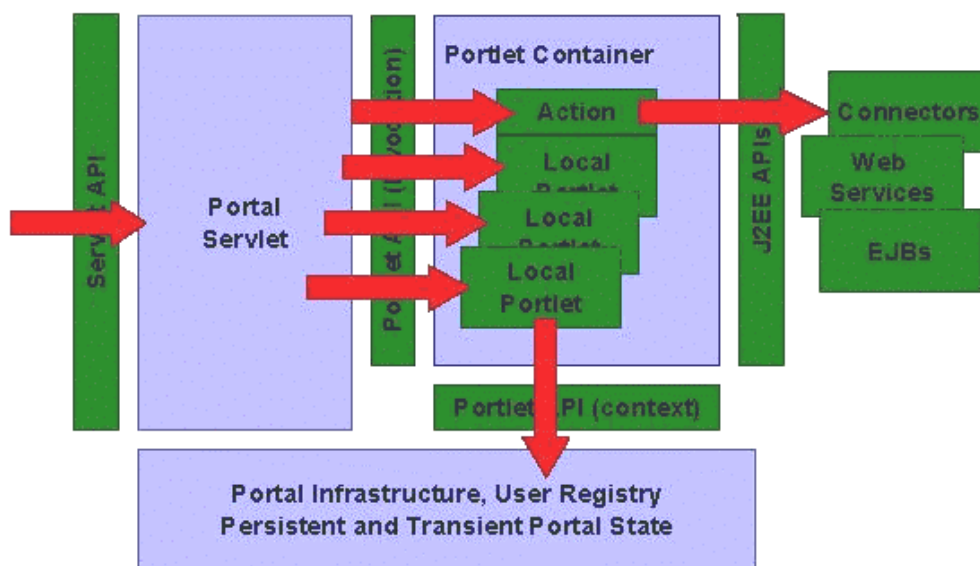


Abbildung 27: Aufruf von Portlets⁷

Portlets sind genau wie Servlets Webkomponenten, aber verfügen über zusätzliche Eigenschaften, die es ihnen erlauben einfach in eingeschlossenen Web-Applikationen wie Portalen eingesetzt zu werden. Portale sind dazu bestimmt in breitem Kontext von verschiedenen Seiten zusammengefügt zu werden. Zum Beispiel: Verschiedene Instanzen desselben Portlets, parametrisiert mit verschiedenen per-Benutzer, per-Instanz Portletdaten können auf derselben Seite koexistieren. Für gewöhnlich werden verschiedenen Portlets für die

⁷ in einem auf J2EE basierten Portal Server

Bearbeitung einer einzigen Anfrage aufgerufen, um die passenden Fragmente in der Portalseite zu erzeugen. Diese Fragmente sind oft mit Triggern (Auslöser) verknüpft, die dann Aktionen im Portlet auszulösen. Aus diesem Grund sind *URL*-Überschreib Methoden oft nötig, die es den Portlets ermöglichen, transparente Links innerhalb der Fragmente, die sie ausgeben, zu erzeugen. Dies muss allerdings ohne das Wissen, wie die *URL* in der jeweiligen Web Anwendung strukturiert ist, geschehen. Intern können die Portlets auf verschiedene Art und Weise implementiert werden. Zum Beispiel können sie nach dem „Model-View-Controller“ Muster implementiert werden, wo das Model und der Kontroller Javaklassen sind und bei den Views es sich um *JSP*'s, *XSLT*-Style sheets oder andere Arten von Templates (Vorlagen) handelt.

Portlets basieren auf Containerinfrastruktur die per Portlet-API für verschieden Funktionen zugänglich ist. Dabei handelt es sich um Funktionen wie:

- Zugänge zu:
 - Benutzerprofilinformationen des aktuellen Anwenders
 - Windows Objekten, die das aktuelle Fenster in dem das Portlet dargestellt wird beschreiben
 - Web Client Informationen
- Beteiligungen im Portalfenster und im „Action Event Model“
- „inter-portlet messaging“
- Standardweg um persistente User- oder Instanzdaten zu speichern und zurückzuholen

2.9.2 Die Notwendigkeit von Standards

Der Mangel von Standards im Portal Bereich hat viele Anbieter für die Portalplattform dazu gezwungen eigene APIs für die lokalen Portalkomponenten sowie für die Aufrufe von entfernten Komponenten zu definieren. Die daraus resultierende Vielfalt an Schnittstellen führt zu Problemen in der Interoperabilität bei den Kunden, Anwendungs- und Inhaltanbietern sowie den Portal Softwareherstellern. Im Folgenden werden einige dieser Gründe aufgeführt:

- Hersteller von Anwendungen die ihre Produkte mit der Portletfunktionalität ausstatten möchten um diese auch über Portale zugänglich zu machen, müssen dafür Sorge tagen, dass ihre Produkte in verschiedenen Versionen für die verschieden *APIs*

der Portal Server Plattformen verfügbar sind. Das hat allerdings erhöhte Entwicklungskosten für die Hersteller zufolge.

- Kunden die ihre eigenen Portlets entwickeln möchten, müssen beträchtlichen Aufwand betreiben um einen Portletcode zu schreiben der allerdings nur auf der Portalplattform lauffähig ist, auf der er auch entwickelt wurde. Somit neigen die Kunden dazu sich auf einer bestimmten Portalplattform einzukesseln, was es sehr schwer macht das bestehende Portal bei möglichen späteren Migrationen auf andere Plattformen zu übernehmen.
- Herstellern von Inhalten die speziell für Portale geschrieben sind, ist es nicht möglich ihre Inhalte in der Form zu veröffentlichen, dass der Kunde sich seine gewünschten Inhalte verschiedener Hersteller auswählen und per „Plug and Play“ in sein Portal einbinden und somit sofort nutzen kann.
- Kleinere Portal Softwareanbieter haben Schwierigkeiten die Unterstützung ihrer eigener Standards bei den Anwendungsherstellern durchzusetzen.

Standards existieren für die Client-to-portal Protokolle sowie die verschiedenen Auszeichnungen. Geräte oder Gateways greifen auf Portale typischerweise über Standard-*HTTP* zu und abhängig vom Gerättyp über verschiedene andere standardisierte Auszeichnungstypen wie *HTML* für die gängigen Webbrowser, *WML* für *WAP*-Handys, *VoiceXML* für Zugang über das Telefon mit Hilfe der Stimme und einem Voice gateway, etc.

Ebenfalls gibt es bereits Standards um Web Services zu beschreiben (*WSDL* – Web Services Description Language), um Web Services in Verzeichnissen (*UDDI* - Universal Description, Discovery & Integration) zu veröffentlichen und aufzufinden aber auch um diese aufzurufen (*SOAP* - Simple Object Access Protocol). In Abb. 28 sind diese Standards grün dargestellt. Siehe dazu auch Kap. 2.8.1.1.

Die zusätzlich benötigten, aber gegenwärtig fehlenden Portalbezogenen Standards für Portal Komponenten sind folgende: für lokal am Portal Server eingesetzte Komponenten (*Portlet-API*) und der Standard für entfernte Komponenten die per „Plug and Play“ mit dem Portal Server interoperabilieren (*WSRP*). In Abb. 28 sind diese gegenwärtig fehlenden Standards rot markiert.

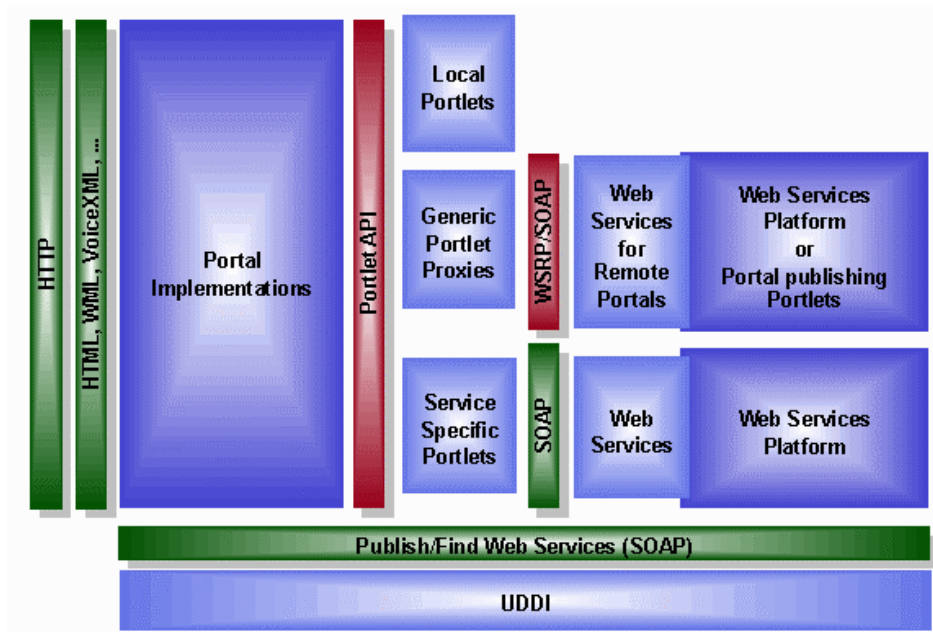


Abbildung 28: Existierende (grün) und fehlende (rot) Portalstandards

In Abb. 29 wird das Ergebnis des Fehlens von gemeinsamen Portlet-API-Standards für Portale erläutert.

Der Anwendungsanbieter, deren Software auch Portlets unterstützen soll, wählt üblicherweise einige gängige Portal Plattformen aus und muss für diese auch mehrere Version eines einzelnen Portlets implementieren. Aus der Portal Server Sicht laufen nur Untergruppen eines Portlets auf jeder individuellen Portal Server Plattform. Portalkunden können damit in Situationen geraten, in denen die von ihnen benötigten Portlet-Kombinationen nicht für ihren Portal Server verfügbar sind.

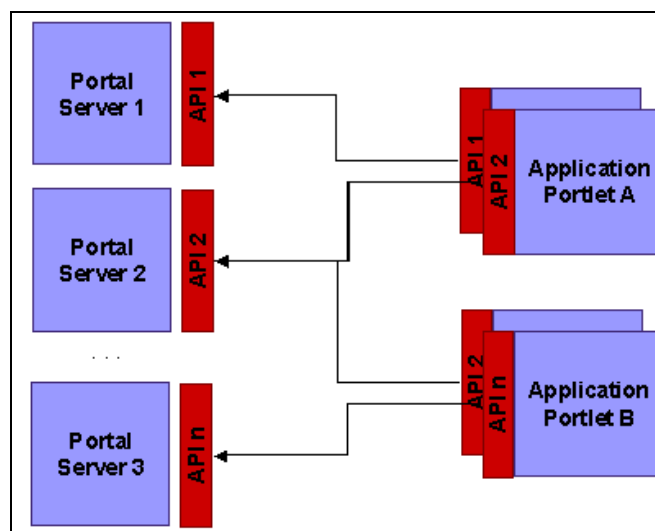


Abbildung 29: Vielfalt von Schnittstellen für Portal Komponenten

Die Java-Portlet-API soll diesem Problem beikommen durch Interoperabilitätsaufbau zwischen Portlets und dem Portal. Wie in Abb. 30 dargestellt, laufen die der Portlet-API konformen Portlets auf allen der API gefälligen Portal Servern.

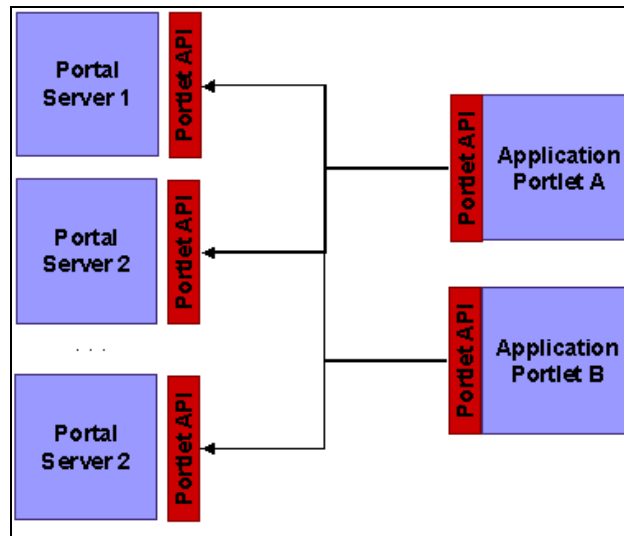


Abbildung 30: Gemeinsame Portlet API für Portal Komponenten

Ähnlich wird auch der WSRP-Standard die Interoperabilität zwischen Portalen und WSRP-unterstützenden, visuellen Benutzer gegenüberstehenden, Web Services für Portale ermöglichen.

Das bedeutet, dass Portal-Software-Anbieter in der Lage sein werden, Portal Server anbieten zu können, die alle Arten von Portlets und Web Services von unterschiedlichen Herstellern ohne weiteres in sich vereinigen können. Entweder lokale Portlets der Java-Portlet-API entsprechend oder dem *WSRP* gemäßen Web Services.

Andererseits werden *ISV*⁸, Anwendungshersteller, Kunden und Inhaltsanbieter in der Lage sein, Portlets und Web Services implementieren zu können, die in allen Portalen genutzt werden können.

⁸ Independent Software Vendor

2.9.3 Java Portlet API

Die Java Portlet *API* definiert eine Java *API* für Web Anwendungskomponenten die mit Portalen interagieren und in diese integriert werden können. Im Weiteren werden diese Komponenten als Portlets bezeichnet. Die Java Portlet *API* wird in dem Java Community Process als „*JSR 168*“ beschreiben und definiert. [JCP]

Siehe dazu auch Kap. 6.11

Die Portlet API soll Standards für die oben erwähnten Funktionen definieren.

Sie wird ebenfalls Schnittstellen für Portlets und *JSP*-Tags, die von Portlets aufgerufen werden, definieren und die Portlets sauber von der ihn umgebenden Portal Server Infrastruktur trennen, so dass die Portlets auf verschiedenen Portal Servern laufen können. Vergleichbar mit Servlets, die auch auf verschiedenen Application Servern laufen.

2.9.4 Web Services for Remote Portals (WSRP)

WSRPs sind visuelle, Benutzer gegenüberstehende Web Services, die durch „Plug and Play“ mit Portalen oder anderen Applikationen verwendet werden können. Sie sind so entworfen, um eine neue Sparte von Business zu ermöglichen, bei der Inhalte angeboten werden, die der Kunde ohne inhalts- oder anwendungsspezifischer Anpassung in sein Portal adoptieren kann.

Ebenso beinhaltet *WSRP* die Präsentationsdefinition, d.h. die *WSRP*-Service-Anbieter bestimmen wie ihre Inhalte und Anwendungen für den Endbenutzer visualisiert werden und welcher Grad von Anpassung, Transcodierung, Übersetzung, etc. erlaubt sein soll.

WSRP-Dienste können in öffentlichen- oder Unternehmensverzeichnissen (*UDDI*) veröffentlicht werden, wo sie von Portalen, die ihre Inhalte verwenden möchten, einfach gefunden werden können. Anwender von Webapplikationen können ihre Middleware anpassen, um sie so für den *WSRP* fähigen Service nutzbar zu machen. Anbieter von Vermittlungs-Anwendungen können ihren Produkten ermöglichen *WSRP* Services zu verwenden.

WSRP verwendende Portale können ganz einfach Inhalte und Anwendungen von internen und externen Anbietern integrieren. Der Portaladministrator kann den verlangten Dienst durch Auswählen aus einer Liste, ohne programmieren, einfach in das Portal einbinden.

Um diese Ziele zu erreichen, wird der *WSRP* Standard eine Web-Service-Schnittstellenbeschreibung definieren die *WSDL*, die ganze Semantik und das Verhalten der Web Services und der sie benutzender Anwendungen, verwendet. Somit wird sichergestellt, dass diese Schnittstelle kompatibel sein wird. Ebenfalls wird sie die Metainformationen definieren, die bei der Publikation in ein *UDDI* Verzeichnis vorhanden sein müssen. Der Standard erlaubt es *WSRP* Diensten auf verschiedene Art und Weise implementiert zu werden. Dabei kann es sich um auf *Java/J2EE* oder einer *.NET* Plattform basierende Web Services handeln oder um ein Portlet, das als *WSRP* Service durch ein Portal veröffentlicht wird. Der Standard ermöglicht die Nutzung von generischen Adaptercodes, um jeden *WSRP* Service an eine dazwischenliegende Anwendung anzudocken, was viel mehr als eine Anforderung eines bestimmten Proxycodes bedeutet. [OASIS]

2.9.5 Zusammenhang zwischen Portlet-API und WSRP

Die Portlet *API* und der *WSRP* werden auf nützliche Weise zusammenarbeiten. *WSRP* Dienste können in Portale durch Portlets-Proxies integriert werden, die für die Portlet *API* geschrieben wurden. Umgekehrt können Portlets in ein *WSRP* Service eingehüllt und als solcher veröffentlicht werden.

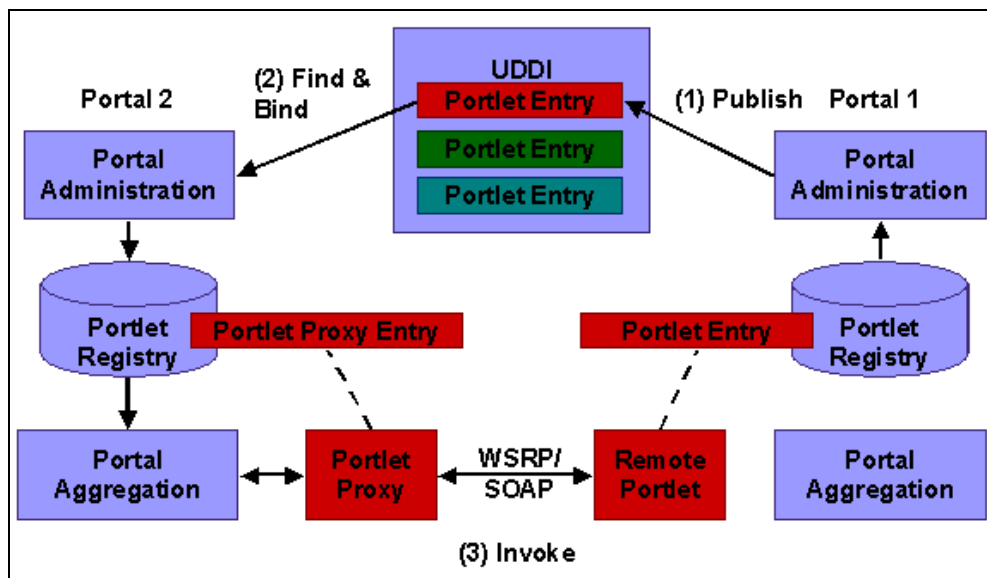


Abbildung 31: Portlets als WSRP Dienste in Portalen einbinden

Abbildung 31 zeigt auf, wie ein Portal Server ein Portlet als *WSRP*-Service in einer *UDDI*-Registry veröffentlicht, so dass andere Portale es ebenfalls finden und einbinden können. Üblicherweise besitzt ein Portal lokale Portlets, die in der Portlet Registry aufgelistet sind.

Um ein Portlet als ein *WSRP* Dienst verfügbar zu machen, muss das Portaladministrations-User-Interface dem Administrator es ermöglichen, Portlets auszuwählen und diese in einem *UDDI*-Verzeichnis veröffentlichen zu können. Ein Adapter gibt das Portlet extern auf, indem es dieses mit den im *WSRP* definierten *SOAP* Operationen versorgt.

Ist das Portlet einmal in dem *UDDI* Verzeichnis aufgeführt, können es andere Portale auffinden und die referenzierten *WSRP*-Dienste einbinden. Um einen *WSRP*-Dienst als Portlet verfügbar zu machen, muss das Portaladministrations-UI einen Portlet Proxyeingang in der lokalen Portlet Registry mit den Informationen, die es von der *UDDI* erhalten hat, erzeugen. Generische Portletproxies umhüllen die *WSRP*-Dienste.

Wenn sich der Portletproxy-Eingang einmal in der lokalen Portletregistry befindet, können die Benutzer sie zum Beispiel anwählen und Teile davon in ihre Portalseiten verwenden. Wird die Portletproxy-Instanz während einer Seitenanhäufung aufgerufen, ordnet der Portletproxy die Methodenargumente in eine *SOAP*-Anfrage (dem *WSRP* Standard entsprechend) und sendet diese an den *WSRP* Dienst. Danach bekommt er die *SOAP* Antwort von dem *WSRP*-Dienst und strukturiert die Ergebnisse für das Portal entsprechend wieder um.

2.9.6 Schlussfolgerung

Die Java Portlet *API* und die Web Services for Remote Portlet (*WSRP*) Standards werden die Interoperabilität zwischen dem Portal Server, lokalen- und entfernten Portlets, visuellen und Benutzer gegenüberstehenden Web Services ermöglichen. Das äußerste Ziel dieser Standards ist es, eine große Anzahl von Portlets verfügbar zu machen, die auf jeden Portal Server, egal ob lokal oder als entferntes Web Service, lauffähig sein sollen. Dies wird den Portalkomponenten Markt fördern auf den eine große Vielfalt von Anwendungsanbietern, unabhängigen Softwareherstellern (*ISVs*) und die Open Source Gemeinde, die alle bereitwillig Portlets oder visuelle Web Services zur Verfügung stellen.

2.9.7 Einsatz der Standards bei Portalsoftware Herstellern

Bei den im Kapitel 2.8 und folgenden vorgestellten Standards für Portale und Portlets steht eine offizielle Verabschiedung durch des Konsortiums *OASIS* und der offenen Organisation von Javaentwicklern *JCP* (Java Community Prozess) noch aus. Deshalb kann kein EIP-Software-Hersteller wie IBM, Plumtree, Microsoft oder Oracle von sich behaupten, diese Standards bereits erfüllen zu können.

Im Rahmen einer Analyse des Produkts Oracle Portal wird im Kapitel 6.11 genauer auf die Zusammenarbeit mit den Oracle Portal Entwicklerteam und den Standardisierungs-Konsortien eingegangen.

3 Konzept für ein EIP am Bsp. der FH-Gummersbach

In diesem Kapitel wird die Funktionsweise eines Enterprise Information Portals anhand eines praktischen Beispiels verdeutlicht. Es soll im Folgenden geschildert werden, wie durch eine Analyse eines bestehenden Zustandes und Aufgaben, der Bedarf für den Einsatz eines EIP's erkannt werden kann.

Da der Aufbau eines voll funktionsfähigen Portals den Zeitraum dieser Arbeit sprengen würde, können hier nur theoretisch die Aufgaben des Portals aufgezeigt werden. Allerdings werden Funktionen angesprochen, die bei einer zukünftigen „go live“ Implementation zum Einsatz kommen könnten.

Zur Verdeutlichung der EIP-Funktionalitäten wird ein Portal für die Fachhochschule Köln, Campus Gummersbach prototypisch erstellt und Beispiele für Anwendungen genannt, die bei einer möglichen, späteren Realisierung eines FH-Portals als ein Grundgerüst verwendet werden können.

Die Anforderungen an das Hochschulportal der Fachhochschule Köln, Campus Gummersbach sind mit einzelnen Anforderungen an ein Unternehmens EIP vergleichbar.

Aus diesem Grund werden anhand der bestehenden FH-Homepage Aufgaben für ein zukünftiges EIP definiert. Zusätzlich werden Rollen für die verschiedenen Anwender erläutert, die in dem Portal verschiedene Zugriffsrechte und Sichten haben werden.

3.1 Definition der FH-Köln, Campus Gummersbach-Homepage

Bevor Beispiele für Anwendungen und Funktionen gegeben werden können, müssen die Anforderungen und Rollen definiert werden. Somit können einzelnen Anwendungen an die einzelnen Rollen und Verwendungsarten der Benutzer angepasst werden.

3.1.1 Aufgaben der bisherigen FH- Homepage

Der bisher bestehende Webauftritt der Fachhochschule Köln, Campus Gummersbach hat einen reinen Informationscharakter für den Besucher. Er kann den gewünschten Fachbereich auswählen, und gelangt dadurch zu den jeweiligen Seiten dieser.

Dabei stehen folgende Fachbereiche zur Auswahl: Informatik, Maschinentechnik und Elektrotechnik.

Die einzelnen Seiten enthalten die gleichen Art von Inhalten, die lediglich auf den jeweiligen Fachbereich angepasst wurden. Wie:

Aktuelles, Allgemeine Informationen, Studienangebot, Personen, Labore, Forschungsprojekte, Kontakte und Links.

Der Zweck dieser Seiten ist es den neuen Studenten oder am Studium interessierte Besucher über das jeweilige Institut zu informieren und Tipps zum Studienverlauf zu geben.

Unter „Aktuelles“ sind oft Termine, Terminänderungen oder Vorlesungsausfälle zu entnehmen. Diese werden allerdings von den einzelnen Webmastern gepflegt, was nicht immer für eine hohe Aktualität spricht. Über den Link „Personal“ und weiter über den gewünschten Dozenten oder Angestellten gelangt der Besucher zu der persönlichen Homepage der ausgewählten Person. Dort findet er Informationen zur Vorlesungen und zur Praktika des jeweiligen Dozenten, sofern dieser über eine eigene Homepage verfügt. Dort kann er Vorlesungsskripte und Praktikumsunterlagen downloaden, wenn der Dozent diese auf seiner Seite zum Download anbietet. Nur in sehr wenigen Fällen sichern die Dozenten ihre Downloads über ein Passwort, welches sie nur den an ihrer Vorlesung oder Praktikum teilnehmenden Studenten bekannt gegeben haben. Damit kann jeder Seitenbesucher an Studienunterlagen gelangen, der sich die Mühe macht, die einzelnen Homepages durchzusuchen.

3.1.2 Benutzer

Als Anwender der FH-Homepage können folgende genannt werden:

- Professoren
- Dozenten
- Angestellte
- Studentischen-Hilfskräfte
- Studenten
- Besucher (Externe Besucher der FH-Homepage)

3.2 Inhalte und Rollen des FH-EIP's

Das Portal der Fachhochschule Köln, Campus Gummersbach (folgend FH-Portal) soll primär den unter 3.1.2 definierten Besuchern einen zentralen Informationspool zum Thema Studium an der FH-Gummersbach bieten. Für angemeldete Benutzer sollen zusätzliche Funktionen und Informationen geboten werden. Diese Funktionen werden von der jeweiligen Rolle des FH-Portal Benutzers abhängig sein.

Im Rahmen dieser dreimonatigen Arbeit wird es nicht möglich sein, das FH-Portal zu entwickeln und mit funktionsfähigen Anwendungen auszustatten. Die hier erwähnten Vorschläge für Applikationen haben das Ziel, den Portalbenutzer mit sinnvollen Anwendungen zu unterstützen und das Portal somit für ihn attraktiv zu machen.

Zum erfolgreichen Aufbau eines EIP's sind die in Kap. 3.1 definierten Aufgaben und Benutzer enorm wichtig, da dadurch der Informations- und Ressourcenbedarf der einzelnen Rollen definiert werden kann. Im Folgenden wird der Inhalt des EIP's dargestellt, indem den einzelnen Rollen Informationen und Ressourcen zugeordnet werden.

Auf die konkrete Umsetzung des FH-Portals wird im Kapitel 5 dieser Arbeit eingegangen. Es wird eine prototypische Lösung mit dem Produkt Oracle Portal 3.0 vorgestellt.

3.2.1 Rollen im FH-Portal

Es können aus den aktuellen Anwendern folgende Rollen für Gruppen des FH-Portal definiert werden.

Dabei ist festzuhalten, dass alle hier erwähnten Gruppen über Ansichtrechte der Hauptseiten des FH-Portals verfügen. Sie besitzen einen eigenen Usernamen und ein Passwort, mit dem sie sich innerhalb des FH-Portals authentifizieren können (ausgenommen Besucher).

Professoren und Dozenten:

Die Anwender in dieser Gruppe sollen über jeweils eine eigene Homepage verfügen, auf der sie Ihre eigenen Anwendungen und Inhalte veröffentlichen können. Zusätzlich können sie über Administrationsrechte für Seiten der verschiedenen Bereiche der Fachhochschule verfügen.

Mitarbeiter und Angestellte

Die Angehörigen dieser Gruppe sollen über editier-Rechte auf der Hauptseite der jeweiligen Abteilung oder Bereichs verfügen können, für den sie tätig sind.

Studentischen-Hilfskräfte:

Ähnlich wie die Mitarbeiter, sollen sie über editier-Rechte der Bereiche verfügen, für die sie tätig sind. Gekennzeichnet durch weniger Rechte und begrenzte Dauer des User-accounts (z.Bsp. ein Semester)

Studenten:

Studenten bekommen Zugriffsrechte von dem jeweiligen Professor auf einzelne Seiten und Inhalte zugewiesen. Des Weiteren können Ihnen Administrationsrechte auf Seiten gegeben werden, die mit Projekten oder Studiums-Aufgaben zusammenhängen.

Besucher:

Verfügt lediglich über Ansichtrechte der Hauptseiten. Kein Einloggen nötig.

3.2.2 Inhalte im FH-Portal

Die im Kapitel 3.2 definierten Inhalte müssen so umgesetzt werden, dass sie von den einzelnen Anwendern (Rolleninhabern) über einen zentralen Zugangspunkt mit dem Webbrowser erreichbar sind. Dabei ist der Sicherheitsaspekt nicht außer Acht zu lassen.

Für jede Rolle werden im folgenden Portalinhalte vorgestellt, die beim Umgang mit dem EIP nützliche Funktionen bieten sollen. Verweise auf im Kap. 5 realisierte Beispielanwendungen werden in („Klammern“) angegeben.

Professoren und Dozenten:

In erster Linie dient die persönliche Professoren- / Dozenten- Seite dazu, die Studenten über die Vorlesungen und Fächer zu informieren. Sprechstundenzeiten und weitere Neuigkeiten zur Person oder zur Vorlesung werden hier veröffentlicht. Des Weiteren können Inhalte die der Vorlesung dienen (Skripte, Aufgabenblätter, Programme, etc.) in persönlichen Content Areas abgelegt und den berechtigten Studenten somit (als Download) zur Verfügung gestellt werden. (Bsp. siehe Kap. 5.3)

Der Professor oder Dozent könnte sich eine Liste aller, an seiner Vorlesung teilnehmender Studenten ausgeben lassen. (Bsp. siehe Kap. 5.4.1.4) Diese könnte er weiter verwenden, um zum Beispiel Projektgruppen einteilen zu können.

- Anmelde Listen für Praktika und Projekte können erstellt werden, in die sich Studenten über das Internet eintragen können. Es können verschiedene Datenbanktriggern aktiviert werden, die bei begrenzter Teilnehmerzahl weiteren Anmeldungen verhindern oder Doppelanmeldungen unterbinden.

Es besteht die Möglichkeit, Klausurergebnisse einzugeben, die dann im Prüfungsamt abgerufen und am Aushängebrett veröffentlicht werden können.

Mitarbeiter und Angestellte

Mitarbeiter in der Verwaltung können mit dem Portal Studenten-, Mitarbeiter- und Professorendaten erfassen und in der Datenbank speichern. (Bsp. siehe Kap. 5.4.1.1)

Diese können für statistische Zwecke durch verschiedene Auswertungstools verwendet werden.

Es könnten zum Beispiel Studentenanzahl, getrennt nach Geschlecht, Studiengang, Studiendauer, etc. angezeigt und ausgewertet werden. (Bsp. siehe Kap.5.4.1.2)

Daten zu den einzelnen Fächern, Räumen und Laboren können von den Angestellten eingegeben und gepflegt werden. Dadurch lassen sich Stundenpläne und Raumbellegungspläne erzeugt.

Studentischen-Hilfskräfte:

Von ihnen könnten beispielsweise die Homepages der Professoren gepflegt werden. Informationen zu den Fächern, Terminänderungen oder Ausfälle könnten in einem XML-Portlet (Bsp. siehe Kap. 5.4.1.5) auf der Professorensseite angezeigt werden und zusätzlich in der Hauptseite der Fachbereichs, ohne dass die Inhalte neu eingefügt werden müssen.

Studenten:

Studenten können Vorlesungsskripte oder zusätzliche Informationen bei den Professoren downloaden. Sie haben die Möglichkeit sich bei Projekten und Praktika des jeweiligen Professors persönlich anzumelden.

Sollten alle Datenschutzfragen gelöst werden, könnte die Möglichkeit gegeben werden, dass die Studenten ihre eigenen Prüfungsergebnisse abrufen oder sich ihren Notenspiegel anzeigen lassen können.

Besucher:

Ihnen dient das FH-Portal als Informationsquelle zum Studium und Einrichtungen der Fachhochschule. Sie können sich hier Anmeldeformulare herunterladen oder über die Studiengänge informieren.

Weitere Anwendungen im FH-Portal

Routenplaner, Suchfunktion (z.Bsp. nach Professoren Namen, Telefonnummern, Raumnummern, etc.), Bücherverfügbarkeit in der Hochschulbibliothek prüfen (z.Bsp. als Web Services zu realisieren)

4 Oracle Portal

Für die Umsetzung des im Kapitel 3 vorgestellten FH-Portals könnte das Produkt Oracle Portal verwendet werden. Um die Möglichkeiten dieser Software kennen zu lernen, wird das Produkt in diesem Kapitel vorgestellt. Als Basis für folgende Übersicht von Oracle Portal dient eine Eigenanalyse sowie eine Produktbeschreibung in englischer Sprache. [OPF300]

4.1 Abstrakt

Das Produkt Oracle Portal ist ein browserbasiertes Entwicklungstool für den Aufbau und die Entwicklung von EIP's. Es ermöglicht mit seinem erweiterbaren Framework den Zugang zu verschiedenen Softwarediensten und Informationsquellen. Die Oberfläche des Oracle Portal ermöglicht eine personalisierte und organisierte Sicht auf Informationen aus dem Intra-, Extra und Internet, auf Geschäftsinformationen und Anwendungen. Dem Anwender wird durch das „Self-service-publishing“ ermöglicht, Informationen zu veröffentlichen und mit anderen Portalbenutzern weltweit zu teilen. Die Administration sämtlicher Portaldienste erfolgt ebenfalls über den Webbrowser.

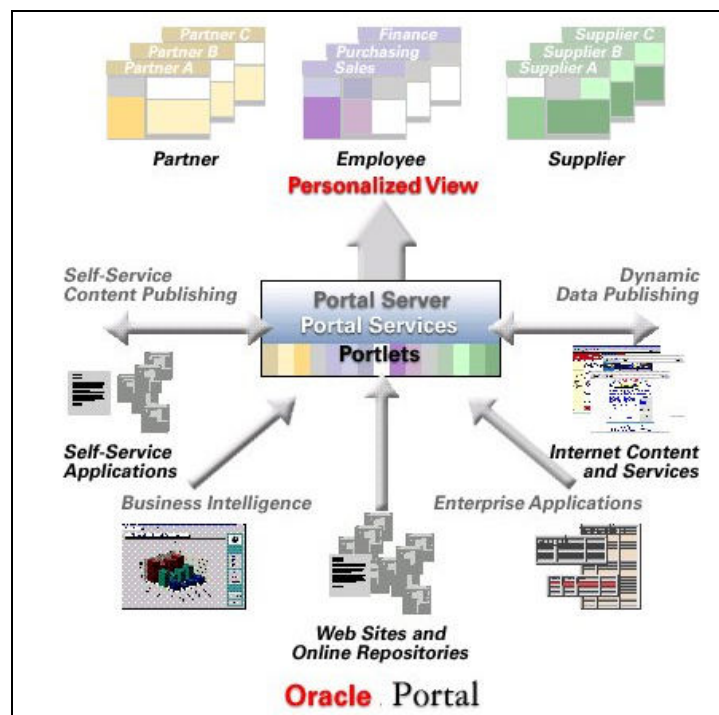


Abbildung 32: Architektur und Dienste in vereinfachter Darstellung

Neben den Kernanwendungen, der Portalentwicklung und dem Datenmanagement, bietet Oracle Portal mit seinem erweiterbaren Framework Integration von Anwendungen und Informationen.

Diese Frameworks basieren auf der von Oracle entwickelten Portlet Technologie.

Bei Portlets handelt es sich um universell verwendbare Komponenten, die personalisierbar sind und Zugriffe auf Webseiten, Anwendungen und diverse Informationsressourcen ermöglichen. Unternehmen haben die Möglichkeit ihre eigenen Portlets zu entwickeln oder auf Portlets von verschiedenen Fremdanbietern zu zugreifen.

Mit Portlets wird erst der Zugang zu den webbasierten Anwendungen ermöglicht.

Das Oracle Portal-Frameworks bietet zusätzliche Dienste, wie Single Sign-On, Inhaltsklassifikation, Suchfunktion, Verzeichnisintegration, Userverwaltung und Sicherheitsfunktion an.

Das Produkt Oracle 9iAS Portal (Release 3.0.9) ist ein Zusatzprodukt zum Oracle Application Server (iAS) und baut auf die Oracle 8i Datenbank auf.

Das Produkt ist für folgende Plattformen verfügbar:

- Compaq Tru64 UNIX
- HP HP-UX (64-bit O/S kernels only)
- IBM AIX (64-bit O/S kernels only)
- Linux Intel
- Solaris SPARC
- Windows NT/2000

In dem folgenden Abschnitt werden die Fähigkeiten und Merkmale des Oracle Portal 9iAS (3.0.9) aufgezeigt

4.2 Portal Framework

Als Grundlage für den Aufbau eines EIP's dienen die Portal Frameworks. Sie ermöglichen die Erstellung und Verwaltung von Portalinfrastrukturen, die sich aus unterschiedlichen Diensten und Informationen zusammensetzen und über Portlets zugänglich und darstellbar gemacht werden.

4.2.1 Portlets

Bei einem Portlet handelt es sich um ein standardisiertes Objekt, durch das der Zugang zu den Informationsressourcen ermöglicht bzw. vereinfacht wird. Durch das Portlet werden die Informationen in vorbestimmten Feldern zusammengefasst dargestellt.

Portlets sind vielseitig einsetzbare Objekte und können zum Beispiel als ein Link zu einer URL dienen, als Zugang zu internen und externen Anwendungen, als Web-Anwendungen oder als Anzeigefeld für verschiedene Informationen in unterschiedlichen Webformaten (HTML, XML/XSL). Der Begriff „verkapseltes Informationsmodul“ welches Informationen anschaulich darstellt, wird im Zusammenhang mit Portlets auch öfters verwendet.



Abbildung 33: Beispiel eines Portlets, zum abrufen von Fluginformationen

Eine Portalseite entsteht in der Regel erst aus der Zusammenführung verschiedener Portlets auf einer Webseite.



Abbildung 34: Beispiel Portalseite

Die Seite setzt sich aus mehreren Portlets zusammen:

- 1) Loginportlet, 2) Contentarea mit Links als Inhalt, 3) Datenbankzugriffs-Portlet
- 4) Contentarea mit Grafik als Inhalt, 5) HTML-Portlet, 6) Suchportlet

Jedes Portlet gehört zu einem Portlet Provider (Anbieter), über den die Kommunikation mit dem Oracle Portal erst ermöglicht wird. Ein Portlet Provider ist innerhalb einer Instanz der Oracle Portal definiert, registriert und dort eindeutig identifizierbar. Er kann ein oder mehrere Portlets zur Verfügung stellen. Erst nach einer Registrierung der Portlet-Provider innerhalb der Oracle Portals kann auf seine Portlets zugegriffen werden.

Jedes Portlet kann einer Applikation zugeordnet werden, mit der es erstellt wurde. Eine Applikation kann mehrere verschiedene Komponenten und Portlets enthalten. Die Zugriffsrechte erbt das Portlet entweder von der Applikation oder sie werden über die Zugriffsverwaltung des Portlets vergeben.

Die meisten Portlets unterstützen die Personalisierungsfunktion. Das bedeutet, dass ein Benutzer das Portlet nach seinen individuellen Bedürfnissen einstellen kann. Beispielsweise kann er bestimmen, ob in der Kundenliste die Kundennamen auf- oder absteigend dargestellt werden sollen.

Oracle Portal selbst ist als webbasiertes Entwicklungstool ebenfalls über Portlets bedienbar. So verfügt es unter anderem über Portlets für die:

- Anwenderverwaltung
- Gruppenverwaltung
- Schemenverwaltung
- Rollenverwaltung
- Portletverwaltung
- Navigation
- Suchdienste

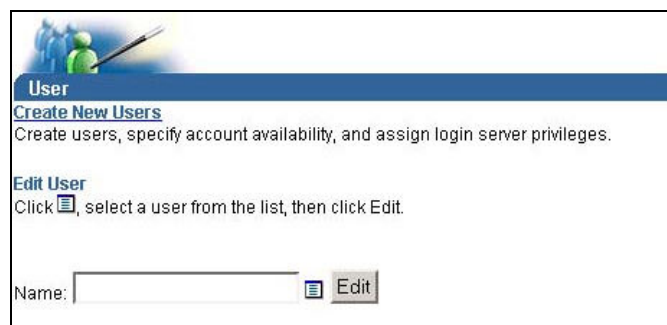


Abbildung 35: Portlet für Benutzerverwaltung innerhalb des Portals

Weitere im Oracle Portal bereits vorhandene nach Bereichen sortierte Portlets sind:

Bereich Portlet Entwicklung:

- Erstellung und Verwaltung von Portalseiten
- Inhalt und Anwendungsverwaltung
- Suchdienst für Objekte
- Anzeige der meist verwendeten Objekte
- Erstellung von Portlets aus *HTML*-Quellen

Bereich Endbenutzer:

- Speicherung und Wiederfinden von oft benutzten *URL*'s
- Veröffentlichung von Informationen / Verwaltung (self-service publishing, self content management)
- Suchdienst
- Zugang zu persönlichen Verzeichnissen, externen Anwendungen

Bereich Portal Administration:

- Benutzerkennung-Verwaltung (user account)
- Benutzer und Gruppenverwaltung
- Portal Einstellungen
- Verwalten von Portlet-Anbietern (lokal und remote)

Des Weiteren sind noch Überwachungs- und Datenbankportlets verfügbar.

Siehe dazu Kap. 4.5.3 und 4.5.4.

Sollten die vorgegebenen Portlets für ein Unternehmens EIP nicht ausreichen, oder nicht die gewünschten Funktionalitäten zur Verfügung stellen, so bietet das Oracle Portal auch die Möglichkeit der individuellen Portletentwicklung.

Hierbei ist es möglich Berichte, Diagramme, *HTML*-Seiten, Links, Menüs und Formulare mit den *BI*-Tools von Oracle Portal relativ einfach zu erstellen und als Portlet zu veröffentlichen. Sollte diese Möglichkeit dennoch nicht zum gewünschten Portlet für das Unternehmen oder den Anwender führen, besteht zusätzlich noch die Möglichkeit die Portlets „eigenhändig“ zu programmieren. Als Programmiersprache stellt Oracle Portal *PL/SQL* oder *Java* mit einer eigenen *API*, dem *JPDK* (Java Portlet Development Kit) zur Verfügung.

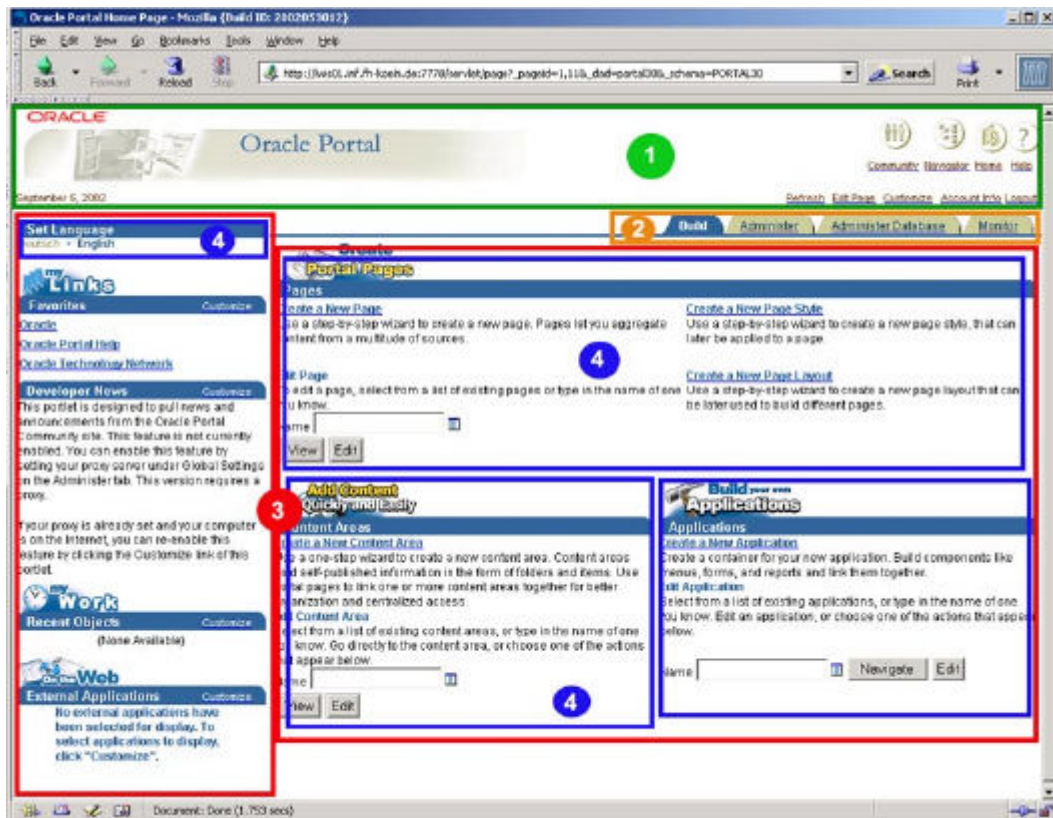


Abbildung 36: Defaultansicht für den Administrator⁹

4.2.2 Portalseiten (Pages)

Bei Portalseiten handelt es sich um Webseiten, in denen ein oder mehrere Portlets dargestellt werden. Diese Seiten werden dynamisch erzeugt, je nach Inhalt und Layout. Eine Portalseite wird durch ihr Layout, Seitenbanner, Seiteninhalt und ihre Sicherheit definiert.

4.2.2.1 Seitenlayout

Eine Portalseite wird dadurch charakterisiert, dass sie aus mehreren voneinander unabhängigen Regionen aufgebaut ist. Regionen sind die Flächen, in denen die einzelnen Portlets dargestellt werden.

⁹ Die Nummern kennzeichnen folgende Komponenten:

(1) Banner, (2) Tabs, (3) Regionen, (4) einzelne Administrations-Portlets

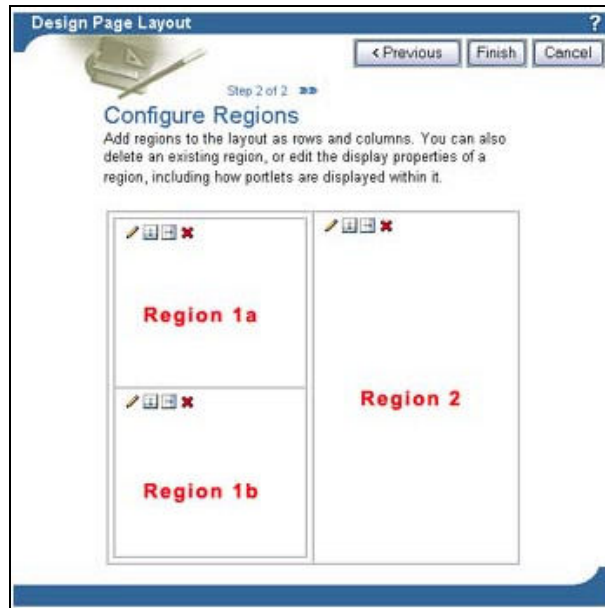


Abbildung 37: Menü zur Erstellung von Regionen¹⁰

Diese Regionen sind einzeln editierbar. Somit kann ein Portlet innerhalb der Region als Zeile oder Spalte angezeigt werden. Die Abstände zwischen den Portlets, der Kopfzeile können justiert werden. Auch der Rahmen der ein Portlet umrandet, kann ein- und ausgeblendet werden. Einen wesentlichen Teil zur leichten Übersicht trägt die Tabset-Funktion bei. Mit dem Tabsets können, vergleichbar mit Karteikarten, verschiedene Regionen übersichtlich dargestellt werden, die über den „Registerreiter“ angewählt werden und somit den Inhalt dieser Region für den Benutzer im Browser darstellen.

Ein erstelltes Seitenlayout kann als Template (Vorlage) gespeichert werden und somit beim Aufbau weiterer Seiten verwendet werden. Dieses Vorgehen trägt zum konsistenten Erscheinungsbild des EIP's bei. Um bei einer Portalseite die einzelnen Regionen auf einfache Weise mit Portlets zu „bestücken“, verfügt das Oracle Portal über eine „Add Portlet“ Funktion, mit der ein berechtigter Benutzer aus einem Portlet-Pool „Portlet Repository“ das gewünschte Portlet durch anklicken auswählt und somit der Region zufügt.

¹⁰ Region 1 ist in zwei Unterregionen unterteilt „a“ und „b“

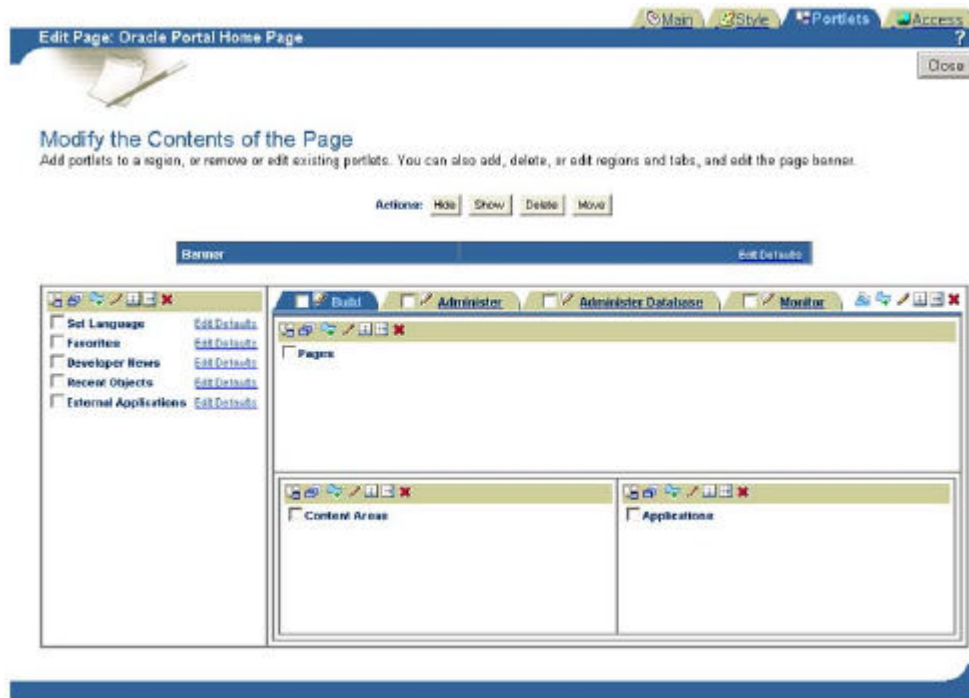


Abbildung 38: Menü zum editieren des gesamten Portalseite

4.2.2.2 Seitenstil

Der Stilmanager erlaubt es dem Benutzer verschiedene Formatvorlagen zu erstellen und zu speichern. Diese können für weitere Seiten verwendet werden und müssen nicht jedes Mal neu erstellt und angepasst werden. Dazu gehört die Erstellung von:

- Portlet Kopfzeilen
- Portlet Inhalten
- Überschriften
- Links
- Hintergrundschrift und -farbe

4.2.2.3 Seitenbanner

Der Seitenbanner als eine Standardkomponente der Portalseite dient hauptsächlich als visuelle Bereicherung dieser, indem hier eigene Logos und Grafiken, individuelle Begrüßungstexte oder Nachrichten eingefügt werden können. Des Weiteren können hier auch Links zu anderen Seiten oder Anwendungen eingesetzt werden. So ist z.B. der Login/Logout Link eine beliebte Bannerkomponente mit der sich ein Benutzer schnell am Portal an- bzw. abmelden kann. Abbildung 36 Punkt (1) zeigt das Defaultlayout des Seitenbanners, wenn keine Änderungen vorgenommen wurden.



Abbildung 39: Menü zur Editierung des Banners¹¹

4.2.2.4 Seitensicherheit

Der Benutzer, der eine Portalseite erstellt hat, ist gleichzeitig der Besitzer dieser und hat somit alle Rechte für ihre weiteren Änderungen und Anpassungen. Er hat gleichzeitig die Möglichkeit diese Seite für andere Anwender frei zu geben. Dazu stehen im Oracle Portal zwei Sichtdefinitionen zur Verfügung.

Privat: Bei „privat“ definierten Seiten ist nur die individuelle Anpassung sichtbar und sie können lediglich von am Portal registrierten Benutzern gesehen werden.

Öffentlich: Bei einer mit „öffentlich“ definierter Seite sind die Änderungen für alle Anwender sichtbar.

Neben dem Seitenersteller besitzen nur der Portaladministrator und der Datenbankadministrator (*DBA*) Rechte auf diese Seiten. Darüber hinaus besitzen die Administratoren alle Rechte auf sämtliche Seiten.

Oracle Portal bietet den Erstellern und Administratoren die Möglichkeit, Rechte auf bestimmte Anwender und Handlungen zu beschränken. Dazu stehen folgende Privilegien zur Verfügung:

¹¹ Über „edit defaults“ (siehe Umrandung) lassen sich Layout, Inhalte und Grafiken des Banners einstellen

Full Control:

Höchstes Privileg. Private und öffentliche Portalseiten können angepasst werden durch:

- Anpassen einer Seiteneigenschaft
- Hinzufügen und Löschen von Regionen
- Auswählen eines Seitenstils
- Bestimmen von Zugangsrechten für die Seite
- Hinzufügen und Löschen von Portlets und Tabsets
- Portlets/Tabsets ein- und ausblenden

Manage Portlets

Private und öffentliche Portalseiten können angepasst werden durch:

- Hinzufügen von Portlets zu vorhandenen Regionen
- Portlets/Tabsets ein- und ausblenden
- Hinzufügen und Löschen von Portlets/Tabs

Manage Style

Für private und öffentliche Portalseiten einen neuen Seitenstil auswählen.

Full Customization

Nur private Portalseiten können angepasst werden durch:

- Portlets ausblenden
- Hinzufügen von Portlets zu vorhandenen Regionen
- Löschen von Portlets, die selbst hinzugefügt wurden

Add-Only Customization

Nur private Portalseiten können angepasst werden durch:

- Hinzufügen von Portlets zu vorhandenen Regionen
- Löschen von Portlets, die selbst hinzugefügt wurden

View Only

Portalseite kann mit den Voreinstellungen betrachtet und kopiert werden.

4.2.2.5 Sprachanpassung

Diese Funktion erlaubt es dem Portalbenutzer die von ihm bevorzugte Landessprache einzustellen. Die Texte werden dynamisch in der ausgewählten Sprache mit dem Seitenaufbau generiert. Die Textattribute im Oracle Portal sind für den *NLS*¹²-Dienst angepasst und werden somit in der jeweiligen Landessprache angezeigt, sofern der Webbrowser NLS unterstützt und die Textattribute in dieser Sprache verfügbar sind.

Zu den Textattributen gehören u.a. Namen, Beschreibungen, Tabreiter-Beschriftung, Begrüßungstext im Banner, Konfigurationsmenü, etc.



Abbildung 40: Bsp. Spracheinstellungen beim „Verwalten“ Tab

4.2.3 Navigation

Das Oracle Portal bietet für die Navigation mehrere Navigationsportlets und den Portal Navigator an. Sie dienen dazu den Portaladministrator, Portalentwickler oder –anwender bei der Navigation innerhalb des Portals zwischen den Portlets, Portalseiten, Portalobjekten und externen URL's zu unterstützen.

Eine gute zusätzliche Navigationshilfe bieten weitere Navigationsportlets wie Favoritenverwaltung, Anzeige der zuletzt benutzten Objekte und Suchmöglichkeit nach Portalobjekten.

¹² National Language Support

4.3 Portaldienste

Oracle Portal beinhaltet eine Reihe von Tools und Diensten, die der Veröffentlichung und Verwaltung von Informationen (Portal Publishing Services) dienen. Jeder berechnigte Anwender kann damit Informationen veröffentlichen, sie verwalten und ihnen Zugriffsrechte zuordnen. Dazu bedient sich Oracle Portal den so genannten „Content Areas“ und „components“.

Content Areas sind nichts anderes als Webseiten, die im Baukastensystem aufgebaut, erweitert und verwaltet werden können.

Bei Components handelt es sich um Komponenten, die Applikationen darstellen und in die Portalseiten eingebunden werden können.

Zu Komponenten zählen u.a.: Berichte, Diagramme, Kalender, Hierarchien, Menüs oder auch JavaScripts.

4.3.1 Content Areas

Bei einer Content Area handelt es sich um eine Webseite, auf der ihr Inhalt über Wizards (Assistenten) veröffentlicht, organisiert, klassifiziert, gepflegt und angezeigt werden kann. Die Definition als Portlet ist ebenfalls möglich und damit auch die Darstellung als ein Bestandteil der Portalseite.

Definition einer Content Area erfolgt durch folgende Elemente:

- Ordner in den der Inhalt dargestellt wird
- der Inhalt selbst
- der Stil

Komponenten für die Navigation, Suche und Klassifizierung sind ebenfalls in der Content Area enthalten.

Die Erstellung einer Content Area wird im Kap. 5.3 beschreiben.

4.3.1.1 Ordner

Ein Ordner beinhaltet Unterordner oder Inhaltsobjekte (Items) und wird wie folgt kategorisiert:

Container:

Ein Ordner, der Elemente aus Oracle Portal enthält.

PL/SQL:

Ein Ordner, bei dem es zur PL/SQL-Code Ausführung kommt.

Search:

Ein Ordner, der einen Suchdienst zur Verfügung stellt.

URL:

Ein Ordner, in dem externe Webseiten eingebunden werden und somit in das EIP integriert. Auf ihren Inhalt kann über Content Areas oder Portalseiten zugegriffen werden.

Mit Hilfe von Assistenten lassen sich die Ordner individuell aufbauen, erweitern und verwalten. Sie besitzen Namen und Titel und können vordefinierten Kategorien und Perspektiven hinzugefügt werden. (Siehe Kap. 4.3.1.3). Innerhalb der Ordner kann das maximale Datenvolumen festgelegt werden und die verschiedenen Versionen der vorhandenen Elemente verwaltet werden. Ebenfalls kann man:

- Layout und Stil hinzufügen / verwalten
- Bilder und Banner integrieren
- Navigationsleiste hinzufügen / anpassen
- Zugriffsrechte verwalten
- Einzelne Ordner als Portlets veröffentlichen

Für die Zugriffsrechte und Sicherheit können 5 Privilegien verwendet werden.

Privileg	Erlaubt den Anwendern und Gruppen :
Own Folder	Höchstes Privileg. Ordner können angepasst werden durch: <ul style="list-style-type: none"> • Hinzufügen, Ändern und Löschen von Items • Ändern des Stils und der Navigationsleiste • Erstellen von Unterverzeichnissen • Ändern der Privilegien für den Ordner
Edit	<ul style="list-style-type: none"> • Ändern des Stils und der Navigationsleiste

Style	
Manage Items	<ul style="list-style-type: none"> • Hinzufügen, Ändern und Löschen von Elementen (Items)
Create With Approval	<ul style="list-style-type: none"> • Hinzufügen von Elementen (Items). Aktion muss vom Besitzer des Ordners bestätigt werden.
View Content	<ul style="list-style-type: none"> • Ansicht der Ordner und Inhalte

Sollte der Anwender mehreren Gruppen mit unterschiedlichen Privilegien zugeordnet sein, so erhält er die Summe der Privilegien dieser Gruppen



Abbildung 41: Darstellung des Content Area Editors¹³

4.3.1.2 Inhalt

Darstellung des Inhalts erfolgt beim Oracle Portal über Items. Darunter versteht man ein Basiselement eines Ordners, der durch den darzustellenden Inhalt und die zugehörigen Attribute definiert wird.

¹³ 1) vor und 2) nach klicken auf den „Edit folder“ Button

Ein Item wird mit Hilfe eines „Add-Item“ Assistenten erzeugt und veröffentlicht. Hierbei kann der Item-Typ, seine Attribute, die dazugehörige Kategorie und die Darstellungsart definiert werden.

Weitere Funktionen des Item Publishers sind: Versionskontrolle, Ein- / Auschecken, Gültigkeitszeitraum, automatische Indizierung, Verwaltung und Suchdienst für bereits veröffentlichte Items.

Bei Oracle Portal werden mehrere Item-Typen unterschieden:

- *File/Multiple Files*: Dateien können in die Datenbank vom Oracle Portal hochgeladen werden.
- *Folder Link*: Ein Verweis (Link) auf einen Ordner innerhalb der Content Area
- *Image*: Ein Bild-Item wird in die Datenbank hochgeladen und innerhalb des Ordners angezeigt.
- *PL/SQL*: Ein Item, bei dem es zur PL/SQL-Code Ausführung kommt
- *URL*: Ein Link, in dem externe Webseiten eingebunden und in dem Ordner dargestellt werden können.
- *Java Application*: Das Item verweist auf eine Java Anwendung (JSP, Servlet). Diese Anwendung führt beim Aufruf eine „jar“ Datei aus. Diese wird dabei in die Datenbank hochgeladen und dort ausgeführt.
- *Application Component*: Aus Oracle Portal Komponenten (Bericht, Grafik, Menü etc.) bestehendes Item, welches im Ordner oder als Verweis zu die jeweiligen Komponente dargestellt werden kann.
- *Zip*: Dieses Item besteht aus einer Zip-Datei (Vom Komprimierungstool erstellte Datei). Diese Datei wird in das Portal hochgeladen wobei seine Struktur durch Ordner und Items weiterhin dargestellt werden kann.
- *Text*: Text kann eingegeben werden, der beim Aufruf angezeigt wird.

Einem Item kann ein oder mehrere verbindliche und optionale Attribute besitzen. Diese Attribute werden beim Erstellen eines Items hinzugefügt oder später ergänzt. Folgenden Tabelle stellt alle Oracle Portal Standard-Item-Typen Attribute dar. Bei mit (*) gekennzeichneten Attributen handelte es sich um verbindliche Attribute.

Attribute	File	Folder Link	Image	URL	PL/SQL	Text	Java Component	Application Component	Zip
Display Region*	X	X	X	X	X	X	X	X	X
File Name*	X		X				X		X
Display Name*	X	X	X	X	X	X	X	X	X
Category*	X	X	X	X	X	X	X	X	X
Expiration Period*	X	X	X	X	X	X	X	X	X
Publish Date*	X	X	X	X	X	X	X	X	X
Description	X	X	X	X	X	X	X	X	X
Perspectives (multiple)	X	X	X	X	X	X	X	X	X
Author	X	X	X	X	X	X	X	X	X
Image	X	X	X	X	X	X	X	X	X
Basic Search Keywords	X	X	X	X	X	X	X	X	X
Enable Check Out	X	X	X	X	X	X	X	X	X
Hide Item	X	X	X	X	X	X	X	X	X
Display in-line with the folder				X	X	X		X	
Display with nav bar				X	X	X		X	
Display in full browser				X	X	X		X	

Tabelle 1: Oracle Portal Standard-Item-Typen

Erklärung einiger Attribute:

- Display Region: Region, in der das Item angezeigt werden soll.
- Categories: Attribut, das verwendet wird um Ordner und Items zu klassifizieren. Es können somit verwandte Ordner und Items unabhängig von der Ordnerhierarchie in einer Kategorie dargestellt werden.
- Perspective: Items und Ordner können über dieses Attribut zusätzlich gruppiert werden. Unabhängig von der Ordnerhierarchie können so verwandte Items oder Ordner über einen Perspektive-Ordner dargestellt werden. Perspektiven werden oft benutzt um die Frage “Wer hat Interesse an diesem Ordner oder Item?” zu beantworten.

- Basic Search Keywords: Über dieses Attribut können zu einem Item Stichwörter hinzugefügt werden, die bei der Suche innerhalb einer Content Area berücksichtigt werden sollen.
- Enable Check-out: Diese Attribut wird verwendet, um ein Item zu sperren. So kann ein Anwender ein Item bearbeiten, ohne dass andere Anwender zur gleichen Zeit auf dieses Item zugreifen.

Zusätzlich zu diesen Attributen erzeugt Oracle Portal weitere Attribute, wie Item Eigenschaften (Zusammenfassung sämtlicher Attribute eines Items).

Dazu gehören:

1. Erstellungsdatum
2. Ersteller-Name
3. Dokumentengröße (beim File-Typ)
4. “neues-,, und “geändertes-Item” Kennung.

Mit Oracle Portal wird dem Anwender die Möglichkeit der individuellen Item-Typendefinition geboten. Es ermöglicht die Erzeugung von Item-Typen mit speziellen Attributen oder auch die Anpassung von Standard-Item-Typen auf individuelle Anforderungen. Der Content Area Administrator kann dann festlegen welche Art von Item-Typen für seine Content Area verwendet werden soll.

4.3.1.3 Inhaltsklassifikation

Der Inhalt einer Content Area lässt sich über Kategorien und Perspektiven klassifizieren und gruppieren. In ihnen werden verwandte Items dargestellt. Der Unterschied zwischen Kategorien und Perspektiven ist folgender:

Ein Item muss einer einzigen Kategorie zugehören, kann aber optional zu einer Perspektive hinzugefügt werden.

Über den Portal Navigator lassen sich diese Perspektiven und Kategorien vom berechtigten Anwender erstellen und über verschiedene Arten zugänglich machen.

- Link über einen Banner
- Link über Item Attribut Kennung
- Link über Navigationsleiste
- Veröffentlichtes Portlet

4.3.1.4 Navigationsleiste

Die Navigationsleiste ist durch die Elemente und den Stil, der von den Elementen zur Darstellung benutzt wird, definiert. Sie dient als zentrale Navigationshilfe innerhalb des Portals.

Im Oracle Portal sind bereits vordefinierte Elemente vorhanden:

- Content Area Name
- Portallink für ein- und ausloggen
- Liste mit Kategorien und Perspektiven
- Links zu definierten Ordnern innerhalb der Content Area
- Suchdienst
- Kontaktlink
- Link zu Startseite

Diese Elemente können vom berechtigtem Benutzer Optional aktiviert / deaktiviert werden.

Die Navigationsleiste kann von bereits bestehenden Content Area Ordnern geerbt oder mit bereits definierten Elementen und deren Zugriffsrechten verbunden werden.

Zusätzliche Elemente wie URL-Links, Texte, Bilder und Ordner können zu den bereits vordefinierten Elementen einer Navigationsleiste hinzugefügt werden. Ihr Stil kann mit dem Stilmanager geändert und angepasst werden.

4.3.1.5 Stil

Mit dem Stil der Content Area wird das Aussehen der Ordner, Banner und der Navigationsleiste bestimmt, sowie die Ausrichtung und Darstellung der Elemente. Mit Hilfe des Stil-Managers können Stil-Elemente zu komplexen Layouts und Stilen vereint werden.

Bei der Navigationsleiste kann folgendes angepasst werden:

- Breite
- Texteingenschaften
- Textfarben (Link, besuchter Link etc.)
- Hintergrundfarben
- Hintergrundbilder

Bei den Bannern können verschiedene Bannerarten eingerichtet werden: Hauptbanner, Nebenbanner, Regionenbanner, Gruppenbanner. Diese lassen sich zusätzlich ausrichten und deren Texteingenschaften, Hintergrundgrafiken oder -farben definieren und anpassen.

Die Ordner der Content Area können mit einer umfangreichen Stilverwaltung angepasst werden. Hierbei stehen vorkonfigurierte Layouts, die aus fünf Regionen (Frames) bestehen, zur Verfügung. Diese Regionen sind geometrisch anpassbar und definieren die Art wie die Elemente (Ordner, Items) innerhalb einer Region dargestellt werden. Dazu gehört:

- Die Reihenfolge
- Gruppierung
- Zeilen, Spalten-Ausrichtung

Die Verwendung von Layouts aus XSL-Dateien wird von Oracle Portal ebenfalls unterstützt



Abbildung 42: Darstellung des Contentarea-Styl-Editors]

Was hier besonders hervorzuheben ist die Eigenschaft, dass ein definierter Stil für mehrere Content Areas verwendet werden kann. Diese Funktionalität soll eine einheitliche „look and feel“ Einhaltung innerhalb des EIP's gewährleisten.

4.3.1.6 Content Area Suchdienst

Im Oracle Portal ist ein vorkonfigurierter Suchdienst enthalten, der nach bestimmten Elementen (Items, Ordner, Attribute, Perspektiven) innerhalb der Content Area suchen kann. Dem Benutzer stehen drei Sucharten zur Verfügung.

- a) Einfache Suche (Basic Search):
Suche innerhalb der Content Area

- b) Erweiterte Suche:
Erweitert die Suchfunktion durch Eingabe von zusätzlichen Suchkriterien. So kann beispielsweise nur innerhalb bestimmter Content Areas nach genau definierten Elementen gesucht werden.

- c) Intermedia Suche:
Bei dieser Suchmöglichkeit wird der Oracle 8i Inter Media Suchdienst, der eine Volltextsuche ermöglicht, hinzugezogen

Die Darstellung der Suchergebnisse erfolgt zusammen mit dem Inhalt-Typ und dem dazugehörigen Verweis. Speichern einer Suche wird ebenfalls ermöglicht.

4.3.2 Komponenten (Components)

Mit dem Oracle Portal wird dem Anwender für die Erstellung, Verwaltung und Einbindung von dynamischen Komponenten eine einfache Entwicklungsumgebung geboten. In dieser können die Komponenten ohne Programmierkenntnisse schrittweise erstellt und in einer Applikationsumgebung verwaltet werden.

Eine Applikation stellt einen Container für Komponenten und Datenobjekte dar.

The screenshot shows the Oracle Portal Navigator interface. At the top, it displays the Oracle logo and the title 'Oracle Portal Navigator'. The date is September 5, 2002. There are navigation tabs for Pages, Content Areas, Applications, and Database Objects. Below the tabs, there is a search bar and a 'Go' button. The main content area shows a list of application components within the EXAMPLE_APP container. The list has columns for Type, Name, Actions, Owner, and Modified. The components include various forms, reports, and charts.

Type	Name	Actions	Owner	Modified
Calendar from SQL Query	EXAMPLE_CAL	Manage, Run, Edit, Delete, Grant Access	EXAMPLE_APP	23-May-2002
List of Values	EXAMPLE_DEPTNO	Manage, Run, Edit, Delete, Grant Access	EXAMPLE_APP	23-May-2002
Dynamic Pages	EXAMPLE_DYNAMIC_PAGE	Manage, Run, Edit, Delete, Grant Access	EXAMPLE_APP	23-May-2002
Forms on Tables/Views	EXAMPLE_FORM	Manage, Run, Edit, Delete, Grant Access	EXAMPLE_APP	23-May-2002
Frame Driver	EXAMPLE_FRAME	Manage, Run, Edit, Delete, Grant Access	EXAMPLE_APP	23-May-2002
Hierarchies	EXAMPLE_HIERARCHY	Manage, Run, Edit, Delete, Grant Access	EXAMPLE_APP	23-May-2002
List of Values	EXAMPLE_JOB	Manage, Run, Edit, Delete, Grant Access	EXAMPLE_APP	23-May-2002
Master Detail Forms	EXAMPLE_MD_FORM	Manage, Run, Edit, Delete, Grant Access	EXAMPLE_APP	23-May-2002
Menus	EXAMPLE_MENU	Manage, Run, Edit, Delete, Grant Access	EXAMPLE_APP	23-May-2002
Query by Example (QBE) Reports	EXAMPLE_QBE_REPORT	Manage, Run, Edit, Delete, Grant Access	EXAMPLE_APP	23-May-2002
Forms on Stored Procedures	EXAMPLE_SP_FORM	Manage, Run, Edit, Delete, Grant Access	EXAMPLE_APP	23-May-2002
Chart from SQL Query	EXAMPLE_SQL_CHART	Manage, Run, Edit, Delete, Grant Access	EXAMPLE_APP	23-May-2002
Report from SQL Query	EXAMPLE_SQL_REPORT	Manage, Run, Edit, Delete, Grant Access	EXAMPLE_APP	23-May-2002
Chart from Query Wizard	EXAMPLE_WIZ_CHART	Manage, Run, Edit, Delete, Grant Access	EXAMPLE_APP	23-May-2002
Report from Query Wizard	EXAMPLE_WIZ_RPT	Manage, Run, Edit, Delete, Grant Access	EXAMPLE_APP	23-May-2002
Link from Wizard	LINK_TO_UPD_EXP	Manage, Run, Edit, Delete, Grant Access	EXAMPLE_APP	23-May-2002

Abbildung 43: Container für Applikationen¹⁴

Entwicklern ist es möglich diese Objekte entweder innerhalb einer Applikation zu erstellen und zu verwalten oder eine neue Applikation innerhalb eines existierenden Schemas zu erzeugen.

Die Komponenten zählen zu den wichtigsten Elementen des Oracle Portal Frameworks, da sie als Portlets innerhalb des EIP's als einzelne Web-Anwendungen entwickelt, verwaltet und integriert werden können.

4.3.2.1 Formulare

Mit den webbasierten Formularen kann der Anwender in Interaktion mit der Datenbank treten. So kann er z.B.: Informationen, Bilder, Audiodateien etc. abfragen, hinzufügen, ändern und löschen.

Das Oracle Portal bietet zwei Formular-Typen:

¹⁴ Hier dargestellt : Inhalt des « EXAMPLE_APP » Containers

a) *Einfache Formulare (Single Form):*

Sie werden direkt mit Tabellen oder Views verknüpft oder durch PL/SQL-Anweisungen, durch die erweiterten Funktionalitäten möglich sind, erstellt.

b) *Master-Detail Formulare:*

Dieser Typ besteht aus einer Master-Spalte und einem Detail-Formular, wobei der Inhalt des Detail-Formulars von der Master-Spalte gesteuert wird.

Zur Erstellung von Formularen steht der Formular-Assistent zur Verfügung. Er baut auf einem Layout-Template auf, wodurch die einzelnen Spalten als Elemente verwaltet und angepasst werden können. Die Einbindung von Javascript und PL/SQL-Anweisungen für bestimmte Aktionen ist ebenfalls möglich.

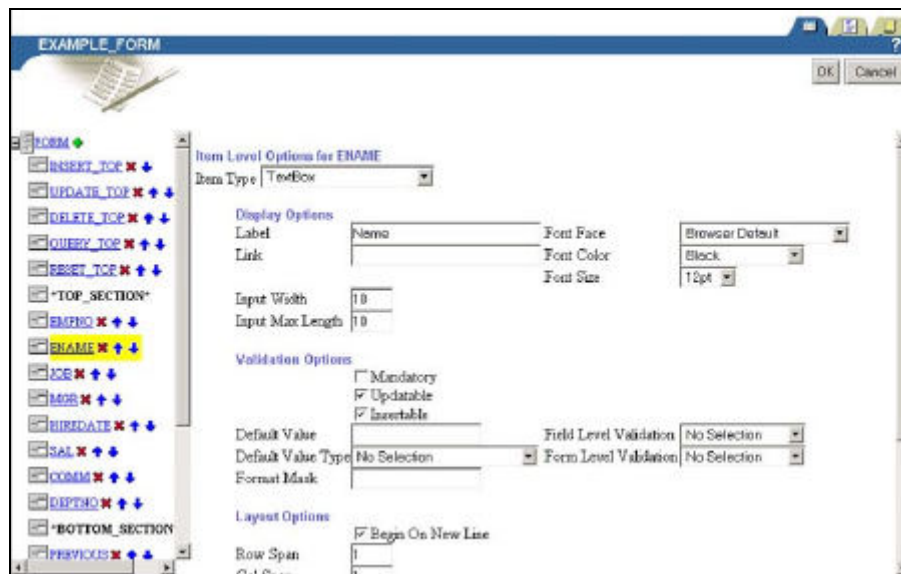


Abbildung 44: Erstellung eines Formulars¹⁵

Ein erstelltes Formular kann den Anwendern als Portlet bereitgestellt werden.

¹⁵ Beispiel: Formular dient der Einfügung von Dateien in die „EPM“ Tabelle

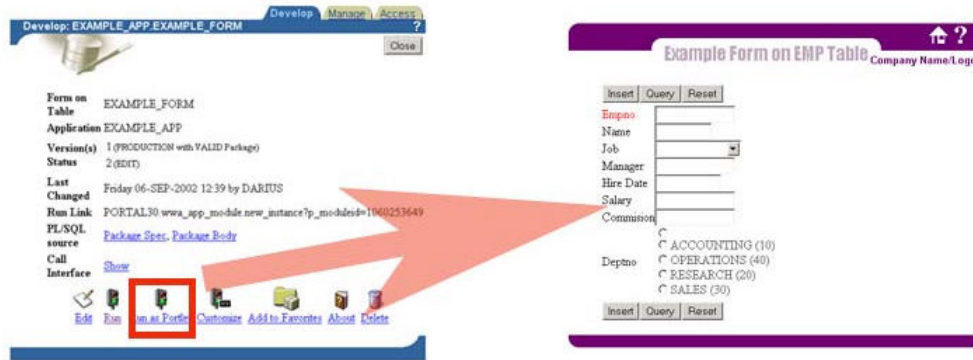


Abbildung 45: Darstellung eines Formulars als Portlet im Browser

4.3.2.2 Berichte

Ähnlich wie bei den Formularen bauen Berichte auf einer oder mehreren Tabellen oder Views auf. Bei ihnen handelt es sich um Web-Komponenten, die eine Darstellung von Daten in Form eines in *HTML*-Code erstellten Berichts ermöglichen.

Dem Anwender ist es überlassen, einen Bericht durch spezifizierte Vorgaben vom Oracle Portal automatisch erstellen oder diesen durch eigene *SQL*-Anweisungen generieren zu lassen.

Hier steht ebenfalls ein Assistent zur Verfügung. Mit dem „Bericht-Entwickler-Assistenten“ lassen sich Felder zusätzlich bearbeiten. Durch die zusätzliche *QBE*¹⁶-Komponente lassen sich Daten einer Tabelle oder Berichte aktualisieren, löschen oder hinzufügen. Sie hat somit eine ähnliche Funktion wie ein Formular.

Die Funktionen von Berichten lassen sich folgend gliedern:

- Titelpersonalisierung
- Zusätzliche Abfrageoptionen
- Festlegung der Sortierreihenfolge
- Festlegung des Layout-Stils
- Festlegung der max. Anzahl der Zeilen pro Seite
- Festlegung der Schriftgröße

Erstellung von Berichten erfolgt durch klicken auf das Icon:



Report from Query Wizard

¹⁶ Query by Example

4.3.2.3 Diagramme

Bei der Diagrammfunktion handelt es sich ebenfalls um eine Web-Komponente, die der dynamischen Darstellung von Daten in Form von Diagrammen dient. Ein Diagramm, welches als eine grafische Form eines Berichts anzusehen ist, kann vergleichbar mit einem Bericht oder Formular, mit einem Assistenten erzeugt werden oder unter Verwendung eigener *SQL*-Abfrage. Zur weiteren Bearbeitung, wie zum Beispiel: Einbindung von Layout Templates, Farb- und Schriftartanpassung, Sortierreihenfolge, etc. muss mit dem Diagrammentwickler gearbeitet werden.

Erstellung von Diagrammen erfolgt durch klicken auf das Icon:



Die Funktionen von Diagrammen lassen sich folgendermaßen gliedern:

- Titelpersonalisierung
- Zusätzliche Abfrageoptionen
- Festlegung der Sortierreihenfolge
- Festlegung des Diagrammtyps (Horizontal / Vertikal)
- Festlegung der Achseneinteilung
- Zusammenfassung (Summe, Durchschnitt, Maximum, etc.)



Abbildung 46: Mit Hilfe des Wizards erstelltes Diagramm

4.3.2.4 Dynamische Seiten

Diese Komponente dient der Einbindung von *PL/SQL*-Code in statische *HTML*-Seiten. So kann eine dynamische Seite, durch die Möglichkeiten des *PL/SQL*, über komplexe Funktionalitäten verfügen.

Um dynamische Seiten zu erstellen sind Programmierkenntnisse erforderlich. Der *PL/SQL*-Code wird innerhalb des *HTML*-Codes in einem Oracle Tag `<ORACLE>` `</ORACLE>` eingebettet.

Beispiel:

```
<HTML>
  <HEAD>
    <TITLE> Beispiel </TITLE>
  </HEAD>
  <BODY>
    <ORACLE> select * from scott.emp </ORACLE>
  </BODY>
</HTML>
```

Diese Funktion ist bei der Entwicklung von Portlets verwendbar und ermöglicht eine Personalisierung des Titels und der Abfrageoptionen.

Erstellung von dynamischen Seiten erfolgt durch klicken auf das Icon:



Dynamic Pages

4.3.2.5 Kalender

Die Kalender Komponente im Oracle Portal dient der Darstellung von Daten in einem Kalender. Es bietet die Möglichkeit eine Art „elektronisches Tagebuch“ aufzubauen, in dem Daten aus der Datenbank dargestellt werden. Mit der Kalender-Komponente werden die darzustellenden Daten ausgewählt, die dann mit einer *SQL*-Abfrage ausgegeben und angezeigt werden. Auch hier existiert ein Tool, mit dem die Darstellung des Kalenders angepasst werden kann. Innerhalb des Kalenders können Links zu verschiedenen Komponenten eingefügt werden. Durch Integration von *PL/SQL*-Code kann der Kalender zu einem persönlichen Terminplaner erweitert werden. Die Kalender-Personalisierung kann

durch Einstellung des Titels, der Darstellungsart (nur Wochentage anzeigen, maximale Anzahl von Monaten pro Seite, Schriftgröße) und der Abfrageoptionen erfolgen.

Erstellung eines *PL/SQL*-Kalenders erfolgt durch klicken auf das Icon:



Calendar from SQL Query

Example Calendar Results							Company Name/Logo
June 2002							
Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Sat	
						01	
02	03	04	05	06	07	08	
09 CARLUS	10	11	12	13	14	15	
16	17	18	19	20	21	22	
23	24	25	26	27	28	29	
30							

Abbildung 47: Kalender mit Zugriff auf Daten aus der DB¹⁷

4.3.2.6 Hierarchie

Die Hierarchie-Komponente dient der Darstellung der Daten in einem hierarchischen Format. Beispielsweise zur Darstellung der Mitarbeiterhierarchie in einem Unternehmen.

Eine Hierarchie baut auf Tabellen und Views auf und kann bis zu drei Ebenen beinhalten. Diese Ebenen können stufenweise durchnavigiert werden.

Mit dem Hierarchie-Entwickler kann das Layout und die Darstellungsart angepasst werden. Integration von *PL/SQL* Anweisungen ist ebenfalls möglich und dient der Erweiterung der Funktionalität. Die Verwendung der Hierarchie-Komponente als ein Portlet ermöglicht dem Anwender eine Personalisierung folgender Merkmale:

- Titel
- Abfrageoption
- Hierarchie-Stil (*HTML*-Tabelle / Top-Down)
- Maximale Anzahl anzuzeigender Elemente in den Ebenen

¹⁷ „scott/tiger“ Schema

4.3.2.7 Frame Driver

Diese Komponente wird für den Aufbau von zwei *HTML*-Frames (Rahmen) verwendet. Hierbei steuern die Aktionen des Master-Frames die Inhalte des Detail-Frames.

Eine Frame-Driver Komponente wird durch eine *SQL*-Abfrage mit CGI-Skript Inhalten, erzeugt. Die Darstellungsparameter können nachträglich vom Benutzer angepasst werden.

Erstellung von Frame-Driver erfolgt durch klicken auf das Icon:



Frame Driver



Abbildung 48: Frame-Driver gesteuerten Abfragemaske

4.3.2.8 Menü

Die Menü-Komponente wird zur Darstellung von Links innerhalb des Portals, URL's und Untermenüs verwendet.

Ein Menü kann aus mehreren Untermenüs und Menüelementen aufgebaut sein. Die Verwaltung der Darstellung der Menüelemente erfolgt über das Komponenten-Entwicklungs-Tool. In einem Menü ist die Darstellung von bis zu fünf Ebenen einer Menühierarchie möglich und es wird in die Portalumgebung als ein Portlet eingebunden. Hierbei kann der Titel, die maximal anzuzeigenden Ebenen und die Anzeige der Erklärungen (bei Menüelementen) personalisiert werden.

4.3.2.9 Gemeinsam benutzte Komponenten

Mit dem Oracle Portal stehen dem Benutzer vier weitere Komponenten zur Veröffentlichung von Informationen zur Verfügung. Dabei handelt es sich um:

- List of Values (Wertelisten)
- Java-Script
- User Interface Templates (Layout Vorlagen)
- Links

List of Values (im Folgenden *LoV*) dient dem Benutzer als Eingabehilfe. Es handelt sich dabei um vorgefertigte Listen mit Werten, aus denen der Benutzer gewünscht Felder auswählen kann.

Eine *LoV* beinhaltet entweder statische oder dynamische Werte. Alle Oracle Portal Komponenten können *LoV*'s verwenden und ihre Darstellung kann als Check-Box, Combo-Box, Pop-up Liste, Radiobutton oder Mehrfachauswahl Liste erfolgen.

Java-Script sorgt für zusätzliche Funktionalität innerhalb der Content Area (in Tags eingebettet). Hauptsächlich findet es Verwendung bei der Gültigkeitsüberprüfung von Eingabefeldern und -formen.

User Interface Templates gewährleisten ein einheitliches Aussehen der Portals („look and feel“). Die verwendeten Farben, Schriftarten und die Gestaltung der Portalseite werden über sie definiert.

Es wird hierbei zwischen Strukturierten und unstrukturierten Vorlagen unterschieden.

Strukturierte Vorlagen: Erfordern keine *HTML*-Kenntnisse, da sie aus bestimmten Attributen wie Links, Grafiken, Farben, Schriftarten erstellt werden.

Unstrukturierte Vorlagen: Werden aus individuellem *HTML*-Code generiert

Links dienen der Navigation innerhalb verschiedener Komponenten. (Oracle Portal Komponenten oder auch externe Webseiten). Bei Webseiten bei deren Aufruf Eingabeparameter erforderlich sind, können diese durch einen Link gleich mit übergeben werden.

4.4 Architektur und Sicherheit

4.4.1 Architektur

Das Oracle Portal besteht aus folgenden 4 Hauptkomponenten:

- I. Oracle Portal Service
- II. Oracle Login Server
- III. Oracle HTTP-Server
- IV. Oracle 8i Datenbank

Diese sind in einer 3-Schichten Architektur eingebunden.

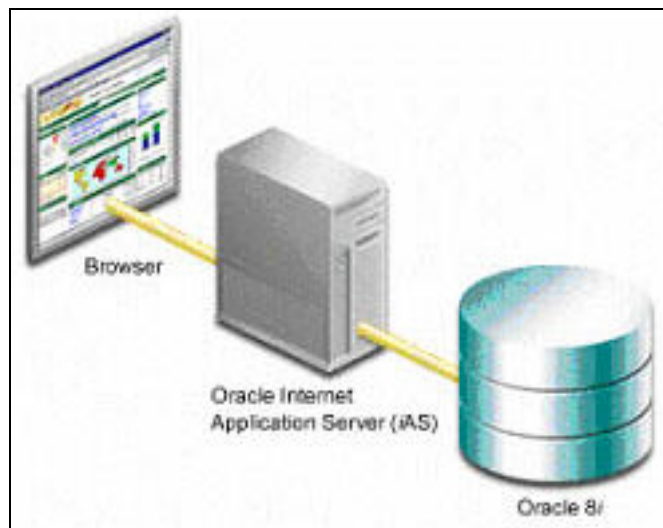


Abbildung 49: Oracle Portal three-tier Architektur(vereinfacht)

Im Folgenden werden die 4 Komponenten erläutert.

I. Oracle Portal Service

Erzeugt die Benutzeroberfläche (Assistenten, Dialoge), verwaltet alle Anwendertätigkeiten und seinen Input, ist für die Anwendungslogik verantwortlich. Diese Komponente ist im Oracle Internet Application Server (iAS) integriert und befindet sich somit in der Mittelschicht (2nd-tier). Die Implementierung der Anwendungslogik erfolgt in Form von *PL/SQL*-Prozeduren, die auf Tabellen zugreifen, welche während der Installation von Oracle Portal in der darunter liegenden Datenbank (third-tier) angelegt wurden.

II. Oracle Login Server

Zuständig für die Authentifizierung und Verwaltung der Anwender sowie des Single Sign-On Dienstes.

III. Oracle HTTP -Server

Stellt die Kommunikation zwischen der Mittelschicht und dem Browser her. Basierend auf einem Apache Server, übersetzt dieser Dienst *HTTP*-Anfragen des Browsers in Aufrufe an die (Portal-) Datenbank. Es werden Java Servlets und *HTTPS* unterstützt. Dieser Dienst wird bei der iAS Installation mit installiert.

IV. Oracle 8i Datenbank

Oracle Datenbank in der Version 8. Sämtliche Informationen über den Aufbau und die Inhalte des Portals werden hier gespeichert. Sie ist für die Ausführung der gespeicherten Prozeduren (stored procedures – Prozeduren die innerhalb der Datenbank ausgeführt werden) zuständig.

Folgende Abbildung soll den Datenaustausch zwischen den Oracle Portal Komponenten vereinfacht darstellen.

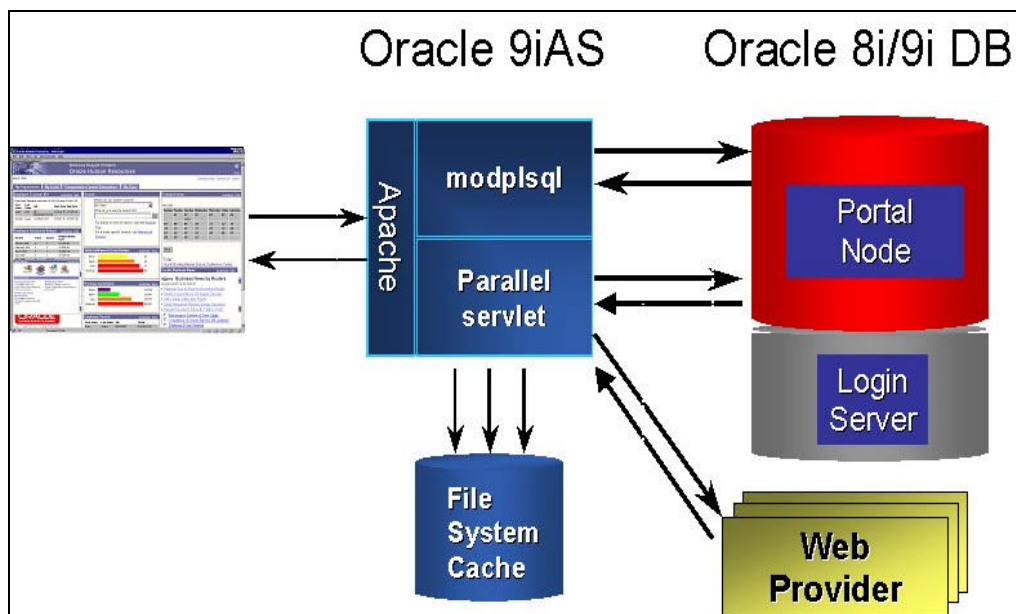


Abbildung 50: Oracle Portal 3.0 Architektur

Nach der Eingabe der URL in den Browser, wird eine Anfrage an den Server gesendet. Die Anfrage wird im *HTTP*-Server in einen Aufruf an die Portal Prozedur umgewandelt. Im Aufruf sind Informationen über die Sitzung des Benutzers und die von ihm angeforderte Seite. Als Antwort bekommt der *HTTP*-Server die Metadaten dieser Seite, ihre Definition und individuelle Einstellungen des Benutzers. Wenn die Seite Identitätsinformationen (Passwort, Username) des Benutzers benötigt, werden diese durch einen „Cookie“ an einen

oder mehrere **Provider** gesendet. Diese Informationen sind zur Erstellung weiterer Sitzungen (Sessions) notwendig.

Die Anwendungslogik des Servles (Parallel Servlet) empfängt die Seitendefinition. Sie sendet dann ein oder mehrere Aufrufe an die Provider. Der Aufruf kann wie folgt erfolgen:

- a) In Form eines Prozedur-Aufrufs im lokalen oder entfernten (remote) Oracle Portal Knoten (node).
- b) In Form einer *HTTP*-Anfrage mit *HTTP*-Aufrufen von Prozeduren/Methoden eines Providers

In dieser *HTTP*-Anfrage sind auch die an die Provider gesendeten „cookies“ enthalten. Es wird auch geprüft, ob aufgerufene Portlets aus dem Filesystem-Cache geladen werden können wenn sie sich bereits dort befinden und noch gültig sind.

Jetzt wird die Anfrage von den Providern ausgeführt und das Ergebnis an das Parallel Servlet zurückgesendet. Dieses erzeugt mit dem Ergebnis die Webseite, indem alle Antworten der Provider zusammengefügt werden. Der Seiteninhalt wird mit den benötigten Sitzungsinformationen (cookie headers) an den Webbrowser zurückgesendet.

Provider = Eine Einheit im Oracle Portal, die eine Datenquelle oder Applikation mit einem oder mehreren Portlets darstellt. Sie ist die Kommunikationsschnittstelle zwischen den Portlets und dem Portal Framework.

Es gibt im Oracle Portal zwei Arten von Providern:

a) *Datenbankprovider:*

Dieser Provider ist in der Datenbank als Stored Procedure (gespeicherte Prozedur) implementiert. Er kann von Oracle Portal nur über lokale Prozeduraufrufe angesprochen werden.

b) *Webprovider:*

Dieser Provider ist auf einem Webserver implementiert. Hierbei handelt es sich um eine Webapplikation mit der das Oracle Portal über das HTTP-Protokoll des Webservers kommuniziert.

Erklärung von Diensten, die in der Architektur integriert sind:

- *Parallel Servlet:*

Diese Anwendung dient der Erzeugung von Webseiten (Page Engine). Sie ist in der Mittelschicht als Servlet implementiert, da es Servlets möglich ist mehrere Anfragen gleichzeitig auszuführen (multithread).

Dieser Dienst dient auch der Unterstützung der remote- und multinode portlet provider.

- *Portlet Caching:*

Hierbei wird zwischen validation based- und expiry based caching unterschieden.

validation based caching: Die Gültigkeit eines Portlets wird geprüft, sollte diese bestehenden wird es aus dem Portal Cache geladen.

expiry based caching: Bei dieser Form wird das im Cache geladene Portlet so lange automatisch verwendet, bis eine bestimmte Zeitdauer abläuft und es neu geladen werden muss.

- *Cascading Style Sheets (CSS):*

Speicherung des Stils und des Layouts von Seiten und Portlets erfolgt in CSS-Klassen.

4.4.2 Sicherheit

Der Anwender wird am Oracle Portal durch den Login Server authentifiziert und mit Hilfe des Single Sign-On (SSO) bei Anwendungen innerhalb des Oracle Portals, die eine zusätzliche Authentifizierung benötigen, automatisch angemeldet.

Dem Portal Administrator stehen zwei Arten der Benutzer-Authentifizierung zur Verfügung.

Lightweight User Authentication

Hierbei wird innerhalb des Oracle Portal ein Benutzerschema für den jeweiligen Benutzer angelegt. Das bedeutet aber, dass der Portal Benutzer lediglich innerhalb des Oracle Portal ein Datenbankbenutzer ist. Die Benutzer und ihre Informationen werden innerhalb von Oracle Portal verwaltet.

External Repository Authentication

Diese Authentifizierungsart basiert auf bestehenden *LDAP-Verzeichnissen* die auf einen externen Server liegen. Der Login Server nimmt Verbindung mit dem *LDAP-Verzeichnis* auf und überprüft hier die gespeicherten Daten des Portalbenutzers.

Der Single Sign-On-Dienst (SSO) ermöglicht es dem Portalbenutzer sich an mehreren Anwendungen (interne und externen) mit einer einmaligen Authentifizierung anzumelden und auf diese zuzugreifen. Dieser Dienst unterscheidet zwischen Partner- und Externen-Anwendungen.

Partneranwendung

Die Partneranwendung benutzt bei der Authentifizierung die *API* des *SSO* und lässt die Identität des Portalbenutzers vom Oracle Login Server prüfen.

Sobald dieser authentifiziert wurde, kann er auf die Partneranwendungen zugreifen.

Externen Anwendungen

Diese Art von Authentifizierung ist nötig, wenn die externen Anwendungen nicht dem Model der o.g. Partneranwendung entsprechen. Die Authentifizierungsdaten des Portalbenutzers werden nach der ersten Registrierung an der externen Anwendung innerhalb von Oracle Portal gespeichert. Bei einem erneuten Zugriff auf diese externe Anwendung sendet der Login Server die benötigten Daten an die anfordernde Anwendung.

4.5 Administration

Dem Portaladministrator stellt das Oracle Portal alle nötigen Dienste zur Verfügung. Diese können uneingeschränkt innerhalb des Portals über den Webbrowser verwaltet werden.

Dazu gehört Verwaltung der:

- Anwender und Gruppen
- Sicherheitseinstellungen
- Datenbank
- Einstellungen für Portalüberwachung



Abbildung 51: Administrations-Seiten im Oracle Portal¹⁸

¹⁸ Zusammenstellung verfügbarer Administrations-Portlets, die über Registerreiter wählbar sind

4.5.1 Anwenderverwaltung

Im Oracle Portal steht für den Portaladministrator ein „Benutzer-Verwaltungsportlet“ zur Verfügung. Damit kann er neue Benutzer anlegen und bestimmen, welchen Gruppen sie zugeordnet werden sollen. So ist es dem Administrator möglich Einfluss auf die Seiten und Inhalte, die für den jeweiligen Benutzer zugänglich sein sollen, zu nehmen. Es wird zwischen public- und authorized-user unterschieden.

Public User (Öffentlicher Benutzer)

Kann alle, als öffentlich deklarierten Objekte im Portal einsehen, jedoch nicht bearbeiten.

Authorized User (autorisierter Benutzer)

Kann Seiten innerhalb des Portals erzeugen, bearbeiten, Stil und Layout ändern und abhängig von seiner Berechtigung sogar löschen.

Ein Oracle Portal bildet sich aus verschiedenen Elementen, wie:

- Portalseiten
- Layouts und Stile
- Portlets
- Anbieter
- Content Areas
- Applikationen
- Gemeinsam benutzte Komponenten
- Anwender
- Gruppen
- Schemen

Auf jedes einzelne Element kann der Portaladministrator den Zugriff steuern.

4.5.2 Gruppenverwaltung

Unter einer Gruppe versteht man eine Ansammlung von Benutzern mit gleichen Interessen und Rechten innerhalb des Portals. Sie kann von jedem autorisierten Benutzer, der somit der Besitzer der Gruppe wird, erstellt werden. Ihm ist es möglich anderen Benutzern „Besitzerrechte“ an seiner Gruppe zu geben.

Des Weiteren können Gruppenbesitzer und Portaladministratoren Gruppen löschen und sich bei anderen Gruppen anmelden oder Mitglieder einer Gruppe unsichtbar darstellen. Eine weitere Eigenschaft von Gruppen ist, dass sie auf bestimmte Content Areas beschränkt werden können.

Während der Installation von Oracle Portal werden automatisch Standardgruppen eingerichtet, die sich mit Zugriffsprivilegien auf das Portal und die Datenbank unterscheiden.

PUBLIC:

Gruppe für öffentliche Benutzer.

AUTHENTICATED_USER:

Gruppe für autorisierte Benutzer

PORTLET_PUBLISHER:

Autorisierte Benutzer, die zusätzlich Portlets veröffentlichen dürfen

PORTAL_DEVELOPERS:

Autorisierte Benutzer, mit Rechten zur Komponenten-, Applikations- und Datenbankobjekterzeugung.

PORTAL_ADMINISTRATORS:

Autorisierte Benutzer mit Rechten auf alle Objekte ausschließlich jener innerhalb der Portal Datenbank.

DBA:

Autorisierte Benutzer mit vollständigen Zugriffsrechten auf alle Objekte, innerhalb des Oracle Portals und der Portal Datenbank.

4.5.3 Datenbankverwaltung

Oracle Portal verwendet eine eigene Datenbank, deren Verwaltung durch bereitgestellten Portlets ermöglicht wird. Dazu gehören folgende Portlets:

Schemata:

Dieses Portlet dient dem Portaladministrator zur Erzeugung und Bearbeitung von Schemata sowie der Zuteilung von Rollen und Privilegien

Rollen:

Dient der Erzeugung und Bearbeitung von Rollen

Privilegienerteilung:

Dient der Privilegienuzuweisung für Datenobjekte an die Benutzer.

Datenbankinformationen:

Mit diesem Portlet können diverse Einstellungen des Portals eingesehen werden. Dazu gehören: Datenbankeinstellung, Name, Portal Version, Log Datei, Initialisierungsdatei, NLS, etc.

Datenbankspeicherung:

Ermöglicht das Einsehen von Informationen über Tabellenspeicherplatz, freier/verwendeter Speicher, Quotas für Schemata, Log Datei, etc.

Datenbankspeicher, -auslastung, -transaktion, -sperrung:

Dieses Portlet ermöglicht den Zugriff auf Informationen die Datenbank betreffend. Dazu gehören: aktuelle Aufgaben, Sessions (Sitzungen), verbundene Benutzer. Dem Administrator ist es möglich mit diesem Portlet Sitzungen zu beenden.

4.5.4 Überwachung

Das Oracle Portal bringt fertige Portlets mit sich, mit denen die Überwachung von Aktivitäten innerhalb des Portals ermöglicht wird. Es soll dem Portaladministrator dazu dienen, Fehlerquellen ausfindig zu machen und die Performance zu verbessern aber auch um die Anwender zu überwachen.

Für jedes Objekt kann eine Auswertung in Form eines Berichtes oder Diagramms erzeugt werden. In ihm können Aktivitäten abhängig von Anwendern, Zeiträumen, IP-Adressen, Sprachen und Browsertypen angezeigt werden. Aktivitäten von Seiten, Portlets, Dokumenten, Items, Suchdiensten, Perspektiven und Anwendungen oder sogar ganzer Content Areas können erfasst und dargestellt werden.

4.5.5 Zusätzliche Portaladministrations-Dienste

Oracle Portal verfügt über drei weitere Verwaltungsportlets, die sich in keine der oben genannten Kategorien einordnen lassen.

Service Portlet (Dienstportlet) dient der Einstellung:

- der Standard-Startseite des Portals
- des Proxy Servers
- des Überwachungsintervalls
- des Login Schemas für den Server
- des *DAD* (Database Access Descriptor)
- des Portal-Listeners
- der Log Dateien
- der Suchdienste
- des Login Servers

Provider Portlet dient der Registrierung und Verwaltung der Portlet Provider, nachdem sie einem Schema innerhalb von Oracle Portal zugefügt wurden.

Node Portlet (Knotenportlet) dient der Erstellung und Bearbeitung von lokalen und externen Oracle Portal Knoten.

- Netzwerkeinstellungen
- Informationen über entfernte Datenbanken
- Listenereinstellungen

4.6 Java Portlet Development Kit (JPDK)

Wie in Kapitel 4.2.1 bereits vorgestellt, bietet das von Oracle bereitgestellt Java Portlet Development Kit vielseitige Funktionen und Beispiele um selbst entwickelte Portlets in das EIP einzubinden. Da im Verlauf dieser Arbeit die praktische Vorstellung eines Java-Portlets nicht realisierbar ist, wird im Folgenden die Funktionalität der JPDK theoretisch erläutert.

4.6.1 JPDK

Das JPDK ist ein Teil des PDK (Portlet Development Kit). Es stellt vordefinierte Java-Klassen für die Erstellung von Java Portlets zur Verfügung. Es besteht aus 4 Bereichen:

- Provider Adapter
- Provider Interface
- Provider Runtime
- Provider Utilities

In der Datei „provider.xml“ werden die Informationen über die einzelnen Provider und Portlets verwaltet. Sie spielt bei der Einbindung von Portlets ins Portal eine wichtige Rolle.

4.6.2 provider.xml

Bei dieser Datei handelt es sich, wie die Endung vermuten lässt, um eine XML-Datei. Diese statische Datei speichert Informationen über den Provider und seine Portlets. Einem Provider ist genau eine „provider.xml“ Datei zugeordnet. Sie enthält Informationen über Klassennamen des Providers, der Portlets und die Darstellungsart dieser Portlets. Die wichtigsten Eigenschaften der „provider.xml“ werden im Folgenden beschrieben.

Siehe dazu Abbildung 52.

<provider ...>

Dieser Tag innerhalb der „provider.xml“ besitzt 2 optionale Attribute. Über das Attribut „class (1)“ wird die Java Klasse benannt, welche die Provider Klasse implementiert. Wenn kein Eintrag erfolgt, wird defaultmässig die Klasse

```
<provider class="oracle.portal.provider.v1.http.DefaultProvider">
```

verwendet.

<session ...>

Das Attribut „session“ (2) wird verwendet, um festzulegen, ob ein eine Servlet Sitzung erzeugt werden soll.

<Portlet ...>

Es ist möglich ein oder mehrere Portlets innerhalb einer „provider.xml“ zu deklarieren. Dazu wird das Portlet-Tag (3) verwendet. Für jede Portlet muss ein eignes Portlet-Tag verwendet werden. Innerhalb des Portlet-Tags befinden sich weitere Attribute und Tags.

Attribute:

- `<class>`: Verwendet wie beim Provider-Tag standardmäßig die „portal.provider.v1.http.DefaultPortlet“
- `<resource>`: Gibt den Klassennamen für ein „Resource Bundle“ an. Dabei handelt es sich um eine Java Klasse, die Meta-Daten über das Portlet speichert.
- `<version>`: Aktuelle Version des Portlets

weitere Tags:

<code><id></code>	Nummerierung, muss eindeutig sein
<code><name></code>	Portletbezeichnung, muss eindeutig sein
<code><title></code>	Portletname der im Portal angezeigt wird
<code><description></code>	Portletbeschreibung
<code><timeout></code>	Timeout Zeit
<code><timeoutMessage></code>	Text wenn Portlet Timed out
<code><acceptContentType></code>	Welche Werte nimmt das Portlet auf. Text, HTML, etc.
<code><showEdit></code>	Boolean Wert, Portlet ist personalisierbar, ja / nein
<code><showEditToPublic></code>	Boolean Wert, Portlet ist vom Public- User personalisierbar, ja / nein
<code><showEditDefault></code>	Boolean Wert, Portlet besitzt Edit-Link als default-Wert, ja / nein
<code><showPreview></code>	Boolean Wert, Portlet hat eine Vorschau, ja/ nein
<code><showDetails></code>	Boolean Wert, Portlet zeigt zusätzliche Informationen an, ja / nein
<code><hasHelp></code>	Boolean Wert, Portlet besitzt Hilfe-Link, ja/nein
<code><hasAbout></code>	Boolean Wert, Portlet zeigt zusätzliche Informationen über sich selbst an, ja/nein
<code><renderer class></code>	Gibt die Klasse an, die die Portlets erstellt. Es besitzt ein Attribut für den Klassennamen und 11 weitere Tags, die für die Pfad- und Dateibeschreibung verwendet werden. Die Attribute werden nur in Verwendung mit <i>JSP</i> -Seiten verwendet.
<code><personalizationManager></code>	Gibt die Klasse an, die für die Personalisierung der Portlets zuständig ist. Hat als Attribut den Klassennamen und zwei Tags: <code><dataClass></code> und <code><multiLangStringClass></code>

Tabelle 2: Weitere Tags innerhalb der provider.xml

Die in diesem Beispiel verwendete „porvider.xml“ enthält nicht alle Varianten der möglichen Tags und Attribute. Zum Beispiel ist darin der `<defaultLocale>` Tag, der die Sprache angibt, die eine Portlet standardmäßig verwendet, sowie ein `<portletSecurityManager>` Tag der die Klasse bezeichnet, die den `PortletSecurityManager` implementiert, nicht enthalten.

```
[Quelle: D:/.../JPDK/providers/Sample]

<?providerDefinition version="2.0"?>
<!DOCTYPE provider [
  <!ENTITY virtualRoot "/jpdk/">
  <!ENTITY physicalRoot "D:\oracle\iSuites\Apache\Apache\htdocs\jpdk\">
]>

<provider class="oracle.portal.provider.v1.http.DefaultProvider"> (1)
  <session>true</session> (2)
  <useOldStyleHeaders>>false</useOldStyleHeaders>

  <portlet (3) class="oracle.portal.provider.v1.http.DefaultPortlet">
    <id>1</id>
    <name>SampleRenderer</name>
    <title>SampleRenderer example</title>
    <description>Example portlet rendered using the SampleRenderer</description>
    <timeout>10</timeout>
    <timeoutMessage>SampleRenderer example timed out</timeoutMessage>
    <acceptContentType>text/html</acceptContentType>
    <showEdit>true</showEdit>
    <showEditToPublic>>false</showEditToPublic>
    <showEditDefault>true</showEditDefault>
    <showPreview>true</showPreview>
```

Abbildung 52: Ausschnitt aus der provider.xml

4.6.3 Provider Adapter

Der Provider Adapter ist für die Kommunikation zwischen Oracle Portal und den Web-Providern zuständig. Dabei werden die Anfragen des Oracle Portals vom Adapter übersetzt und an die angesprochenen Web-Provider weitergeleitet. Diese führen die Anfragen aus, senden das Ergebnis an den Adapter zurück, der die Ergebnisse für das Oracle Portal übersetzt und sie an dieses weitergibt. Der Provider Adapter wird implementiert durch folgendes Java *Servlet*:

oracle.portal.provider.v1.http.HttpProvider implementiert

Jedem Web-Provider ist genau ein Adapter zugeordnet, wobei eine Zuordnung auf zwei Arten erfolgen kann.

1. Eine neue Adapter Klasse wird erstellt, die von der Klasse
oracle.portal.provider.v1.http.HttpProvider
 abgeleitet wird und diese Provider-Klasse deklariert.

2. Die *XML*-Datei „provider.xml“, welche die Klassennamen enthält, wird verwendet. Dazu bedarf es eines Eintrags in der vom Apache-Webserver, bei der Initialisierung, verwendeten Datei „zone.properties“. Dieser Eintrag enthält Informationen über den Speicherort der „provider.xml“ Datei. Für den Zugriff des Web-Servers auf den Provider Adapter muss zuerst der Web-Server konfiguriert werden.

Weitere Informationen dazu sind unter [PDK02] zu finden.

4.6.4 Provider Interfaces

Die Provider Interfaces definieren die *API*-Funktionen, die für die weitere Implementierung notwendig sind. Es muss ein Rahmen für die Kommunikation zwischen den Provider Adaptern und den Web-Providern definiert werden. Dieser Rahmen wird durch die Provider Interfaces realisiert. Die Provider bestehen aus 5 *Servlets*, die im Folgenden benannt werden:

<i>Provider interface</i>	oracle.portal.provider.v1.http.Provider
<i>Portlet interface</i>	oracle.portal.provider.v1.Portlet
<i>Portlet Renderer interface</i>	oracle.portal.provider.v1.PortletRenderer
<i>Portlet Personalization Manager interface</i>	oracle.portal.provider.v1. PortletPersonalizationManager
<i>Portlet Security Manager interface</i>	oracle.portal.provider.v1. PortletSecurityManager

Die Provider Interfaces werden in der Provider Runtime implementiert.

4.6.5 Provider Runtime

In der Provider Runtime werden die im oberen Kapitel definierten Interfaces einer Klasse implementiert.

DefaultProvider:

Diese Klasse implementiert das Provider Interface, welches die Verwaltung der Portlets zur Aufgabe hat. Der DefaultProvider wird beim ersten Aufruf von Oracle Portal initialisiert, wodurch er vom Provider Adapter aufgerufen und instanziiert (Erzeugung eines Java Objektes) wird. Die Informationen aus der „provider.xml“ Datei werden anschließend eingelesen und für jedes Portlet eine Instanz erzeugt. Der DefaultProvider legt dabei eine Liste der Portlets an, die bei einer Anfrage verwendet werden können.

DefaultPortlet:

Diese Klasse speichert und verwaltet Informationen über die Darstellungsmodi der Portlets. Sie wird von der DefaultProvider-Klasse aufgerufen und implementiert das Portlet Interface. Diese Informationen werden bei einer Anfrage zurückgegeben und die entsprechenden Klassen aufgerufen. Wobei die eigentliche Portleterstellung vom PageRenderer und DefaultPortlet-PersonalizationManager übernommen wird.

PageRenderer

Die Aufgabe des PageRenderers ist die Erstellung der Portlets. Der PageRenderer kommt nur bei *JSP's* und statischen *HTML*-Dateien zur Geltung und verwendet ebenfalls Informationen aus der „provider.xml“

Um Servlets verwenden zu können, muss die Klasse abgeleitet werden. Wobei die wichtigste Methode dieser Klasse „renderer ()“ ist, die bei der Erstellung eines Portlets aufgerufen wird. Die PageRendrer Klasse implementiert das PortletRenderer Interface.

DefaultPortletPersonalizationManager:

Durch diese Klasse wird die PortletPersonalizationManager Klasse implementiert. Über sie werden Attribute definiert, die im Portlet personalisierbar sein werden. Die personalisierten Daten werden in der Umgebung des Providers in Dateien gespeichert und von dieser Klasse verwaltet.

PortletSecurityManager

Über das Interface dieser Klasse werden Methoden zur Verfügung gestellt, über die Anwenderrechte dem Portlet zugeordnet werden.

4.6.6 Provider Utilities

Die Provider Utilities sind in der Klasse „HttpPortletRendererUtil“ implementiert. In dieser Klasse sind Methoden enthalten, die der Vereinfachung bei der Portletentwicklung und -darstellung dienen sollen. Dazu gehören verschiedene Methoden wie:

<i>Get, set, use</i>	Für Cache-Verwaltung
<i>renderPortletHeader</i> <i>renderPortletTitlebar</i> <i>renderPortletFooer</i>	Für die Erstellung von Kopf- / Fußzeilen innerhalb des Portlets
<i>getEditData</i> <i>submitEditData</i>	Für das Lesen und Schreiben der personalisierten Daten
<i>htmlFormActionLink</i> <i>htmlFormHiddenFiled</i> s	Für die Erstellung und Verwaltung von Web-Forms
<i>SendRedirect</i>	Für Umleitung der Ausgabe

5 Praktische Umsetzung des FH-Portals

In diesem Kapitel wird die praktische Umsetzung des FH-Portals dokumentiert. Dabei wird versucht eine einfache Vorgehensweise zu beschreiben, die es dem Leser ermöglichen soll, die einzelnen Schritte nachzuvollziehen.

5.1 Konzept

Bevor man mit der Erstellung des Portals beginnen kann, sollte der Portalentwickler eine Vorstellung vom Portaldesign, den Benutzerrollen und Inhalten, die ins Portal eingefügt werden sollen.

5.1.1 Design der einzelnen Portalseiten

Am einfachsten ist es, den ungefähren Aufbau grafisch darzustellen (z.B. mit einem Grafikprogramm) um eine ungefähre Vorstellung davon zu haben, wo und welche Tabreiter auf der zu erstellenden Seite erscheinen sollen. Auch die Einteilung der Regionen und des Banners kann man dadurch am besten festlegen.

Für das Demoportale der Fachhochschule werden vorerst drei Seiten benötigt, deren Design auf den im Kap. 3 vorgestellten Anforderungen aufbaut.

1. Design der Startseite



Abbildung 53: Design der FH-Startseite

2. Design der Informatik Homepage

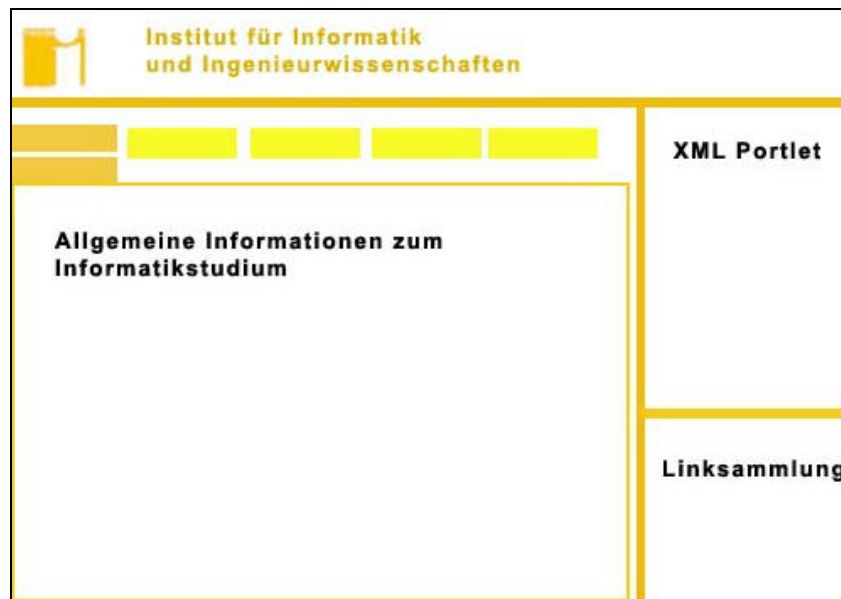


Abbildung 54: Design der Informatik Homepage

3. Design der Professor Homepage



Abbildung 55: Design der Professoren Homepage

5.1.2 Benutzer und Gruppen

Im FH-Demoportal sind folgende Gruppen vorgesehen:

- Portaladministrator
- Professor
- Student
- Besucher

Daraus ergeben sich folgende Benutzer mit bestimmten Rechten:

Administrator:

DARIUS: Volle Zugriffsrechte (Administrator)

PORTAL30: Vom Portal angelegter User mit vollen Zugriffsrechten

Professor:

H_WOYKE: Administrationsrechte der eigenen Seite

Und Editierrechte der Fachbereichs-Seite. Hier nur **Informatikbereich**.

Student:

STUDENT1: Erweiterte Sicht nach Anmeldung auf der Professorseite

Besucher:

Jeder Portalbesucher, der nicht mit Namen und Passwort eingeloggt ist.

Zugang zum Hochschulportal:

Der Zugang erfolgt für alle Benutzer über dieselbe *URL* [FHPOR]:

<http://lwis01.inf.fh-loeln.de:7778>

Nach erfolgreicher Login-Prozedur, wird der Benutzer auf die für ihn vordefinierte Startseite geleitet.

Im Einzelnen gelangt er auf:

Administrator: Oracle Portal Administratorseite

Professor: Seine Persönliche Professoren Homepage

Student: Defaulteinstellung ist die allgemeine Startseite des FH-Portals. Es kann allerdings vom Administrator eingestellt werden, dass z.Bsp. Alle Informatik Studenten direkt auf die Informatik Homepage gelangen, ohne den Informatikbutton auf der Startseite klicken zu müssen.

Besucher ohne Anmeldung: Allgemeine Startseite des FH-Portals

5.1.3 Inhalte

Im diesem praktischen Teil beschränkt sich die Funktionalität des FH-Portals nur auf Inhalte, die der Information der Studenten und Portalbenutzer dienen. Es sind keine, der im Kap. 3 erwähnten Funktionen realisiert worden.

FH-Portalseite: Startseite

- Allgemeine Informationen über das Studium, und die einzelnen Studiengänge
- XML-News Portlet
- Links zum Thema Studium und die örtliche Umgebung.

Institut für Informatik und Ingenieurwissenschaften Startseite:

- Allgemeine Informationen zum Institut für Informatik.
- Namensliste aller in diesem Bereich tätigen Professoren und Verlinkungen zu ihren privaten Homepages.

Wobei der Link von Prof. Dr. H. Faeskorn – Woyke auf eine Seite innerhalb des Portals führt. An dieser privaten Userpage soll exemplarisch dargestellt werden, wie jeder einzelne Professor seine Homepage mit Hilfe von Oracle Portal gestalten und verwalten kann.

- Linksammlung zum IT-Bereich
- XML-Portlet mit Namen und Studiengang aller Zusatzseminar –Teilnehmer

Professorseite:

- Persönliche Inhalte und Informationen für die Studenten
- Beispielanwendung

5.2 Erstellung von Seiten und Benutzern**5.2.1 Erstellung der Startseite des FH-Portals****5.2.1.1 Layoutvorlagen erstellen**

Bevor mit der Erstellung der Portalseite begonnen wird, sollte das Layouts und der Stil für die jeweilige Seite, gemäß der im Kap. 5.1.1 vorgefertigten Designs erfolgen. So kann später auf einfache Art jeder Seite das entsprechende „look and feel“ zugeordnet werden.

5.2.1.1.1 Seitenstil erstellen

Durch klicken auf „Create a New Page Style“ im „**create your own Portal Pages**“ Portlet, gelangt man zu einem Assistenten der den Benutzer bei Erstellung eines eigenen Seitenstiles unterstützt.

Schritt	Funktion	Bemerkung
1	Namen vergeben, Cachingart auswählen	FH_GM_PORTAL_STYLE
2	Farben und Schriftart für die einzelnen Seitenelemente bestimmen	

5.2.1.1.2 Seitenlayout erstellen

Durch klicken auf „Create a New Page Layout“ im „**create your own Portal Pages**“ Portlet, gelangt man zu einem Assistenten der den Benutzer bei Erstellung eines eigenen Seitenlayouts unterstützt.

Schritt	Funktion	Bemerkung
1	Namen vergeben	FH_GM_PORTAL_LAYOUT
2	Festlegung der Anzahl und Aufteilung der Regionen	[siehe Bsp. Abb. 56]



Abbildung 56: Menü zur Erstellen von Regionen

5.2.1.2 Seite erstellen

Durch klicken auf „Create a New Page“ im „**create your own Portal Pages**“ Portlet, gelangt man zu einem Assistenten, der den Benutzer bei der Erstellung einer neuen Portalseite unterstützt.

Schritt	Funktion	Bemerkung
1	Namen vergeben, Banner-Vorlage auswählen, Caching Optionen	FH_GM_PORTAL_STARTPAGE
2	Erstelltes Layout und Style zuordnen	Siehe Abb. 57
3	Seitenbanner gestalten (3-a) und Seiteninhalte (3-b) hinzufügen	Siehe dazu Abb. 58
4	Zugriffsrechte auf diese Seite vergeben	erfolgt später

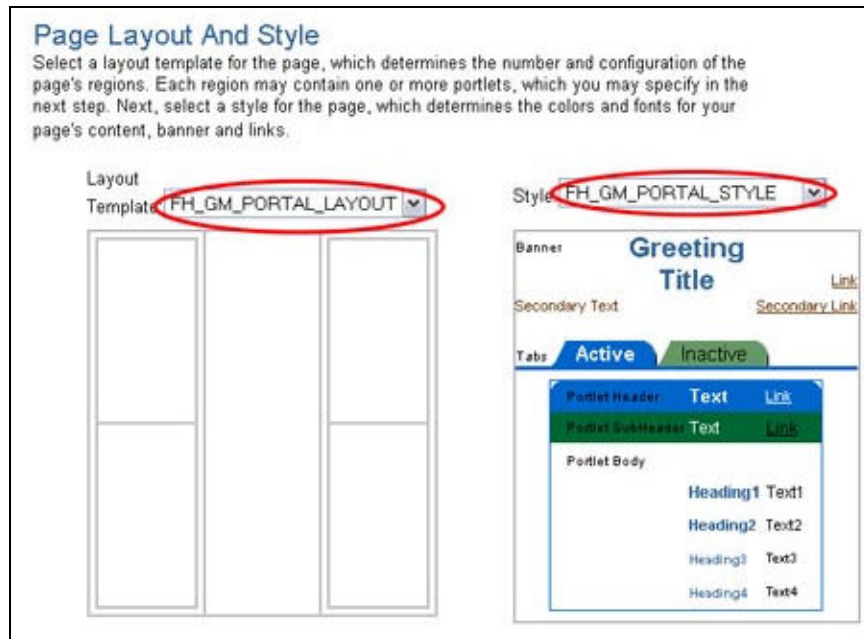


Abbildung 57: Layouts und Stile zuordnen

3-a: Seitenbanner¹⁹

Durch anklicken von „edit Defaults“ im Schritt 3, gelangt man zu in ein Menü, indem sich der Seitenbanner individualisieren lässt (siehe (1) in Abb. 58). Es können eigene Grafiken und Logos eingefügt werden, ausgewählt welche Default-Links im Banner erscheinen sollen oder sogar eigene *URL* 's können eingefügt werden, die dann im Banner sichtbar sind.

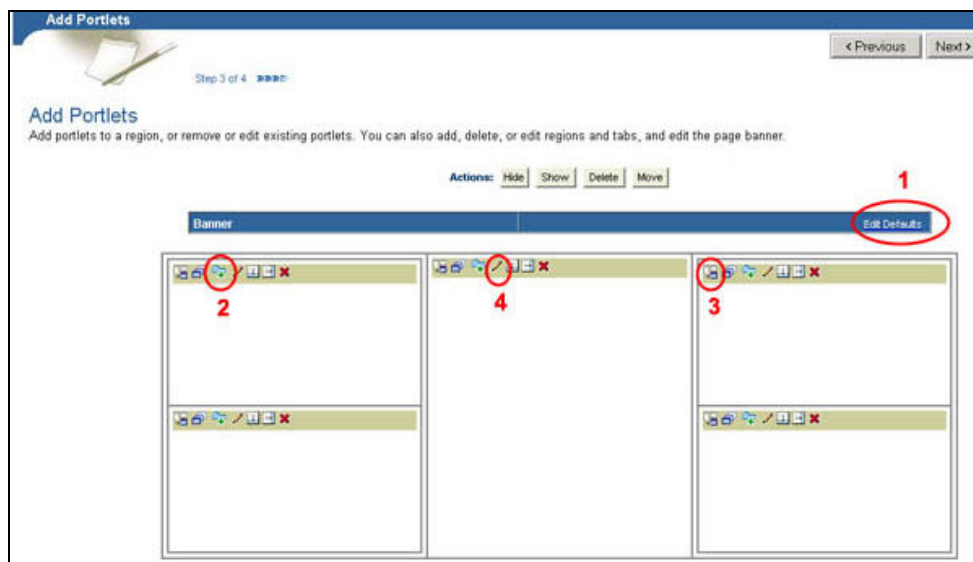


Abbildung 58: Banner-Edit Menü²⁰

²⁰ Damit lässt sich der Portalbanner individuell anpassen und die Seite mit Portlets füllen

Ein eigenes Logo wird in das Portal geladen:

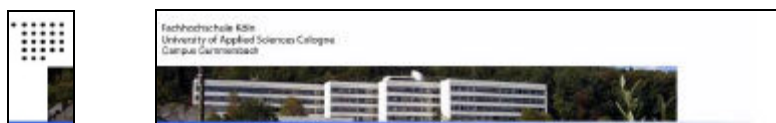


Abbildung 59: Linker- und rechter Teil des Logos der FH-Startseite

3-b: Eigene Portalseite mit Inhalten füllen

Im Schritt 3 können ebenfalls alle Regionen einzeln mit Tabreitern (Abb. 58 - (2) und Portlets (3) bestückt werden.

Durch Anklicken des „Stiftsymbols“ (4) kann die prozentuelle Aufteilung der einzelnen Regionen festgelegt werden. Das ist dann sinnvoll, wenn die Portalseite über Inhalte mit geringer Darstellungsgröße verfügt (Bsp. kurzer *HTML*-Text), aber im Portal die voreingestellte Einteilung viel Leerraum darstellen würde.

5.2.2 Homepage der Informatik Fakultät

Die Vorgehensweise ist die gleiche, wie im Kapitel 5.2.1 beschreiben, es werden jedoch Namen vergeben, mit denen sich der Stil, das Layout und der Seitenname eindeutig identifizieren lassen.

5.2.2.1 Layoutvorlagen erstellen

5.2.2.1.1 Seitenstil erstellen

Schritt	Funktion	Bemerkung
1	Namen vergeben, Cachingart auswählen	FB_INFORMATIK_STYLE
2	Farben und Schriftart für die einzelnen Seitenelemente bestimmen	

5.2.2.1.2 Seitenlayout erstellen

Schritt	Funktion	Bemerkung
1	Namen vergeben	FB_INFORMATIK_LAYOUT
2	Festlegung der Anzahl und Aufteilung der Regionen	

5.2.2.2 Seite erstellen

Schritt	Funktion	Bemerkung
1	Namen vergeben, Banner-Vorlage auswählen, Caching Optionen	FB_INFORMATIK_STARTPAGE
2	Erstelltes Layout und Style zuordnen	
3	Seitenbanner gestalten (3-a) und Seiteninhalte (3-b) hinzufügen	
4	Zugriffsrechte auf diese Seite vergeben	erfolgt später

3-a: Seitenbanner

Ein eigenes Logo wird in das Portal geladen:



Abbildung 60: Teil des Logos der Informatik-Homepage

3-b: Portalseite mit Inhalten füllen

Die im Kap. 5.4 erstellten Portlets könnten hier integriert werden.

5.2.3 Homepage des FH-Personals

In unserem Beispiel wird nur die persönliche Homepage für einen Professor erstellt. Für das FH-Portal heißt das jedoch, dass für jeden Professor, Mitarbeiter oder Lehrbeauftragten nach folgendem Schema eine eigene Portalseite eingerichtet werden kann.

5.2.3.1 Layoutvorlagen erstellen

5.2.3.1.1 Seitenstiel erstellen

Schritt	Funktion	Bemerkung
1	Namen vergeben, Cachingart auswählen	FB_INF_PROF_STYLE
2	Farben und Schriftart für die einzelnen Seitenelemente bestimmen	

5.2.3.1.2 Seitenlayout erstellen

Schritt	Funktion	Bemerkung
1	Namen vergeben	FB_INF_PROF_LAYOUT
2	Festlegung der Anzahl und Aufteilung der Regionen	

5.2.3.2 Seite erstellen

Schritt	Funktion	Bemerkung
1	Namen vergeben, Banner-Vorlage auswählen, Caching Optionen	FB_INF_PROF_STARTPAGE
2	Erstelltes Layout und Style zuordnen	
3	Seitenbanner gestalten (3-a) und Seiteninhalte (3-b) hinzufügen	
4	Zugriffsrechte auf diese Seite vergeben	Erfolgt später

3-a: Seitenbanner

Ein eigenes Logo wird in das Portal geladen:



Abbildung 61: Teil des Logos der Professoren Homepage

3-b: Portalseite mit Inhalten füllen

Diese Startseite wird nicht mit Inhalten gefüllt. Sie soll den Professoren lediglich als ein Gerüst dienen, auf dem sie ihre individuelle Homepage aufbauen und editieren können.

5.2.4 Benutzergruppen erstellen

Durch klicken auf den „Administer“ Tab gelangt man in die Administrations-Region von Oracle Portal. Dort klickt man im „Group“ Portlet auf „Creat New Groups“. Dadurch gelangt man zu einem Assistenten, der dem Portalentwickler bei Erstellung von Benutzergruppen unterstützt.

Es wird die Gruppe „Professoren“ erstellt. Der Name lautet hierbei:

INF_PROF_GROUP

Unter „Group Home Page“ wird die Startseite definiert, auf die Benutzer beim erfolgreichen einloggen weitergeleitet werden. Da in unserem Beispiel nur ein Benutzer mit der Professoren Rolle erstellt wird, ist dieser Vorgang nicht unbedingt notwendig.

In diesem Menü können beispielsweise allen Professoren der Informatik Fakultät, die gleichen Zugriffsrechte auf die Informatik Homepage gegeben werden, die bei einer späteren Erstellung von einzelnen Benutzern nicht explizit vergeben müssen.

Siehe dazu Abb. 62.

Section	Object Type	Privileges
Page Privileges	All Pages	Customization(Full)
	All Styles	Create
	All Layouts	Create
	All Providers	Publish
	All Portlets	Publish
Content Area Privileges	All Content Areas	Create
Application Privileges	All Applications	Create
	All Shared Components	Manage
Administration Privileges	All User Profiles	Edit
	All Groups	Create
	All Schemas	None
	All Logs	None

Abbildung 62: Vergabe von Zugriffsrechten für die Gruppen

5.2.5 Benutzer erstellen

Durch klicken auf den „Administer“ Tab gelangt man in die Administrations Region von Oracle Portal. Dort klickt man im „User“ Portlet auf „Creat New User“. Dadurch gelangt man zu einem Assistenten der dem Portalentwickler bei Erstellung neuer Benutzer unterstützt.

5.2.5.1 Benutzer mit Professorenrolle erstellen

In diesem Menü werden folgende Angaben zu den Userdaten eingegeben:

- Benutzernamen: USER_H_WOYKE
- Passwort: user1

- Gültigkeitsdauer dieses User-Accounts kann festgelegt werden, wenn dieses nötig sein sollte.

- Login Server Privilegien festlegen: Enduser oder Full Administrator

Sobald die eingegebenen Daten an die Datenbank gesendet wurden erscheint folgende Zeile:

Click USER_H_WOYKE to edit the portal user's profile.

Hier können noch weitere Angaben und Rechte eingegeben werden.

Unter dem Tab „Main“:

- Personal Details
- Portal Privileges
- Group Membership
- Database Schema
- Organizational Details
- Photograph

Unter dem Tab „Preferences“:

- Additional Personal Details
- Personal Preferences
- Default Group: INF_PROF_GROUP
- Default Home Page: FB_INF_PROF_STARTPAGE

Unter dem Tab „Privileges“:

- Einzelne Privilegien des Users festlegen für:
 - Seiten
 - Content Areas
 - Anwendungen
 - Administration

Nun kann dem Benutzer „USER_H_WOYKE“ die im Kap. 5.2.3 erstellte Seite, als Startseite zugeordnet werden. Dabei geht man folgendermaßen vor:

Durch klicken auf den „Build“ Tab gelangt man in die Portalentwickler Region von Oracle Portal. Dort klickt man unter „Edit Page“ auf das „Browse“ Icon. Im neu geöffneten Fenster, wählt man unter „My Pages“ die Seite „FB_INF_PROF_STARTPAGE“ mit Select. Der gewählte Name erscheint im Auswahlfenster unter „Edit Page“. Durch klicken auf „Edit“ gelangt man ins Konfigurationsmenü dieser Seite.

Unter dem Tab „Access“ gewährt man dem Benutzer „USER_H_WOYKE“ volle MANAGE-Rechte auf diese Seite.

5.2.5.2 Benutzer mit Studentenrolle erstellen

In diesem Menü werden folgende Angaben zu den Userdaten eingegeben:

- Benutzernamen: USER_STUDENT1
- Passwort: student
- Gültigkeitsdauer dieses User-Accounts kann festgelegt werden wenn dieses nötig sein sollte. Bei Studenten ist es sogar eine sinnvolle Funktion.
- Login Server Privilegien festlegen: Enduser

Sobald die eingegebenen Daten an die Datenbank gesendet wurden erscheint folgende Zeile:

Click `USER_STUDENT1` to edit the portal user's profile.

Hier können noch weitere Angaben und Rechte eingegeben werden.

Unter dem Tab „Main“:

(wie unter 5.2.5.1)

Unter dem Tab „Preferences“:

(wie unter 5.2.5.1)

- Default Home Page: `FH_GM_PORTAL_STARTPAGE`

Unter dem Tab „Privileges“:

(wie unter 5.2.5.1)

5.3 Content Area Integration

Die Content Areas können vielseitig innerhalb des Portals eingesetzt werden. Siehe dazu Kap. 4.3.1. Für das exemplarische Beispiel dieses Kapitels wird eine Content Area erstellt, die der Speicherung verschiedener Dateien dienen soll. Beispielsweise können Vorlesungsskripte vom Professor dort abgelegt werden, damit die Studenten sich diese bei Bedarf herunterladen können.

5.3.1 Neue Content Area erstellen

Bei der Erstellung einer Content Area geht man folgendermaßen vor:

In der „Build“ Region der Administrations Seite vom Oracle Portal klick man im Content Area Portlet auf „**Create a New Content Area**“ Siehe dazu auch Abb. 36

Alternativ dazu kann über den Navigator und den Tab „Content Area“ dasselbe Resultat hervorgerufen werden.

Im Menü werden der Name, die Sprache und der Administrator dieser Content Area bestimmt.

In diesem Beispiel sind folgende Angaben gemacht worden:

Name: CONTENT_VORLESUNG

Language: ENGLISH

Content Area Administrator: USER_H_WOYKE

Nach der Bestätigung dieser Angaben durch klick auf „create“, erscheint folgende Zeile:

„Click **CONTENT_VORLESUNG** to edit the content area.“

Das dadurch anwählbare Menü ermöglicht eine persönliche Anpassung der eben erstellten Content Area. Im Folgenden werden nur Auszüge aus den Personalisierungsoptionen erwähnt.

Unter dem Tabreiter „**Main**“ kann z.Bsp. ein eigenes Logo eingespielt werden.

Unter dem Tabreiter „**Labels**“ kann bestimmt werden, welche Default-links die Content Area beinhalten soll und welche Symbole dafür angezeigt werden sollen.

Beispielsweise: Log On / Log off (Schlüsselsymbol), Home Link (Haussymbol) etc.

Unter „**Access**“ werden die Benutzer bestimmt, die zusätzlich zum Administrator und dem Benutzer USER_H_WOYKE selbst, auf diese Area Zugriff haben sollen.

Da die Inhalte für die Studenten zugänglich sein sollen, wird der Benutzer:

USER_STUDENT1

zusätzlich mit „Make Public“ Privilegien aufgenommen

Nach Bestätigung der Angaben findet man sich im Navigator-Menü wieder. Nun klickt man auf den Namen der eben erstellten Content Area.

5.3.2 Inhalte in Content Area einspielen

Bei der nun erscheinenden Seite handelt es sich um den vom Benutzer erstellten Content Bereich, den er über „Edit folder“ (siehe Abb. 63) mit Inhalten füllen kann.



Abbildung 63: Neu erstellte Content Area.

Für dieses Beispiel wird eine *PDF*-Datei in die Content Area eingespielt mit dem Namen „JDBC-Folien im *PDF*-Format“

Nach dem Klick auf „Edit folder“ wählt man „Add item“ im neu eingeblendeten Menü.

Zur Auswahl des Item-Typs steht folgendes bereit: Siehe dazu Abb. 64 und Kap. 4.3.1.2

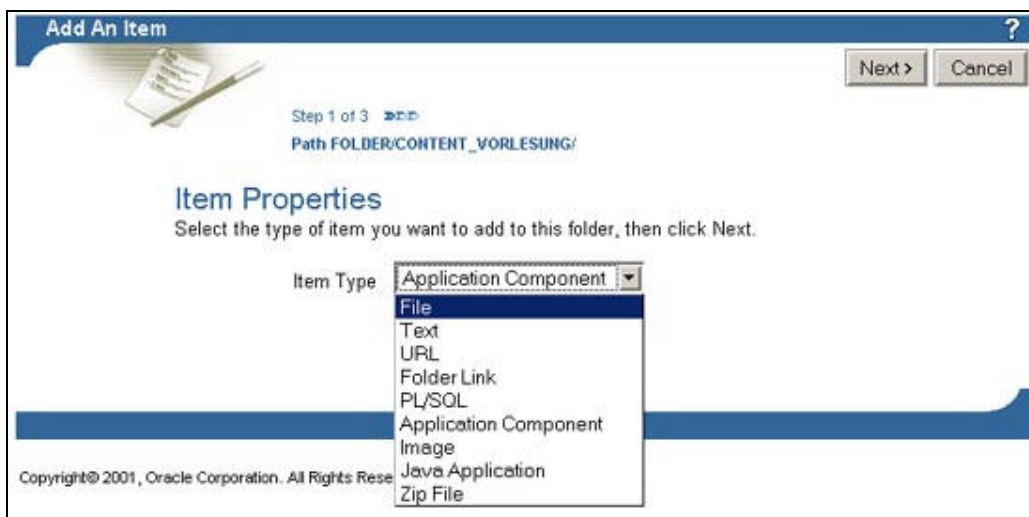


Abbildung 64: Auswahlmenü der verschiedenen Item Typen

Da es sich um eine *PDF*-Datei handelt, wählt man hier den Typ: „File“

Im nächsten Menü wird mit „Browse“ der aktuelle Speicherort der Datei gewählt, ein Name unter dem die Datei in der Content Area angezeigt werden soll, wenn notwendig eine

Beschreibung des Inhaltes und das Datum, wie lange die Datei verfügbar sein soll (default: keine Beschränkung)

mit „Finish“ wird die PDF-Datei in die Datenbank geladen und ist sofort in der Content Area verfügbar.

5.3.3 Content Area als Portlet verfügbar machen

Da die Inhalte der Content Area für die Studenten innerhalb der Professoren Seite verfügbar sein sollen, muss die neu erstellte Content Area zuerst als Portlet verfügbar gemacht werden.

Dazu klickt man im „Navigationsmenü“ unter dem Tab „Content Areas“ unter Actions auf den Link „Edit Root Folder“ neben dem Namen der neu erstellten Area. Siehe dazu Abb. 65



Type	Name	Actions	Owner	Modified
Content Area	CONTENT_VORLESUNG	Contents, Edit Properties, Delete, Edit Root Folder , Copy Root Folder	USER_HJMOVKE	11-Nov-2002

Abbildung 65: Content Area Verzeichnis mit Edit-Möglichkeit

Nun muss lediglich ein „Häkchen“ bei „Publish As Portlet“ gesetzt werden und mit „OK“ bestätigt werden.

Dadurch wird diese Content Area dem Portlet repository hinzugefügt.

5.3.4 Content Area ins Portal integrieren

Auf der Professoren Startseite wird der Link „Edit Page“ oder „Customize“ gewählt (vergleiche dazu auch Abb. 69 Markierung (2)). In der Regel sind diese Links im Seitenbanner als Defaulteinstellung sichtbar. Damit gelangt man zum „Edit Menü“ der jeweiligen Seite. Dort kann nun die Content Area oder jedes andere verfügbare Portlet in der entsprechenden Region, innerhalb der Portalseiten, aus dem Portlet repository ausgewählt und eingefügt werden. Durch Klicken auf das in Abb. 66 markierte Icon, gelangt man zum Portlet-Auswahl-Menü. Nach Auswahl und Bestätigung, sollte der Portletname in der entsprechenden Region erscheinen.

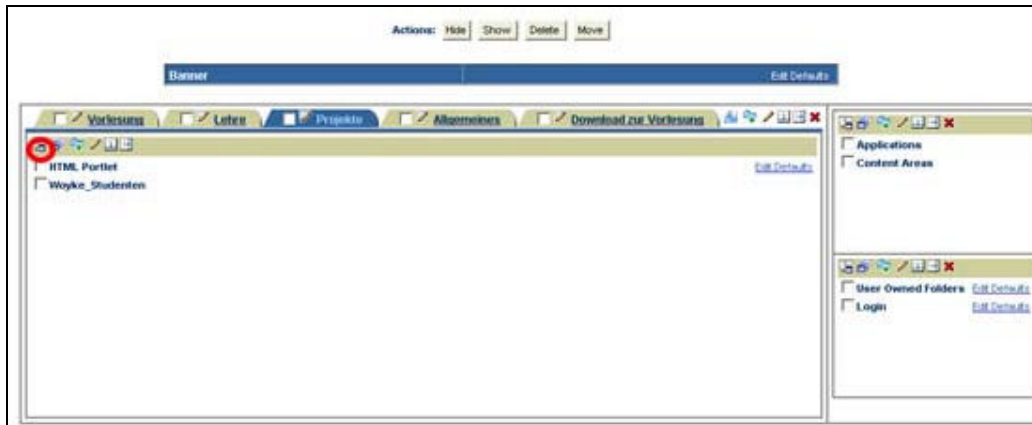


Abbildung 66: Menü zum Einfügen von Portlets

5.4 Erstellung von Anwendungen für das FH-Portal

Nachdem nun Standardbenutzer erstellt worden sind, kann mit der Erstellung von Anwendungen begonnen werden. Im Rahmen dieser Diplomarbeit werden vier Anwendungen erstellt und ins Portal integriert. Daran soll verdeutlicht werden, wie einfach und vielseitig das Portal und seine Funktionalitäten eingesetzt werden können.

Für zwei der Beispiele werden fiktive Studentendaten benötigt. Dazu muss zuerst eine Tabelle erzeugt werden, in der diese Werte gespeichert werden sollen.

Mit diesen Werten werden dann ein Diagramm und ein Report im Portal erzeugt und angezeigt.

Datenbankschema:

Für dieses Beispiel ist eine einfache Tabelle notwendig. Hier wird das bei der Installation erstellte „Scott“ Schema in das, per *SQL*-Befehl eine Studententabelle erstellt wurde verwendet. Diese beinhaltet folgende Spalten.

(Name, Vorname, Studiengang, Geschlecht)

Der Zugriff erfolgt über die Eingabe folgenden Pfades innerhalb des Portals. „SCOTT.STUDENTEN“.

Benutzerrechte:

Bei der Erzeugung der Gruppen und Benutzer wurden gleich die Rechte für die Erzeugung von Anwendungen vergeben. Es ist möglich, bestimmten Benutzern noch zusätzliche Rechte nachträglich zu geben.

5.4.1 Anwendungsentwicklung

Um die Studententabelle mit Werten zu füllen, gibt es verschiedene Möglichkeiten. Zum einen können die willkürlichen Werte über *SQL-Plus* dort eingespielt werden oder zum anderen direkt über eine Eingabemaske innerhalb des Portals vom berechtigten Benutzer eingegeben werden.

Wir werden in diesem Fall eine Eingabemaske mit Hilfe des im Kap 4.3.2.1 beschriebenen Formularassistenten erzeugen.

Dabei wird der Anwendungssteller der Benutzer: USER_H_WOYKE sein.

Um mit der Entwicklung der ersten Anwendung zu beginnen, muss zuerst vom Portaladministrator ein Applikationsverzeichnis erstellt und für den Benutzer freigegeben werden. Darunter kann man sich eine Art Behälter für bestimmte Benutzergruppen vorstellen, in den sie ihre selbst erstellte Applikationen speichern können.

Gleichzeitig wird mit der Erstellung des Verzeichnisses das Datenbankschema auf das vom Benutzer oder Benutzergruppe Zugegriffen werden darf zugeordnet.

In unserem Beispiel also das „SCOTT“ Schama.

Name des Applikationsverzeichnisses lautet: WOYKE_APP1

Am Portal als Benutzer USER_H_WOYKE oder Administrator angemeldet, kann mit der Anwendungsentwicklung begonnen werden.

Dazu klick man auf das Navigations-Icon in der Kopfzeile des Portals.

Anschließend auf den Applications-Registerreiter.

Bei den nun aufgeführten Anwendungen muss der Entwickler die für ihn erstellte Anwendungsumgebung wählen. Hier: WOYKE_APP1

Dadurch wechselt er in das Verzeichnis und kann hier alle von ihm erstellten Anwendungen einsehen, testen und editieren oder neue erstellen.

5.4.1.1 Anwendung: Eingabemaske

Da zuerst das erwähnte Eingabeformular erstellt werden soll, klickt man auf den „**Create New... FORM**“ Link der sich über den aufgelisteten Anwendungen befindet.

Im nächsten Bild klickt man auf den Link

„**Form based on table or view**“ klicken, da eine Anwendung erstellt werden soll, die auf einer Tabelle beruht.

Im nachfolgenden Assistenten wird der Anwender in 7 Schritten durch Menüs geführt, die es ihm erlauben seine Anwendung nach individuellen Vorstellungen anzupassen.

Schritt	Funktion	Bemerkung
1	Namen der Applikation vergeben	WOYKE_STUDENTEN
2	Auswahl der des Datenbankschemas und Tabelle für diese Applikation	SCOTT.STUDENTEN
3	Darstellungsart Layout bestimmen	Tabellarisch
4	Layout festlegen	
5	Layout festlegen	
6	Auswahl eines Rahmenlayouts der die Applikation bei Anwendung umgibt	
7	Zusätzlicher PL-SQL Code vor, während und nach der Applikationsausführung.	

„Finish“

Über Run kann das eben erstellte Formular angeschaut werden.

Damit es als Portlet ins Portal integriert werden kann, muss es nur noch dem Portal als Portlet bekannt gemacht werden. Dies geschieht indem man auf den „Access“ Tabreiter klickt“ und im Feld **Publish to Portal** ein Häkchen setzt.

„Apply“ und „Close“ wählen.

Zurück in das Navigatorfenster.

Im nächsten Schritt wird das eben erstellte Formular, zur Eingabe von Studentendaten, ins Portal als Portlet auf der Professorenseite eingebunden.

Dazu muss der Seiteneditor benutzt werden. In diesem Menü wird das Portlet über das Icon „Portlet hinzufügen“ aus dem vorhandenen Portlets ausgewählt und an der gewünschten Stelle innerhalb der Portalbereiche platziert.

Siehe dazu Kap. 5.3.4

Keht man auf die Portalseite zurück sieht man das eben eingefügte Portlet in dem entsprechenden Bereich. Siehe Abb. 67



Abbildung 67: Anwendung: Eingabemaske zur Erfassung von Studentendaten

Nun können Daten in die Studententabelle eingefügt werden. Dabei ist zu beachten, dass in diesem einfachen Beispiel keine Fehlerbehandlung erfolgt und somit auf Groß- und Kleinschreibung geachtet werden sollte.

Der Inhalt sieht nun folgendermaßen aus (Abb. 68):

NAME	VORNAME	STUDIENGANG	GESCHLECHT
Langner	Darius	WI	M
Waldorf	Maria	WI	W
Neumann	Karl	AI	M
Assmann	Heike	TI	W
Koenig	Willhelm	AI	M
Bauer	Anna	WI	W
Schmitt	Katja	WI	W
Wilk	Karen	AI	W
Blod	Heinz	AI	M
Lutz	Katja	WI	W

Abbildung 68: Inhalt der „Studenten“ Tabelle²¹

²¹ Darstellung im TOAD-Tool.

5.4.1.2 Anwendung: Studentenauswertung

Nachdem die Studententabelle mit Werten gefüllt wurde, kommen wir zur Erstellung einer Anwendung, mit der sich ermitteln lässt wie viele Studenten im Informatikbereich eingeschrieben sind. Nach *AI*, *TI* und *WI* aufgeteilt.

Dazu geht man wieder über den Navigator in das Applikationen Verzeichnis `WOYKE_APP1` und wählt dieses mal **„Create new ... Chart“**

Da einen Diagramm mit Hilfe eines Assistenten erzeugen werden soll, wählt man aus dem Menü: **Charts From Query Wizard**

Im nachfolgenden Assistenten wird der Anwender in 8 Schritten durch Menüs geführt, die es ihm erlauben, seine Anwendung nach individuellen Vorstellungen anzupassen.

Schritt	Funktion	Bemerkung
1	Namen der Applikation vergeben	WOYKE_FB_CHART
2	Auswahl des Datenbankschemas und der Tabelle für diese Applikation	SCOTT.STUDENTEN
3	Table or View Columns bestimmen welche Spalte die Werte liefern soll	Studiengang: FB_1
4	Column Conditions	
5	Display Options: Breite und Höhe des Diagramms kann bestimmt werden	
6	Customization Form Display Options	
7	Auswahl eines Rahmenlayouts der die Applikation bei Anwendung umgibt	
8	Zusätzlicher PL SQL Code vor während und nach der Applikationsausführung.	

„Finish“

Über **„Run“** kann das eben erstellte Formular angeschaut werden.

Damit es als Portlet ins Portal integriert werden kann, muss es nur noch mit dem Portal als Portlet bekannt gemacht werden. Dies geschieht indem man auf den „Access“ Tabreiter klickt“ und im Feld **Publish to Portal** ein Häkchen setzt.

„Apply“ und „Close“

Zurück in das Navigatorfenster.

Im nächsten Schritt wird die Anwendung ins Portal als Portlet auf der Professorensite eingebunden.

Siehe dazu Kap.: 5.3.4

Dazu muss der Seiteneditor benutzt werden. In diesem Menü wird das Portlet über das Icon „Portlet hinzufügen“ aus dem vorhandenen Portlets ausgewählt und an der gewünschten Stelle innerhalb der Portalbereiche platziert.

Keht man auf die Portalseite zurück sieht man das eben eingefügte Portlet in dem entsprechenden Bereich. Siehe Abb. 71.

5.4.1.3 Anwendung: Studentenauswertungs-Report

Damit der Inhalt der Studenten Tabelle auch ohne *SQL-Plus* angezeigt werden kann, wird im Folgenden beschreiben, wie eine Report zum auslesen und darstellen der Inhalte erstellt werden kann.

Dazu gehen man wieder erneut über den Navigator in das Applikationen Verzeichnis WOYKE_APP1 und wählt dieses Mal **Create new ... Report**

Da ein Report mit Hilfe des Assistenten erzeugen werden soll, wählt man:

Query By Example (QBE) Reports

Im Nachfolgenden Assistenten wird der Benutzer in 9 Schritten durch Menüs geführt, die es ihm erlauben die Anwendung nach individuellen Vorstellungen anzupassen.

Schritt	Funktion	Bemerkung
1	Namen der Applikation vergeben	WOYKE_STUDENT_REPORT
2	Auswahl des Datenbankschemas und der Tabelle für diese Applikation	SCOTT.STUDENTEN
3	-	Schritt wird ausgelassen
4	Table or View Columns: Hier werden die Spalten definiert, die im Report dargestellt werden sollen	STUDENTEN.NAME und STUDENTEN.STUDIENGA NG
5	Column Formating: Einstellen, wie die Daten in der Tabelle dargestellt werden sollen.	In diesem Beispiel sollen die Studiengänge mit unterschiedlichen Farben gekennzeichnet werden und der Studiengang „TI“ zusätzlich mit einer dickeren Schriftart. [Report_Condit]
6	Display Options	
7	Customization Form Display Options	
8	QBE Report and Customization Form Text : Zusätzlicher Text für Kopf und Fußzeile	
9	Zusätzlicher PL SQL Code vor während und nach der Applikationsausführung.	

„Finish“

Über **Run** kann der eben erstellte Report angeschaut werden.

Damit es als Portlet ins Portal integriert werden kann, muss es nur noch dem Portal als Portlet bekannt gemacht werden. Dies geschieht indem man auf den „Access“ Tabreiter klickt“ und im Feld **Publish to Portal** ein Häkchen setzt.

Im nächsten Schritt wird der eben erstellte Report im Portal als Link für alle Studenten zugänglich gemacht.

Dazu wird in der Content Area ein Link hinzugefügt in dessen *URL*-Feld die Adresse des Reports hineinkopiert wird. Als Namen für den Link wird „*Liste aller Informatik-Studenten*“ gewählt.

Vergleiche dazu Kap.: 5.3.2

Nach verlassen des Editier Modus, kann der Report durch Anwählen des Links in der Content Area angezeigt werden.

5.4.1.4 Anwendung: XML-Datenbankabfrage

Im folgenden Beispiel soll ein *XML*-Portlet erstellt werden, mit dem es möglich ist Teilnehmer an bestimmten Kursen anzuzeigen. In diesem Fall sollen alle *WI*-Studenten automatisch als Teilnehmer fungieren.

Dazu gehen man erneut über den Navigator in das Applikationen Verzeichnis `WOYKE_APP1` und wählt: **Create new ... XML Component**

Im Assistenten folgen 9 Schritte, wobei nur die ersten vier für das Beispiel von Bedeutung sind.

1 → Der Name für diese Applikation lautet: `XML_TeilnehmerWI`

2 → Hier wird der XML Quellcode eingegeben:

```
<?xml version="1.0"?>
<CATALOG>
  <KURS>
    <TAG>Montag</TAG>
    <DATUM>10.10.02</DATUM>

    <ORACLE>
select Name||', '|| substr(vorname,1,1), Studiengang from studenten where Studiengang ='WI';
    </ORACLE>
  </KURS>
</CATALOG>
```

3 → Hier kann die *SQL*-Anweisung die sich im <ORACLE> Tag befindet editiert werden. Ist aber nicht notwendig, da sie mit unserem *XML*-Code eingefügt wurde.

4 → Hier wird der XSL Quellcode eingegeben:

```
<?xml version="1.0"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
  <html>
  <body>
    <table border="1" bgcolor="orange">
      <tr>
        <th>Tag</th>
        <th>Datum</th>
        <th>Teilnehmer</th>
      </tr>
      <xsl:for-each select="CATALOG/KURS">
        <tr>
          <td><xsl:value-of select="TAG"/></td>
          <td><xsl:value-of select="DATUM"/></td>
          <td><xsl:value-of select="ORACLE"/></td>
        </tr>
      </xsl:for-each>
    </table>
  </body>
</html>
</xsl:template>
</xsl:stylesheet>
```

5 → Hier wird der DTD Quellcode eingegeben: In unserem Beispiel bleibt das Feld leer, da keine besonderen Darstellungsformen notwendig sind.

6 → Full Page Display Options.

Hier kann bestimmt werden ob die Log Aktivität und das Timing angezeigt werden sollen, sowie eine Expire Zeit eingestellt werden.

7 → Customization Form Display Options

8 → Customization Form Text

9 → Additional PL/SQL Code → "Finish"

Über „**Run**“ kann der eben erstellte XML-Abfrage angeschaut werden.

Damit es als Portlet ins Portal integriert werden kann, muss es nur noch mit dem Portal als Portlet bekannt gemacht werden. Dies geschieht indem man auf den „Access“ Tabreiter klickt“ und im Feld **Publish to Portal** ein Häkchen setzt.

Im nächsten Schritt wird das XML-Portlet auf der Informatik Homepage eingefügt, damit es permanent angezeigt werden kann.

Siehe Kap. 5.3.4

Dazu navigiert man auf die FB_INFORMATIK_Portalseite und begibt sich in den „Edit Page“ mode. Nun klickt man in dem gewünschten Bereich in dem das XML-Portlet erscheinen soll auf „**add Portlet**“ und wählt das eben erstellte Portlet aus der Liste aus. Durch „Close“ gelangt man wieder auf die Portalseite und kann das Ergebnis der Abfrage unverzüglich sehen.

5.4.1.5 Anwendung: XML-News

Mit dem folgenden XML-Portlet sollen Neuigkeiten auf der Startseite angezeigt werden, wobei ihr Inhalt in einem separaten XML-Dokument gepflegt werden soll.

Dazu geht man erneut über den Navigator in das Applikationen Verzeichnis WOYKE_APP1 und wählt dieses mal

Create new ... XML Component,

1 → Der Name für diese Applikation lautet: XML_FH_NEWS

2 → Hier wird der XML Quellcode eingegeben:

Nachrichten und Texte müssen in den XML-Tag, an dieser Stelle eingegeben werden.


```

<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- Edited with XML Spy v4.2 -->
<FH_NEWS>
  <news>
    <titel> titel 1</titel>
    <date>datum</date>
    <description>nachrichtentext 1</description>
  </news>
  <news>
    <titel> titel 2</titel>
    <date> datum </date>
    <description> nachrichtentext 2</description>
  </news>
</FH_NEWS>

```

3 → Hier kann die SQL-Anweisung die sich im <ORACLE> Tag befindet editiert werden. In diesem Beispiel wird keine Datenbankabfrage vorgenommen, da die Daten direkt aus dem XML-Dokument bezogen werden.

4 → Hier wird der XSL Quellcode eingegeben:

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- Edited with XML Spy v4.2 -->
<htmlxsl:version="1.0"xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns="http://www.w3.org/TR/xhtml1/strict">

  <body style="font-family:Arial,Helvetica,sans-serif;font-size:9pt; background-color:white">
    <xsl:for-each select="FH_NEWS/news">
      <div style="background-color:blue;color:white;padding:4px">
        <span style="font-weight:bold;color:white">
          <xsl:value-of select="titel"/></span>
          - <xsl:value-of select="date"/>
        </div>
        <div style="margin-left:10px;margin-bottom:1em;font-size:8pt">
          <xsl:value-of select="description"/>
        </div>
      </xsl:for-each>
    </body>
  </html>

```

Schritte 5 –9 siehe im Kap. 5.4.1.4

„Finish“

Über „**Run**“ kann der eben XML-Ausgabe angeschaut werden.

Damit es als Portlet ins Portal integriert werden kann, muss es nur noch mit dem Portal als Portlet bekannt gemacht werden. Dies geschieht indem man auf den „Access“ Tabreiter klickt und im Feld **Publish to Portal** ein Häkchen setzt.

Im nächsten Schritt wird das *XML*-Portlet auf der FH-Startseite eingefügt, damit es permanent angezeigt werden kann.

Dazu navigiert man auf die *FH_GM_Portal_Startpage* und begibt sich in den „Edit Page“ mode.

Siehe Kap. 5.3.4

Nun klickt man in dem gewünschten Bereich in dem das XML-Portlet erscheinen soll auf „**add Portlet**“ und wählt das eben erstellte Portlet aus der Liste aus. Durch „Close“ gelangt man wieder auf die Portalseite und kann das Ergebnis der Abfrage unverzüglich sehen.

5.5 Ergebnisse der praktischen Umsetzung

Screenshots der erstellten Portalseiten und Inhalte sind im Anhang als Abb. 81, 82 und 83 dargestellt.

Life: Der Zugang erfolgt für alle Benutzer über dieselbe URL [FHPOR].

<http://lwis01.inf.fh-loeln.de:7778>

Nach erfolgreicher Login-Prozedur, wird der Benutzer auf die jeweils vordefinierte Startseite geleitet. Usernamen und Passwörter sind im Kap. 5.2.5.1 aufgeführt.

6 Analyse

In diesem Kapitel wird Oracle Portal (Vers. 3.0.9) bezüglich der in Kapitel 2.3 definierten Eigenschaften theoretisch analysiert und anhand einzelner Beispiele, aus den praktischen Einsatz im Kap. 5 vorgestellt. Dabei fließen eigene Erfahrungen, die sich aus dem praktischen Anwendung ergeben mit ein, um eine kritische Auseinandersetzung mit dem Produkt zu ermöglichen.

6.1 Einfacher Gebrauch

Die web-basierte Oracle Portal Software ist über jeden Standard-Webbrowser einfach zu bedienen. Wobei die jeweils aktuelle Browserversion zu verwenden ist, um Darstellungsfehler zu vermeiden. Die integrierten Assistenten erleichtern dem Benutzer die Realisierung einer Portalinfrastruktur, in welcher dem Anwender Informationen schnell und übersichtlich zur Verfügung gestellt werden. Er kann innerhalb des EIP's einfach über Links zu den einzelnen Content Areas (4.3.1) und den Portalseiten (4.2.3) navigieren in denen er gewünschte Informationen oder Anwendungen zur Verfügung gestellt bekommt.

Die im Oracle Portal verfügbaren Assistenten zur Veröffentlichung von Informationen über Content Areas sind selbsterklärend und somit einfach und zum größten Teil ohne Programmierkenntnisse zu bedienen. Sie begleiten den Anwender Schritt für Schritt bei der Entwicklung eigener Anwendungen. Bei der Erstellung von Item-Typen mit Java oder PL/SQL wird sich so mancher Endanwender überfordert fühlen, deshalb bietet das Oracle Portal auch die Möglichkeit, Item-Typen einer Content Area bei der Erstellung auszuwählen. Weiterhin besteht die Möglichkeit eigene Item-Typen zu definieren, mit den Funktionalitäten von Standard-Item-Typen. Beispielsweise kann ein Item-Typ mit dem Namen „Neuen Bericht hinzufügen“ erstellt werden, der die Funktionalität eines File-Item-Typs besitzt. Somit kann jeder Anwender die Funktionalität des Items interpretieren.

Die Bearbeitung einer Navigationsleiste erweist sich durch den Assistenten als sehr einfach. Die Möglichkeit auf Ordner innerhalb einer Content Area direkt über Links in der Navigationsleiste zu verweisen (4.3.1.4) erwies sich als eine sehr benutzerfreundliche Funktion. Die entsprechenden Zugriffsrechte über die der jeweilige Ordner verfügt, werden dem Link assoziiert und somit in die Navigationsleiste eingebunden. Der über den Link aufgerufene Ordner kann somit nur vom Zugriffsberechtigten eingesehen werden.

Die Bedienung des vorhandenen Stil-Assistenten erweist sich ebenfalls als sehr einfach und übersichtlich. Über eine Pop-Up Liste kann der Anwender das gewünschte Stil-Element aus einer Liste bereits vorhandener Stile auswählen und sich diesen über eine Preview-Funktion ansehen. Bei Bedarf kann er selber diverse Eigenschaften, wie Farbe, Ausrichtung, Schriftart und -größe ändern und dem Stil zuweisen.

Für die Erstellung von einfachen Anwendungen bieten die integrierten Assistenten eine ausreichende Unterstützung. Es bietet aber auch Unterstützung bei der Entwicklung komplexer Anwendungen, die *XML*-, *Java*- oder *PL/SQL*-Kenntnisse erfordern. Für erfahrene Benutzer bietet das Oracle Portal eine weitumfassende Hilfefunktion zu diesen Themen an. Sollte der Anwender dort nicht fündig werden, kann er weitere Beispiele und Tutorien auf der eigens für Oracle Portal geführten Webseite finden [PORST].

Portalbenutzer mit geringen Erfahrungen in Umgang mit *Java*, *SQL* und *XML* werden sich anfangs überfordert fühlen, wenn sie versuchen Anwendungen zu erstellen und selbst ins Portal zu integrieren. Da es verschiedene (oft kleine) Aspekte sein können, die zum Scheitern der Integrationsaufgabe führen. Es kann zum Beispiel vorkommen, dass ein fehlerhaftes Portlet sich noch im Portal-Cache befindet und zu einem unbefriedigenden Ergebnissen führt, obwohl der Entwickler bereits einen fehlerfreien Programmcode ins Portal eingespielt hat.

Das Löschen des Caches über eine Funktion im Portal führt nicht zu dem Löschergebnis wie das manuelle Löschen über einen Dos-Befehl.

Nicht nur der Portal- und Portletentwickler sieht sich überfordert, sondern auch der Endanwender, wenn er versucht seine eigene Startseite zu personalisieren. Beispielsweise beim Arbeiten mit *HTML*-Code. Wählt der Benutzer in Abb. 69 Customize unter Punkt (1) und fügt in diesem Menü seinen *HTML*-Code ein, wird die Änderung nur für ihn sichtbar. Besucher der Seite können nur *HTML*-Inhalte sehen, die über das Portalseitenübergreifende Edit- oder Customize-Menü (2) eingefügt wurden. Der Benutzer, der die Seite erstellt, merkt von diesem Manko leider nichts, da für ihn die Änderung sofort sichtbar ist. Er müsste unter einem anderen Benutzernamen auf seine Seite zugreifen um die Funktionalität zu testen. Was oft nicht möglich ist.

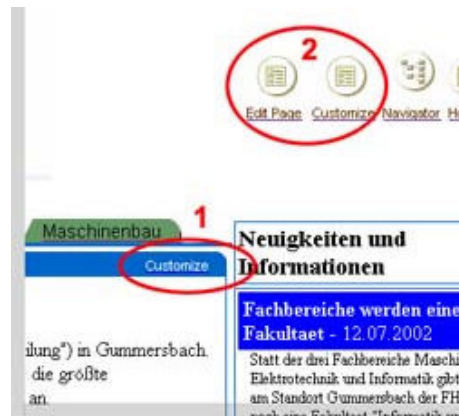


Abbildung 69: Ausschnitt aus der FH-Portal Startseite

Ein weiterer Punkt, den der Benutzer beim Arbeiten mit *HTML* beachten sollte, kommt besonders zum tragen, wenn die Portalumgebung in englischer Sprache sein sollte. In dieser Einstellung wird die Verwendung von, der in deutscher Sprache sehr gängigen, Umlauten nicht unterstützt. Sollten Umlaute innerhalb von *HTML* verwendet werden, ist es zu empfehlen, sie mit dem ASCII-Code darzustellen.

Bsp:

Zeichen	ASCII-Code
ä	ä
ü	ü
ß	ß
...	...

Weitere ASCII-Zeichen sind unter der URL [ASCII] zu entnehmen.

Nichtbeachtung dieser Portaleigenheit führt zu fehlerhaften oder leeren *HTML*-Portlet Darstellung innerhalb des Portals. Bei der Ausführung von Anwendungen, die beispielsweise Namen aus der Datenbank auslesen und diese in *HTML*-Tabellen schreiben, die Umlaute zum Inhalt haben, kann es zu extrem langen Ausführungszeiten kommen.

Diese oben beschriebenen Mängel stellen keine wirkliche Hürde bei der Arbeit mit Oracle Portal dar. Es ist jedoch für den Benutzer wichtig dieses zu wissen, um bei der Erstellung von Portalinhalten nicht unnötig Fehler einzubauen. Genauso wichtig ist für den Neu-User sich mit der Bedienoberfläche vertraut zu machen. Die einzelnen in Abb. 70 umrandeten Icons, verfügen über keine Art von Erklärungstext bei Berührung mit dem Mauseursor, wie es der Benutzer aus diversen Windows Programmen gewohnt sein könnte. Zudem sind

sie nicht immer selbsterklärend, so dass der Benutzer ihre Funktion nicht sofort aus dem Icon ableiten kann. Siehe Abb. 70.



Abbildung 70: Seiten Edit Menü mit den einzelnen Funktionen.

Erklärung zur Abbildung 70:

Unter (1) kann der Anwender die zuvor mit einem Häkchen markierten Tabreiter (2) oder Portlets mit „hide“ für den Seitenbetrachter unsichtbar machen, ohne sie aus dem Portal entfernen zu müssen und bei Bedarf mit „show“ wieder anzeigen lassen. Mit „delete“ können sie von der Seite ganz entfernt werden und mit „move“ in andere Seitenregionen verschoben werden.

Unter (2) kann der Anwender den Namen des Tabs ändern und im Untermenü die Zugriffsrechte auf das jeweilige Tab vergeben. Im Beispiel des FH-Portals stellt sich diese Funktion als sinnvoll heraus, da hier angemeldeten Studenten Zugriff auf Content Areas im der Region eines für den „Public User“ ausgeblendeten Tabs, gewährt werden kann.

Unter (3) kann der Benutzer die Tabs in der Anzeigereihenfolge tauschen, neue Tabs hinzufügen, die Breite der Region bestimmen, Regionen innerhalb der „Oberregion“ waagrecht und senkrecht hinzufügen und mit dem „X“ die gesamte Region löschen.

Unter (4) sind die gleichen Funktionen möglich, da es sich hierbei um eine „Unterregion“ handelt, ist hier mit dem ersten Icon noch die Möglichkeit gegeben, Portlets dieser Region hinzuzufügen.

Das Produkt Oracle Portal entwickelt sich trotz kleiner Kompliziertheiten nach einer gewissen Einarbeitungszeit zu einem einfach bedienbaren Tool, mit dem simple EIP's mitsamt Anwendungen im Unternehmen erstellt und verwaltet werden können. Möchte ein Unternehmen jedoch die gesamte IT-Infrastruktur im Portal abbilden, ist das Hinzuziehen von Consultingteams und Oracleeigenen Portalentwickler unumgänglich. Was natürlich erhebliche Kosten mit sich zieht.

6.2 Universell einsetzbar

Das Oracle Portal bietet in vielerlei Hinsicht eine hervorragende Möglichkeit ein universell einsetzbares EIP zu realisieren. Dazu bedient es sich der Portlets (4.2.1), Portalseiten (4.2.2) und der Content Areas (4.3.1). Unabhängig seiner primären Funktion kann das Unternehmensportal so aufgebaut werden, dass ein Anwender bei einem hohen Informations- und Ressourcenbedarf nie die Übersicht darin verliert. Alle Unternehmensinformationen können stets, unter Verwendung der Tabs, Portlets und der Content Areas, strukturiert zur Verfügung gestellt werden.

Mit Hilfe der Portlets lassen sich die meisten Anwendungen eines Unternehmens ins EIP einbinden und als Web-Applikation darstellen. Aber auch die Verwendung von Ordnern (4.3.1.1) bietet eine gute Möglichkeit Anwendungen auszuführen oder darzustellen. In diesen Ordnern lassen beispielsweise *PL/SQL*- oder *Java* Code integrieren, der auf andere Ressourcen zugreift oder bestimmte Aufgaben erfüllt.

Wie im Kap 4.2.2 erläutert, erstellt Oracle Portal alle Informationen dynamisch. Was zufolge hat, dass jede Anfrage alle Informationen über den Anwender und die Anfrage selbst enthält. Diese Funktion wird für die Erstellung der Antwort benötigt, kann aber auch bei der Integration von Fremdapplikationen verwendet werden. Was den universellen Einsatz unterstreicht.

Auch der *SSO*-Dienst (Single Sign-On) trägt dazu bei, indem es die Integration von Fremdanwendungen dadurch unterstützt, dass sich der Anwender nicht erneut an jeder Anwendung einzeln authentifizieren muss, sondern diese Authentifizierung vom *SSO*-Dienst übernommen wird.

Im Folgenden wird kurz darauf eingegangen für welche Businessbereiche sich das Oracle Portal eignet. Dabei ist zu beachten, dass es nicht ein „maßgeschneidertes“ Produkt ersetzen kann, sondern lediglich unterstützen oder als universelle Lösung betrachtet werden kann.

Content Management (CM)

Oracle Portal integriert Tools des Content Managements. Dazu gehören die Content Areas (4.3.1), in denen Informationen strukturiert, veröffentlicht und verwaltet werden können.

Unter Verwendung der Regionen, Ordner und Navigationsleisten kann sehr einfach eine Struktur aufgebaut werden, in der sich Inhalte darstellen und veröffentlichen lassen.

Die Trennung von Inhalt und Layout (als eine Anforderung an ein *CMS*) ist mit Oracle Portal gewährleistet. Durch die Möglichkeit Items als unsichtbar darzustellen, ist es möglich diese für eine spätere Veröffentlichung vorzubereiten.

CMS werden oft im redaktionellen Bereich eingesetzt. Diesen Anforderungen ist das Oracle Portal nur bedingt gewachsen. Denn es bietet keine Standardkomponente, mit der bestimmte Inhaltskombinationen, wie zum Beispiel: Grafik links, Text rechts daneben, erstellt werden kann. Dazu ist die Entwicklung von Custom-Itemtypen, mit zusätzlichem Programmieraufwand nötig. Auch für die Formatierung und Bearbeitung von Texten bietet Oracle Portal keine ausreichenden Funktionen. Es ist nur bedingt möglich die Layouts einer Content Area mit den verschiedenen Textstilen zu definieren. Wie zum Beispiel „Überschrift 1 Arial –12“, „Bild Text – Times New Roman – 10“.

Dazu müssen zusätzliche Layout und Stilkomponenten definiert werden.

Business Intelligence (BI)

Oracle Portal bietet auch für das Business Intelligence unterschiedliche Komponenten für Abfragen und Berichtserstattung. Dazu gehören Formulare (4.3.2.1), Diagramme (4.3.2.3) und Berichte (4.3.2.2), die alle auf Informationen aus der Datenbank zugreifen. Diese personalisierbaren Komponenten werden mit Hilfe von Assistenten erstellt. Oracle Portal ermöglicht es dem Entwickler einen Bericht als ASCII oder Excel-Datei zu erstellen. Wodurch die Auswertung von Informationen eines Data Warehouses oder Data Marts ermöglicht wird.

Data Mining

Im Oracle Portal sind keine direkten Tools für Data Mining vorhanden. Es ist jedoch möglich mit Hilfe der Abfragefunktionen komplexe Anfragen zu erstellen, die den Anforderungen eines Data Warehouses oder Data Marts durchaus gewachsen sind. Dazu eignet sich besonders die Diagrammfunktion (4.3.2.3) mit der eine Darstellung von Summen, Durchschnittswerten, Maxima, Minima und der Anzahl ermöglicht wird. Diese Darstellungsarten sind bei der Analyse der Daten besonders hilfreich.

Data Management

Auch für das Data Management sind keine direkten Tools vorhanden. Da das Oracle Portal aber auf der Oracle Datenbank (8i) aufbaut, ist es einfach die in der Datenbank integrierten

Tools zu verwenden. Dazu gehören Tool des Data Managements wie z.Bsp. Aktualisierung, Planung und Administration von Daten.

Standard Office Tools

Oracle Portal bietet keine direkten Office Tools, die bei der täglichen Arbeit im Unternehmen benötigt werden. Dazu gehörten ein Email-Client und ein Terminplaner. Diesem Mangel kann durch die Einbindung bestehender Systeme, wie Lotus Notes, über ein entsprechendes Portlet Abhilfe geleistet werden. [LOTU]

Ebenfalls kann die Integration von Officeprogrammen wie MS-Word oder MS-Powerpoint und unter Verwendung der Oracle Portal Schnittstellen realisiert werden, was jedoch mit erheblichem Programmieraufwand verbunden ist.

Für das im Kapitel 3 und 5 beschriebene Projekt eines Hochschulportals eignet sich Oracle Portal besonders gut. Die Content Areas können mit Zugriffsrechten versehen werden, so dass nur berechtigte Personen auf die Inhalte zugreifen können. Durch den direkten Zugriff auf die Datenbank können alle Inhalte wie Stundenpläne, Prüfungsergebnisse, Telefonnummern etc. direkt innerhalb des Portals verwendet werden. Beispielsweise für statistische Zwecke. Beispiel siehe Kap. 5.4.12. und Abb. 71.

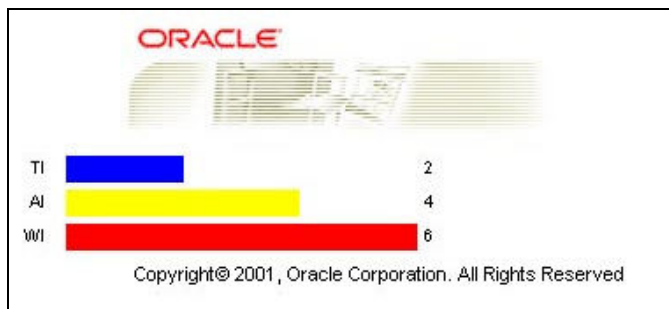


Abbildung 71: Auswertungsportlet mit Anzahl der Studenten

6.3 Suche und Navigation

Das Oracle Portal stellt mehrere Suchdienste zur Verfügung. Das sind im Einzelnen die einfache Suche, erweiterte Suche und Intermedia Search (4.3.1.6).

Die Indizierung erfolgt bei der Erstellung von Items und Ordnern. Der Anwender kann mehrere Stichwörter eingeben, die von der Suchmaschine in der Datenbank indiziert und abgelegt werden. Diese Stichwörter werden bei der einfachen und erweiterten Suche benötigt, um die Items und Ordner schnell zu finden. Die erweiterte Suche bietet weitere Suchfelder, durch die sich mit zusätzlichen Begriffen die Suche spezifizieren lässt. Die Suchergebnisse werden dem Anwender zusammen mit dem Inhalt-Typ und dem entsprechenden Verweis dargestellt. Die Suche wird von den Zugriffsrechten des Benutzers beeinflusst, so dass dem Anwender nur die Suchergebnisse angezeigt werden auf die er auch Zugriffsrechte besitzt. Unter Intermedia Search versteht man eine Oracle eigne Volltextsuche. Dabei werden Dateien nach dem Einspielen in die Datenbank von der Intermedia Searchengine eingelesen und indiziert. Somit kann innerhalb von beispielsweise Word-Dokumenten und *PDF*-Dateien nach bestimmten Wörtern oder ganzen Sätzen gesucht werden.

Durch die Möglichkeit die Suche speichern zu können, wird der Aufwand für oft durchgeführte Suchaufgaben auf nur wenige Klicks reduziert. Oracle Portal bietet Suchmöglichkeiten die sich nur auf die Inhalte der Content Areas beschränken. Informationen, die sich innerhalb von Portlets (Bsp.: *HTML*-Portlet 4.2.1) befinden, werden bei der Suche nicht berücksichtigt. Oracle Portal 3.0.9 bietet keine Möglichkeit die EIP interne Suche auf eine Internetsuche auszuweiten, die Ergebnisse als Links innerhalb des Portals zur Verfügung stellt.

Für den Portaladministrator stehen weitere Suchdienste für Portlets, Datenobjekte, Anwender, Gruppen, Schemata und Provider zur Verfügung.

Zur Vereinfachung der Navigation innerhalb des EIP's stellt Oracle Portal verschiedene Komponenten bereit.

Innerhalb der Portalseiten stehen Tabsets, Menü-Portlets und Navigationsleisten-Portlets zur Verfügung. Informationen innerhalb von Portalseiten können übersichtlich mit den Tabsets (Karteireitern) in einzelne Regionen aufgeteilt und somit strukturiert werden.

Navigationsleisten-Portlets und Menü-Portlets werden zur Navigation auf Content Areas oder als Verweis auf externe Anwendungen verwendet. Eine Navigationsleiste, kann sehr

einfach erstellt werden und mit Menü-Portlets zusammen innerhalb einer Content Area zur übersichtlichen und schnellen Navigation dienen.



Abbildung 72: Navigationsmenü im FH-Portal.

Abb. 72 zeigt das erstellte Navigationsmenü im FH-Portal. Es dient der Navigation auf die Informatik-, Maschinentechnik- oder Elektrotechnik-Homepage. Es ist mit einem „Rollover-Effekt“ versehen, bei dem die Bezeichnung erscheint, sobald der Mauscursor über das entsprechende Symbol fährt.

Die im Kapitel 4.3.2.1 beschriebene Master-Detail Navigationsstruktur sollte ebenfalls mit Ordnern und deren Items möglich sein. Da die Items jedoch nicht in der Navigationsleiste eingebunden werden können, ist diese Funktion nur Beschränkt möglich. Ordner, die externe Anwendungen integrieren, beinhalten oft nur ein Item. Dem Anwender ist es nicht möglich direkt auf die Anwendung zuzugreifen, da er zuerst diesen Ordner öffnen muss.

Für die Navigation innerhalb von Oracle Portal bietet das Produkt mehrere Navigationsportlets und den Portal Navigator. Dadurch werden Portalentwickler bei der Navigation zwischen den Portlets, den verschiedenen Portalobjekten und externen URL's unterstützt. Der Portal Navigator wird über einen direkten Link aufgerufen:



[Navigator](#)

Abbildung 73: Navigator Linkbutton

Er bietet dem Anwender einen direkten und einfachen Zugriff zu allen Portalseiten, Anwendungen, Inhaltsregionen und Datenobjekten auf die der Anwender Zugriffsrechte hat.

Die im Kap. 4.2.3 vorgestellten Navigationsportlets bieten eine zusätzliche Unterstützung bei der Navigation innerhalb des Portals. Es können beispielsweise die zuletzt benutzten Objekte angezeigt werden, Favoriten zusammengestellt oder eine Suche von Portal Objekten gestartet werden.

Oracle Portal ermöglicht den Aufbau eines klassifizierten und übersichtlichen EIP's. Zusätzlich bietet es für die Klassifizierung des Inhalts die Möglichkeit der Kategorien und Perspektiven. Damit ist es möglich, jedem Ordner oder Item eine Kategorie oder Perspektive mit verschiedenen Zugriffsmöglichkeiten zuzuordnen (siehe Kap. 4.3.1.3)

Beim Zugriff auf Ordner und Elemente innerhalb einer Content Area, verliert der Anwender nie den Überblick in welchem Verzeichnis er sich aktuell befindet. Er kann jederzeit zum Ober- und dem Homeverzeichnis springen. Diese Möglichkeit bietet nur die Navigation innerhalb von Content Areas. Sobald der Anwender auf eine externe Seite außerhalb des EIP's zugreift, verliert er die Verknüpfung zu diesem. Die Rückkehr zum EIP, als zentraler Zugangsknoten zu allen Ressourcen, gelingt nur über den „zurück“ Button des Browsers. Was nicht unbedingt der Ideologie der Portalarchitektur entspricht.

Für den praktischen Einsatz stellten sich die Funktionen als hilfreich heraus, denn die Inhalte der einzelnen Portalseiten können nach Stichworten durchsucht und den berechtigten Benutzer zur Verfügung gestellt werden. Allerdings lassen sich die Inhalte, auch wenn sie nur als HTML-Text eingefügt wurden, nicht von externen Suchmaschinen finden. Zum Beispiel durch Google oder Yahoo. Da die Suchmaschinen die Linkauflösung der Portalparameter nicht kennen. Lediglich durch Angabe von Begriffen, die jede Seite kennzeichnen sollen, im „Meta-Tag“ innerhalb des HTML-Codes, kann eine erfolgreiche Indizierung der Seiten für die Suchmaschinen erreicht werden.

6.4 Erweiterbarkeit

Portalentwickler und Anwender können mit dem Oracle Portal ihr EIP problemlos erweitern. Dazu dienen die Content Areas (4.3.1), die Ordner (4.3.1.1) und die Items (4.3.1.2). Mit Hilfe von Assistenten können die Anwender Informationen und Ressourcen in einer strukturierten Form durch Ordner und Items veröffentlichen. Persönliche Ordner können problemlos vom Anwender in seiner Startseite eingebunden werden. Ebenfalls erweiterbar ist die Navigationsstruktur (4.3.1.4) einer Content Area. Mit Hilfe eines Assistenten kann ein Link zu einem Ordner in der Navigationsleiste hinzugefügt werden.

Am einfachsten erfolgt die Erweiterung von Portalseiten durch Hinzufügen von Tabreibern innerhalb des Portals. Ein mit einem Tabreiter neu erstellter Bereich kann vom Benutzer mit gewünschten Portlets und Inhalten gefüllt werden.

Die Funktionalitäten des EIP's lassen sich durch hinzufügen von Item-Typen, *PL/SQL*, Application Components und Java zusätzlich erweitern.

Ein mit Oracle Portal erstelltes EIP kann durch Anwendungen, welche die integrierte API-Schnittstelle benutzen, erweitert werden. Hierzu stellt Oracle eine *PL/SQL* und *Java-API* bereit. Um diese *API* nutzen zu können, bietet Oracle zum Portal das Entwicklungstool „Portlet Development Kit (*PDK*)“. Es ist für jeden frei verfügbar und kann in der aktuellsten Version unter [PDK02] heruntergeladen werden. Dieses Tool beinhaltet Werkzeuge, Richtlinien und Beispiele, mit denen Entwickler eigene *Java*-Portlets erstellen können. Näheres dazu ist im Kapitel 4.6 zu finden.

Im Folgenden werden die verschiedenen API-Packages vorgestellt, die von der *PL/SQL-API* zur Verfügung gestellt werden.

Provider API packages:

- wwpro_api_provider_registry package,
- wwpro_api_provider package,
- wwpro_api_parameters package,
- wwpro_api_node_registry package,
- SNP provider synchronization scripts

Sie stellen mehrere Methoden für die Registrierung, Bearbeitung und Erstellung von Providern und Portlets zur Verfügung. Des Weiteren dient sie der Anmeldung von Remotenodes und deren Synchronization.

Security API packages:

- wwsec_api package
- wwsec_app_priv package

Diese dienen der Erstellung und Verwaltung von Anwendern, Gruppen und deren Zugriffsberechtigungen. Weitere Methoden stehen für die Zugriffsberechtigung auf einzelne Anwendungen zur Verfügung.

Session context API package:

wwctx_api package

Hier können Methoden für die Verwaltung einer Anwendersitzung (user session) benutzt werden.

National Language support packages:

wwnls_api package

Sie bieten Zugriff auf das NLS System der Oracle 8i Datenbank. Es beinhaltet Methoden für die Speicherung und Verwaltung von Daten, die für multilinguale Portlets benutzt werden können.

Error handling and display API packages:

wwerr_api_error package
wwerr_api_error_ui package

Sie beinhalten Methoden für das Auffangen, Bearbeiten und Anzeigen von Fehlermeldungen.

Event logging API packages:

wwlog_api package
wwlog_api_admin package

Diese stellen Methoden für das Aufzeichnen und Bearbeiten von Aktionen zur Verfügung.

Preference API packages:

wwpre_api_name package

wwpre_api_value package

Diese Packages beinhalten Methoden für die Verwaltung von personalisierten Daten.

Session storage API package:

wwsto_api_session

Dieses dient zur Speicherung von Sitzungsinformationen. Sie können mehreren Anwendungen zur Verfügung gestellt werden.

User interface portlet API package:

wwui_api_Portlet

Es beinhaltet Methoden für die Integration von Seitenstilen in Portlets. Weitere Methoden stehen für das Erstellen von Rahmen, Kopf- und Fußzeilen und Rahmen von Portlets zur Verfügung.

Die Java-API bietet ähnliche Funktionen an. Diese werden im Kapitel 4.6 erläutert.

Web-Portlets können mit jeder webfähigen Programmiersprache erstellt und in das EIP integriert werden. Hierfür bietet Oracle Portal eine API für *SQL* und *Java*. Diese sind für die Erstellung von neuen Anwendungen ausreichend. Es fehlt allerdings eine API für viele im Unternehmen bereits bestehende *C/C++* Anwendungen. Dadurch stoßen Portalentwickler auf Hindernisse und Grenzen bei der Integration von bestehenden Softwarelandschaften eines Unternehmens ins Portal.

6.5 Kollaborativ

Oracle Portal ermöglicht es den Anwendern Informationen zu verteilen und untereinander zu teilen. In den Content Areas können Gruppenverzeichnisse eingereicht werden. In diesen können berechtigte Benutzer Informationen veröffentlichen und auf diese zugreifen. Es ist damit möglich, verschiedene Dateiformate als Items ohne Datenbankkenntnisse in die Datenbank zu laden. Durch den „Add-Item“-Assistenten wird eine sehr einfache Möglichkeit zur Veröffentlichung von Informationen und Anwendungen geboten.

Unter Verwendung von Content Areas kann auf einfache Art und Weise ein Informationsportal (KM-Portal) aufgebaut werden. Dazu können mehrere Content Regionen definiert werden, die unterschiedliche Themen beinhalten. Innerhalb dieser Regionen werden Ordner, die verschiedene Unterordner enthalten, erstellt. Der Inhalt wird mit Hilfe von Items veröffentlicht. Die Navigationsleiste kann aus verschiedenen Links, Regionen und einem Suchdienst aufgebaut werden. Ein solches KM-Portal kann mit den „out-of-the-box“-Funktionalitäten von Oracle Portal einfach und schnell erstellt werden. Es sollte auch Dienste und Tools für Verteilerlisten und Diskussionsforen bieten, die der Anwender aktiv nutzen kann. Zu diesem Punkt stellt Oracle Portal in der getesteten Version keine Komponenten bereit. Eine Anmeldung in einer Verteilerliste kann mit der Formular-Komponente durchgeführt werden, doch weitere Funktionalitäten müssen zusätzlich programmiert werden. Es ist möglich mit Oracle Portal Informationen und Anwendungen durch Portlets zu verteilen. Des Weiteren kann der Portalentwickler einen „News-Dienst“ realisieren, der auf Daten aus der Datenbank-Tabelle zugreift. Dazu wird ein Bericht Portlet (4.2.2.2) verwendet, dieses greift auf die Datenbank zu, wobei die Datenpflege über ein Formular-Portlet (4.2.2.1) erfolgt.

Eine gemeinsame Bearbeitung von Aufgaben (z.Bsp. innerhalb eines Teams, bei einem umfangreichen Projekt) ist jederzeit möglich. So können mehrere Portalentwickler gemeinsam verschiedene Aufgaben durchführen. Die Ergebnisse können sofort betrachtet und ausgewertet werden. Durch diese Möglichkeit kann der Projektstatus genauer eingeschätzt werden, um den Projektzeitraum besser bestimmen zu können.

Um inkonsistente Zustände bei der Bearbeitung von Komponenten zu vermeiden, sollte eine Funktion für den Entwickler vorhanden sein, die das gleichzeitige Bearbeiten derselben Komponente durch andere Anwender verhindert. Oracle Portal bietet in der getesteten Version hierfür keine Unterstützung an.

6.6 Administrierbar

Zur Administrationszwecken des EIP's stellt Oracle Portal übersichtliche und benutzerfreundlichen Tools und Assistenten zur Verfügung. Die Administration erfolgt ebenfalls über Webbrowser und ermöglicht somit eine von Aufenthaltsort des Portaladministrators unabhängige Administration der Dienste. Die Möglichkeit der Aufteilung von Administrationsaufgaben gehört zu einer weiteren Stärke des Oracle Portals. Es können mehrere Benutzer mit verschiedenen Administrationsrechten und Privilegien eingereicht werden. Somit können nur bestimmte Teile des EIP's administriert werden, ohne dass der Hauptadministrator die gesamten Rechte an einen anderen vergeben muss.

Oracle Portal ermöglicht die Administration sämtlicher, für das EIP notwendiger, Dienste. Die Anwenderverwaltung beim Einsatz von *LDAP*²²-Verzeichnissen weist allerdings einige Schwächen auf (siehe Kap. 6.7).

Portaladministratoren können problemlos folgende Portalkomponenten von Oracle Portal verwalten:

- Anwender (4.5.1)
- Gruppen (4.5.2)
- Datenbank (4.5.3)
- Loginserver (4.5.5)
- Listener (4.5.5)
- Weitere Portaldienste (4.5.5)

Zur erfolgreichen Optimierung des Portals tragen die Monitoring-Dienste (4.5.4) einen zusätzlichen Beitrag bei. Ein Beispiel eines Auswertungsportlets ist in Abbildung 74 zu sehen. Mit ihrer Hilfe kann der Portaladministrator beispielsweise sehen, welche Komponenten wie oft, von welchen Usern aufgerufen wurden und dementsprechend das Cacheverhalten einstellen.

Mit der Möglichkeit eigene Stile (4.1.3.2) und Layouts (4.1.3.1) als Vorlagen für das EIP erstellen zu können, trägt Oracle Portal dazu bei, das „look and feel“ dem Erscheinungsbild des Unternehmens anzupassen.

²² LDAP – Einträge im LDAP-Verzeichnis erlauben die system- und ortsunabhängige Auffindung und Nutzung aller Objekte im Netzwerk. [LDAP1]

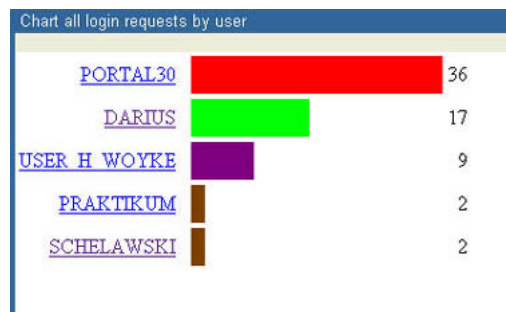


Abbildung 74: Requests auf das FH-Portal im Monat 11-2002.

6.7 Sicherheit

Das umfangreiche Sicherheitskonzept von Oracle Portal ermöglicht eine Berechtigungsverwaltung und Kontrolle auf verschiedenen Ebenen. Portal Anwender sind keine Datenbankanwender, da sie einem eigenen Datenbankschema zugeordnet werden. Diesem Schema können Datenbankrechte und Rollen zugeordnet werden, die auf alle dem Schema zugeordneten Anwendern übertragen werden. Erfolgt ein Zugriff eines Anwenders auf die Datenbank, so wird über das zugehörige Schema die Berechtigung geprüft. Die Zugriffskontrolle auf Datenbankebene ist dadurch gewährleistet. Jedem einzelnen Anwender können portaleigene Privilegien zugeordnet werden um seine Zugriffsberechtigungen auf die Portalobjekte zu gewährleisten. Damit können globale und objektspezifische Privilegien für folgende Objekttypen vergeben werden:

- Portalseiten
- Stile
- Layouts
- Portlets
- Items
- Applikationen
- Anbieter
- Content Areas
- gemeinsam benutzte Komponenten
- Anwender
- Gruppen
- Schemata

Die globalen Privilegien werden einem Anwender oder einer Gruppe über Anwender- bzw. Gruppenverwaltung zugeordnet. Diese Privilegien werden objektspezifisch vergeben und verwaltet. Das hat zum Vorteil, dass Privilegien auf unterschiedlichen Ebenen zugeordnet werden können.

Beim praktischen Einsatz stellte sich jedoch diese Möglichkeit als Nachteil heraus, weil bei Änderungen von Zugriffsrechten, diese an verschiedenen Stellen (Portalseiten, Tabreibern, etc.) vorgenommen werden müssen.

Aus Kap. 4.4.2 geht hervor, dass Oracle Portal zwischen Lightweight User Authentication und External Repository Authentication unterscheidet. Dadurch wird die Verwendung von im Unternehmen bereits bestehenden User Accounts ermöglicht.

Bei der Verwendung von „External Repositories“ (LDAP-Verzeichnis) wären in diesem Zusammenhang einige Schwachpunkte aufzuführen:

- Meldet sich ein Anwender zum ersten Mal am Portal an, erfolgt ein dynamischer Eintrag in das Oracle Portal Schema, mit allen dazugehörenden Userinformationen. Erst nach diesem Eintrage kann der Anwender als Portal Anwender innerhalb des Portals agieren.
- Es ist nicht möglich mit den Diensten von Oracle Portal einen neuen Anwender innerhalb des LDAP-Verzeichnisses anzulegen.
- Beim Löschen eines Anwenders wird sein Eintrag lediglich aus dem Oracle Portal gelöscht, nicht jedoch aus dem LDAP-Verzeichnis. Somit ist es dem gelöschten Anwender möglich sich erneut am Portal anzumelden. Da er (wie im ersten Punkt erläutert) dynamisch neu angelegt wird.

Als eine Bereicherung der Portalfunctionalitäten zählt der Single Sign-On Dienst. Er nimmt den Anwendern, die als lästig empfunden tägliche Anmeldung an den verschiedenen Anwendungen und Diensten. (Email, SAP, Peoplesoft, Yahoo!, etc.). Dieser SSO-Dienst ist nur für Webanwendungen einsetzbar, bei denen das Einloggen über eine HTML-Form erfolgt und keine zusätzlichen Sitzungsinformationen benötigt werden.

Zur Authentifizierung der Benutzer verwendet Oracle Portal den Oracle Login Server. Dieser ist in der Lage zwischen Partner- und externen-Anwendung zu unterscheiden und für diese den SSO-Dienst zu verwenden. Das erleichtert für den Portalentwickler die

Integration von Anwendungen ins EIP. Ist die Authentifizierung des Benutzers am Login Server erfolgt, erzeugt Oracle Portal eine Anwendersitzung. Dazu verwendet Oracle Portal für das Session Management zuständige Komponenten (Apache mod_PL/SQL Modul, Parallel Engine, Portal Provider System und Session Context Subsystem (WWCTX)). Diese Komponenten verwenden zur Speicherung und Verwaltung von Sitzungsinformationen öffentliche und private Packages. Für das Session Management werden Cookies und *URL*-Parameter verwendet. Diese Session Informationen werden verschlüsselt in der Datenbank gespeichert. Zu den Session Informationen gehört:

- Session Identifier
- IP-Adresse des Browsers
- Node-ID des Portal Knotens, der die letzte Sitzung aktualisiert hat
- einen Zeitstempel

Das von Oracle Portal verwendete Session Management entspricht den üblichen Technologiestandards und weist (soweit bekannt) keine Schwachstellen auf. Es bietet die Möglichkeit bei Verwendung einer verteilten Portal Architektur ein einheitliches Session Management zu gewährleisten.

Sollte eine zusätzliche Sicherheitsstufe bei der Verwendung des EIP's nötig sein, bietet der Oracle Login Server und das Oracle Portal hierfür eine Lösung in Form eines verschlüsselten Datenaustausches an. Dazu müssen der Login Server und das Oracle Portal für den „*HTTPS*-Modus“ konfiguriert werden.

Da Oracle Portal allerdings den *HTTP*-Modus für die Verbindung zum Login-Server verwendet, kann kein „reiner“ *HTTPS*-Modus realisiert werden.

In diesem Kapitel wurden die wichtigsten Sicherheitsaspekte vom Oracle Portal erläutert. Es beinhaltet jedoch viele weitere Funktionen und Möglichkeiten, deren Erläuterung den Rahmen dieser dreimonatigen Arbeit sprengen würde. Näheres dazu ist unter [PAEN00] zu finden

6.8 Skalierbar

Die 3-Schichten (three-tier) Architektur macht eine Skalierbarkeit möglich, die den heutigen Anforderung an ein Portalsystem in jeglicher Hinsicht gewachsen ist. Damit ist besonders die Anzahl der Anfragen die gleichzeitig bearbeitet werden können und die daraus resultierende Geschwindigkeit zur Bearbeitung der Prozesse gemeint.

Die Mittelschicht des Oracle Portal besteht aus folgenden Komponenten:

- Apache Web-Server
- Parallel Page Engine
- mod_PL/SQL
- Portal Cache
- Apache Jserv

Es ist möglich mehrere Apache-*HTTP*-Server und mehrere Parallel Engines gleichzeitig einzusetzen. Diese Möglichkeit dient dem Einsatz von Load-Balancing Systemen, welche für zusätzliche Performance sorgen können.

In Abbildung 75 wird diese Möglichkeit verdeutlicht, wobei die Anfragenverteilung (Load-Balancing) hardwaremäßig oder durch einen *DNS*-Dienst erfolgen kann. [PAMI00]

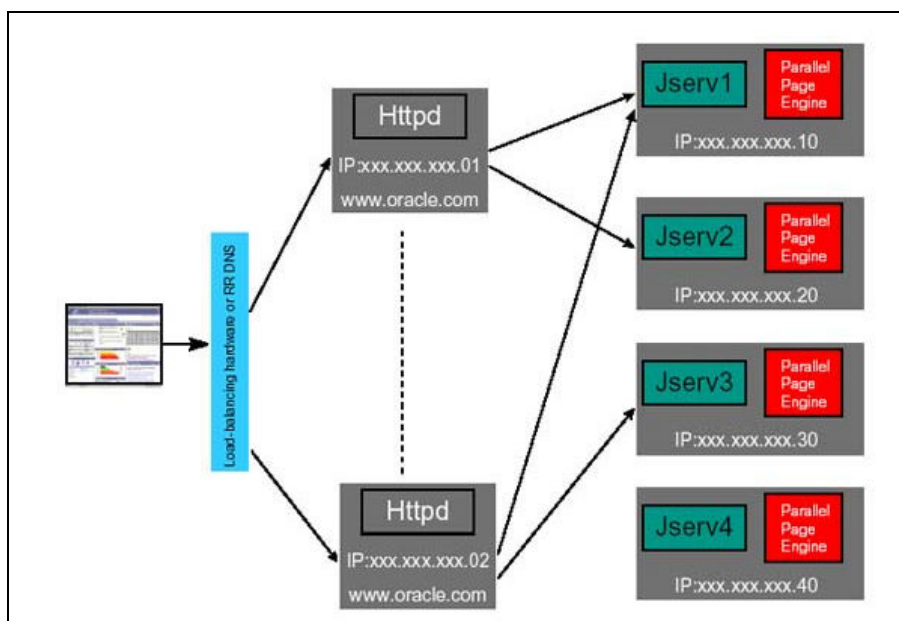


Abbildung 75: Konfigurationsmöglichkeiten des Apache Servers²³

²³ Die Java Virtual Machine des Apache Servers ist hier in der Parallel Engine integriert

Jede Instanz eines Apache *HTTP*-Servers (Httpd) besitzt eine eigene TCP/IP-Adresse, wodurch mehrere Instanzen auf einen einzigen oder auf mehreren Rechnern verteilt werden können. Jeder Apache HTTP-Server verfügt über einen `mod_PL/SQL` Dienst und einen Listener.

Der Listener dient dem Empfang der Anfragen, die er an die `mod_PL/SQL` weiterleitet. Dieser ruft Oracle Portal Prozeduren auf, von denen er Meta-Daten für die angefragte Seite erhält. Diese Meta-Daten sendet er nun an die Parallel Engine weiter. Diese Parallel Engine ist eine Schlüsselkomponente in der Oracle Portal Architektur. Sie läuft unter dem Apache Jserv und erhält die Meta-Daten einer `mod_PL/SQL`-Anfrage des *HTTP*-Servers. Sie ruft die entsprechenden Provider auf, baut die Ergebnisse in einer Seite zusammen und leitet diese an den Browser weiter. Jede Apache Jserv Instanz ist, wie der HTTP-Server, eindeutig identifizierbar und kann ebenfalls auf mehreren Rechnern verteilt werden.

Der Einsatz von mehreren Oracle Portal Instanzen sorgt für zusätzliche Skalierbarkeit. Diese Instanzen werden auch Knoten (node) bezeichnet, die eine eigene Datenbank besitzen und einzeln installiert werden können. Jede Portal Instanz kann als alleinstehende Anwendung, aber auch als Datenbank-Provider für andere Instanzen zur Verfügung stehen. Dazu bedienen sie sich eines Kommunikationskanals durch den sie untereinander bekannt sind.

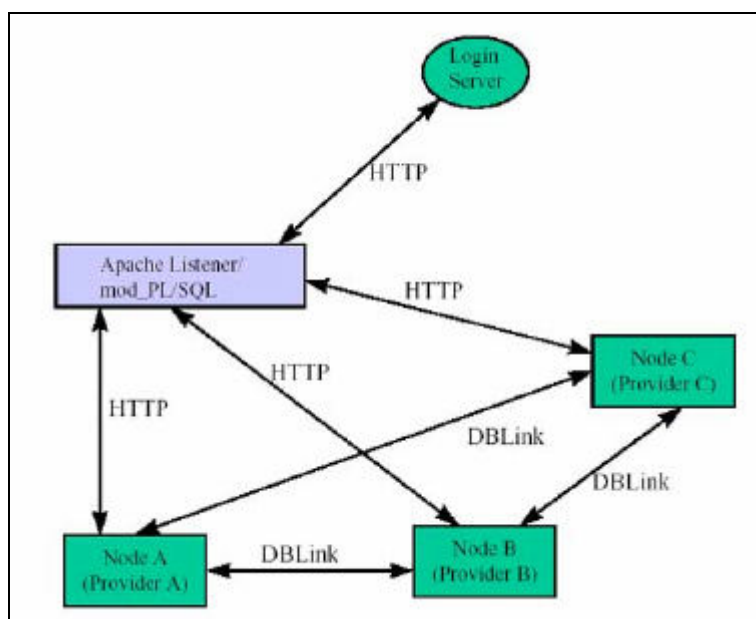


Abbildung 76: : Verteilte Oracle Portal Architektur

Der Apache HTTP-Server kann die unterschiedlichen Instanzen problemlos gemeinsam nutzen, wobei die unterschiedlichen Knoten als Remote-Provider für Portleterstellung von der Parallel Engine verwendet werden können.

Eine parallele Verarbeitung von Anfragen wird durch eine Erweiterung dieser Architektur ermöglicht oder durch die Verwendung mehrerer Web-Provider. Dabei stellen mehrere Web-Server verschiedene Web-Portlets zur Verfügung. Diese Methode führt zu einer verkürzten Antwortzeit des Systems auf konkurrierende Anfragen, was eine beachtliche Performancesteigerung mit sich bringt.

Erhöhung der Performance lässt sich auch durch den effektiven Einsatz der Caching-Dienste des Oracle Portal herbeiführen. Dabei stehen vier Caching-Arten für Portalseiten und Portlets zur Verfügung:

Validation-Based- und Time-Based-Caching (siehe Kap. 4.4.1 Architektur)

Sowie die Unterscheidung von User- und Systemlevel-Cache.

Beim Systemlevel-Cache verwendet eine Gruppe von Portalbenutzern einen gemeinsamen Cache. Sobald ein Anwender aber eine Änderung des Inhalts vornimmt, z.Bsp. indem er ein Portlet seinen Anforderungen anpasst, wird für diesen Anwender ein Userlevel-Cache angelegt. Die restlichen Portalanwender werden weiterhin mit Portlets aus dem Systemlevel-Cache versorgt.

Auf weitere Funktionalitäten des Oracle Caches wird in dieser Arbeit nicht eingegangen. Näheres ist für den interessierten Leser unter [PAEN00] zu erfahren.

Oracle Portal verwendet in der getesteten Version die Oracle 8i Datenbank und kann somit auch deren Dienste benutzen. Dazu gehört u.A. der Parallel Server-Dienst [OPSGS], der die Verwendungen mehrerer Datenbankinstanzen zur Zentralen Datenhaltung unterstützt. Dabei wird eine Datenbank von mehreren Instanzen, mit gleichen Lese- und Schreibrechten, verwendet. Was eine Erhöhung des Transaktionsvolumens und somit Performancesteigerung zur Folge hat.

Die Datenbank kann bei der Verwendung einer einzigen Oracle Portal Instanz unter Umständen zum Engpass werden. Da Oracle Portal selbst, in Form von PL/SQL Stored Procedures, in der Datenbank ausgeführt wird, kann es bei der Verwendung mehrerer Datenbankprovider (deren Portlets ebenfalls in der Datenbank ausgeführt werden) zu besagten Engpässen und somit zu Geschwindigkeits-Einbussen bei der Ausführung kommen. [PAEN00]

6.9 Verfügbar

Die in Kap. 6.8 erläuterten Architekturen des Oracle Portal ermöglichen und stellen einen 24/7 [24 Stunden, 7 Tage die Woche] Einsatz, eins mit Oracle Portal erstellten EIP's, sicher. Dabei lässt sich die Mittelschicht und die Datenbankschicht einzeln betrachten.

Sollte in der Mittelschicht einer der mehreren Apache-HTTP-Server oder Apache Jserv Instanzen ausfallen (Vergleiche Abb. 75), wird der weitere Betrieb durch eine Umverteilung der Lasten auf die verbleibenden Instanzen gesichert.

Auf der Datenbankebene wird der Oracle Parallel Server Dienst [OPSGS] zur Wahrung der Verfügbarkeit eingesetzt. Dieser Dienst sorgt nicht nur für eine Performancesteigerung (siehe Kap. 6.8), sondern auch für eine Umverteilung der Anfragen, beim Ausfall einer Datenbankinstanz, auf die verbleibenden Instanzen.

Zur Sicherung der physikalischen Daten werden die Datenträger (Festplatten) in der Regel gespiegelt (Schreiben der Daten auf mehreren Datenträgern gleichzeitig). Somit wird gewährleistet, dass beim Ausfall eines Datenträgers die erforderlichen Daten von einer der Spiegelungen geladen werden können.

In praktischen Einsatz stellte sich das Portal nicht als 24/7-verfügbar heraus. In der hier verwendeten Version hat der Apache Server sporadisch seine Dienste verweigert. Was dazu führte, dass in diesem Fall Zugriffe auf das FH-Portal von außerhalb des Serverraumes überhaupt nicht möglich waren. (Fehlermeldung im Browser: *The requested URL could not be retrieved*) Es musste jedes Mal der Administrator des Oracle Application Servers bemüht werden, um den Apache-Dienst neu zu starten.

6.10 Personalisierbar

Dem Anwender bietet Oracle Portal mehrere Möglichkeiten zur Personalisierung und Anpassung. Nach erfolgreicher Login-Prozedur gelangt der Benutzer auf seine (dynamisch erstellte) persönliche Startseite, die er selbstständig nach seinen Anforderungen anpassen kann. Dazu gehört das Hinzufügen oder Entfernen von Portlets, ändern von vordefinierten Layouts und Stilen oder bei entsprechender Berechtigung, Erstellen eigener Layouts und Stile.

Oracle Portal bietet auch die Möglichkeit der Portletpersonalisierung. Dazu muss das Portlet eine Personalisierung unterstützen und der Benutzer über entsprechende Rechte

verfügen. Von den meisten Oracle Portal Komponenten wird eine Personalisierung ermöglicht, damit der Benutzer die Informationen individuell aufbereiten kann. So kann der Anwender innerhalb einer Content Area einen persönlichen Ordner (4.3.1.1) erstellen, den er als einen virtuellen Datenträger betrachten kann. In dem kann er seine Anwendungen und Informationen ablegen. Auf die dort abgelegten Inhalte haben nur er selbst und der Portaladministrator Zugriffsrechte, sofern diese vom Anwender selbst nicht an andere Benutzer mit vergeben wurden. Diesen Ordner kann er auch als Portlet auf seiner Startseite einbinden.

Durch das Berechtigungskonzept von Oracle Portal ist eine gezielte Zuordnung von Informationen und Ressourcen möglich, was eine unnötige Informationsflut für den Anwender verhindert. So ist es möglich verschiedenen Anwendern oder ganzen Anwendergruppen bestimmte Informationen zugänglich zu machen. Innerhalb einer Content Area ist es möglich einzelne Ordner und Items bestimmten Anwendern sichtbar/zugänglich zu machen. Das gleiche Prinzip ist auch mit der Navigationsleiste möglich, auf der ein Anwender nur die Links angezeigt bekommt, zu denen auch über Zugriffsrechte verfügt.

Die getestete Version (3.0.9) von Oracle Portal unterstützt die dynamische Sprachanpassung (Kap. 4.2.2.5). Damit ist gemeint, dass der Portalbenutzer die gewünschte Sprache auswählen kann oder das EIP verwendet automatisch die voreingestellte Sprache des Webbrowsers. Die Sprachauswahl bestimmt, in welcher Sprache die Oracle Portal spezifischen Textattribute und Formate für den Anwender angezeigt werden. Das Oracle Portal verfügt über ein „Sprachauswahl Portlet“ (Language Portlet), über das die jeweilige Sprache jederzeit geändert werden kann. Die verfügbaren Sprachepakete müssen allerdings zuvor von dem Portaladministrator über eine Prozedur in die Datenbank eingespielt worden sein.



Abbildung 77: Sprachauswahl Portlet

Unter einer dynamischen Sprachanpassung eines EIP's versteht man allerdings mehr als nur das Anzeigen verschiedener Textbausteine in unterschiedlichen Sprachen. Es sollte eine Möglichkeit für Anwender und Entwickler vorhanden sein, die Attribute eines Ordners oder Items, wie Namen oder Beschreibungen direkt in verschiedenen Sprachen zur Verfügung zu stellen, damit sie ebenfalls den *NLS*-Dienst [National-Language-Support] benutzen. In der getesteten Version ist dies nur über zusätzlichen Programmieraufwand möglich.

Ein weiterer Mangel, der beim praktischen Einsatz aufgekommen ist, sind die Inhalte der *HTML*-Portlets. Ist das Portal beispielsweise auf *NSL*: Englisch gestellt und es werden *HTML*-Inhalte in die dafür vorgesehenen Portlets eingefügt, erscheinen diese *HTML*-Portlets jedoch inhaltslos, wenn die Sprache auf Deutsch gestellt wird.

Das ist zwar unterstützend bei der mehrsprachigen Portalentwicklung, aber störend, wenn lediglich die Benutzeroberfläche des Portals die gewählte Sprache annehmen soll, nicht aber die Inhalte. Eine Auswahlmöglichkeit, ob bestimmte *HTML*-Portlets in allen Sprachversionen mit gleichem Inhalt angezeigt werden sollen, wäre hier wünschenswert.

6.11 Oracle Portal und die Portlet-Standards

Das im Rahmen dieser Diplomarbeit vorgestellte Produkt Oracle Portal 3.0 und das neue Release Oracle 9iAS Portal sollen lt. Herrn Michael Freedman (Mitglied des Oracle 9iAS Portal Entwicklungsteams und des *OASIS*-Konsortiums) in Zukunft auf die im Kapitel 2.9 vorgestellten Standards ausgerichtet sein.

Im folgenden Kapitel werden diese Standards noch mal aus der Sicht des Unternehmens Oracle betrachtet und dargestellt, wie sie zum aktuellen Zeitpunkt und in Zukunft mit dem Produkt 9iAS Portal verbunden werden können. Es wird ebenfalls aufgezeigt, wie *WSRP* und *JSR168* miteinander in Verbindung stehen. Diese Informationen sind aus einem Interview mit Herrn Michael Freedman entnommen. [OTN02 - Suchbegriff: "Freedman"]

6.11.1 WSRP Standard von OASIS

Als ein weltweit agierendes Konsortium das die Entwicklung, Zusammenführung und Adoption von e-Business-Standards lenkt, ist *OASIS* mit mehr als 400 beteiligten

Unternehmen, als eines der einflussreichsten Organisationen für die Verabschiedung neuer Standards bekannt.

Das *WSRP*-Komitee ist aus Mitgliedern von *OASIS* gegründet worden und dessen Initiative wird von allen großen Portal-Software-Anbietern unterstützt.

Ziele von *WSRP* sind es, ein *XML* und Web Services Standard zu definieren, mit dem es möglich sein wird, durch einfaches „Plug and Play“ interaktive Web Services mit standardkonformen Portalen oder dazwischenliegenden Webanwendungen zu verknüpfen.

Als weitere Ziele sieht *WSRP* folgendes vor:

- Ermöglichung der Erstellung und Veröffentlichung von Inhalten und Anwendungen durch benutzerorientierte Web Services
- Ermöglichen von Suchvorgängen nach *WSRP*-Diensten ohne Programmieraufwand
- Ermöglichen der Portletveröffentlichung durch ein Portal, sodass ein anderes Portal diesen Dienst einfach und ohne Programmieraufwand anwenden kann.
- Das Internet als einen Marktplatz für visuelle Web Services zunutze machen, die sofort zur Integration ins Portal zur Verfügung stehen.

6.11.2 Verwandtschaft von *WSIA* mit *WSRP*

Vereinfacht lässt sich sagen, dass es sich bei *WSIA* (Web Services for Interactive Applications) und *WSRP* (Web Services for Remote Portals) um Standards für visuelle, dem Benutzer gegenüberstehende Web Services Komponenten handelt.

WSIA wurde allerdings vor *WSRP* von *OASIS* eingeführt und deren Fokus liegt auf „Plug and Play“ Diensten, die es den Entwicklern ermöglichen herstellerunabhängige Vermittler-Webapplikationen zu erstellen.

Bei Portalen handelt es sich um diese Art von Applikationen. Denn ein Portal ist eine Vermittler-Webanwendung, die zwischen dem Client (*Webbrowser*) und dem Inhalt (Portlets) zum tragen kommt.

Die Beziehung zwischen *WSIA* und *WSRP* liegt darin, dass *WSIA* beabsichtigt die generischen Interfaces zwischen beliebigen Arten von Web Applikationen zu definieren, um sich unabhängige Content-Erzeuger zunutze zu machen, die entfernt von der Applikation über den Web Service Speicher funktionsfähig sind. *WSRP* hingegen beabsichtigt spezifische Interfaces zu definieren, für Web Applikationen innerhalb des Portals.

Da sich diese beiden Standardbemühungen in vielen Bereichen überlappen, wird auch der Versuch unternommen, ein gemeinsames Interface zu definieren mit der Absicht, dass *WSIA*-Komponenten in Portalen verwendet werden können und umgekehrt, *WSRP*-Komponenten in portalfremden *WSIA* Anwendungen genutzt werden.

6.11.3 JSR168 eine neue Java Portlet Spezifikation

Die *JCP* (Java Community Prozess) ist eine offene Organisation für internationale Java Entwickler und Lizenznehmer deren Bestreben es ist Technologie-Spezifikationen, referenz Implementationen und Kompatibilitäts-Werkzeuge für Java zu entwickeln und zu verbessern.

Bei *JSRs* (Java Specification Requests) handelt es sich um tatsächliche Beschreibungen von vorgeschlagenen und entschiedenen Spezifikationen für die Java Plattform.

JSR168 ist eine Portletspezifikation die eine Interoperabilität zwischen dem Portal und den Portlets erlauben wird. Diese Spezifikation wird eine Serie von *APIs* definieren, die in Portalen verwendet die Adressierung der Aggregations-, Präsentations-, Personalisierungs- und Sicherheitsebene übernimmt.

Die aktuellen Ziele der *JSR168* sind:

- eine gemeinsame Portal Metapher definieren
- eine Standard-Portlet-Java *API* definieren
- Interoperabilität und Portabilität garantieren
- Unterstützung verschiedener Bezeichnungssprachen ermöglichen
- Unterstützung bestehender Java *APIs* gewährleisten
- Kompatibilität mit anderen Technologien garantieren

6.11.4 WSRP und Oracle 9iAS Portal

WSRP sorgt für Interoperabilität zwischen den verschiedenen Portal Anbietern und Remote Portlets, die in einer für Web Services ausgelegten Umgebung interagieren sollen. D.h. *WSRP* definiert den Web Services Standard für eine Kommunikation zwischen Portalen und Remote Portlets.

Die Oracle 9iAS Portal Web Provider Infrastruktur baut auf der Web Services Technologie auf. Da es aber bisher keinen einheitlichen Standard gibt, stützt sich das Oracle 9iAS Portal auf eigenen proprietären Protokollen.

Das hat zur Folge, dass in der Oracle 9iAS Web Provider Umgebung entwickelte Portlets nicht in 9iAS fremden Umgebungen lauffähig sind.

Ebenso funktionieren Remote Portlets, die für eine Oracle 9iAS Portal fremde Umgebung entwickelt wurden, in Oracle 9iAS nicht.

Ein Standard wie *WSRP* könnte diesem Mangel ein Ende setzen. Er könnte aus der Architektursicht, dort zum Einsatz kommen, wo aktuell der Web Provider das *SOAP*-Interface verwendet. Es definiert die Serie der *SOAP*-Aufrufe vom Portal an die Remote Portlets. Das primäre Ziel bei der Entwicklung dieses gegenwärtigen Web Providers war es, einen effektiven und skalierbaren Gebrauch zu garantieren, dabei war die allgemeine Verwendbarkeit nebensächlich. Sie erfordert vom Entwickler auf der oberen Schicht des *SOAP*-Dienstes eingeklinkt zu werden. Damit wird dem Entwickler ein Interface geboten, das einfach ist und über eine bequem verwendbare Portlet *API* verfügt.

Das bedeutet für die Entwickler, dass sie von anderen *APIs* isoliert sind und somit auch vom *WSRP* nichts sehen werden oder etwas darüber lernen müssen.

Abb. 78 stellt dar an welcher Stelle *WSRP* zum tragen kommt.

6.11.5 JSR168 und Oracle 9iAS Portal

Mit dem JSR168 wird das Portlet Interface definiert, das die Entwickler nutzen müssen wenn sie Portlets erstellen wollen. Wird es in einer entfernten Umgebung genutzt, ist dieses das Standard-Interface dem die Java basierten *WSRP*-Dienste ausgesetzt sind. Es ist äquivalent zu jetzigen Oracle 9iAS Portal Web Provider Java *API*.

Entwickler sollten sich deshalb lieber auf *JSR168* einstellen, da es in naher Zukunft das Standard-Java-Interface für Portlets werden wird. *WSRP* dagegen ist eher etwas für

Portalentwickler, die eine Portletentwicklung für Remote Portlets für verschiedene Laufzeitumgebungen gewährleisten möchten.

Abb. 78 stellt dar an welcher Stelle JSR168 zum tragen kommt.

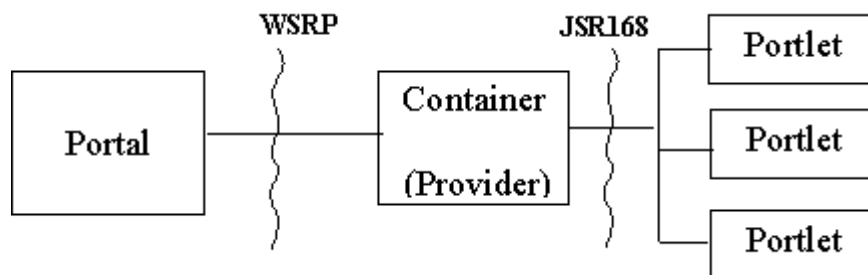


Abbildung 78: Wirkungsbereiches von WSRP und JSR168²⁴

6.11.6 Wann sollen diese Standards verabschiedet werden?

WSRP befindet sich immer noch in einer Vorbereitungsphase, wobei eine offizielle Vorlage bei *OASIS* zur endgültigen Zustimmung für Dezember 2002 vorgesehen ist. Das würde bedeuten, dass *WSRP* frühestens Anfang 2003 ein allgemeiner Standard werden könnte.

WSRP wird in innerhalb eines offenen Prozesses entwickelt. Was zufolge hat, dass eine interessierte Partei, obwohl nicht in der Expertengruppe aktiv, jederzeit den Prozess dieser Spezifikation mitverfolgen und kommentieren kann, wenn es gewissen Prozederen des *OASIS*-Entwicklerkomitees [*OASIS*] befolgt.

Obwohl *JSR168* anderen Richtlinien unterliegt, sieht der offizielle Zeitplan zur endgültigen Verabschiedung des Standards auch Anfang 2003 vor. Die Bestimmungen des *JCP* (Java Community Process) unterliegen einem mehr formalen Prozess, in dem der Standard von einem Expertenteam vorbereitet wird, bevor er der Öffentlichkeit zur Diskussion vorgestellt wird. Sobald ein öffentlicher Entwurf publiziert wird, steht er eine gewisse Zeit für Kritiken und Bemerkungen zur Verfügung. Danach nimmt sich das Expertenteam die Anregungen vor und versucht diese im neuen Entwurf umzusetzen.

²⁴ Vereinfachte Darstellung

6.11.7 WSRP und JSR168 im Oracle 9iAS Portal

Auf die Frage, wie *WSRP* und *JSR168* im Oracle 9iAS Portal unterstützt werden, kommt es zum folgenden Resultat:

Das Oracle Portal Entwicklungsteam verfolgt den Prozess dieser Standards mit der Absicht sie, sobald diese genehmigt sind, auch im Produkt Oracle 9iAS Portal zu unterstützen.

Da sie zurzeit jedoch nicht spruchreif sind, lassen sich nur Schätzungen abgeben, wie sie letztendlich das Produkt beeinflussen werden.

Gegenwärtig kristallisiert sich heraus, dass die angestrebten Standards sehr ähnlich dem Oracle Portal Web-Provider-Framework sind.

Die spezifischen Interfaces sind allerdings verschieden. Sowohl die neuen Standards, als auch die bereits vorhandene Oracle Web Provider Anhäufungen sollen dann unterstützt werden. Diese „Zweigleisigkeit“ solle es dann dem Portletentwickler überlassen, wann er mit der Entwicklung von standarisierten Portlets beginnen möchte.

Durch die ähnlichen Web Provider Funktionalitäten sollte der Übergang zum Standardportlet relativ einfach und problemlos vollzogen werden können.

7 Oracle WebDB und Portal 3.0

Im Jahr 1998 brachte Oracle als einer der ersten Softwareanbieter mit „Oracle WebDB“ ein Produkt auf den Markt, mit dem eine Contentverwaltung mit Datenbankzugriff direkt über den Browser möglich war. Das Produkt entwickelte sich vom Release zu Release immer mehr zu einem Tool, mit dem es möglich war eine portalähnliche Benutzeroberfläche aufzubauen. Zuletzt in der Version WebDB 3.0.

Mit aufkommen des im Kapitel 1 erwähnten EIP-Boom in der IT-Branche, entschied man bei Oracle, eine auf dem Produkt WebDB aufbauende Software zu entwickeln, mit der es möglich sein sollte, den im Kap. 2.2 erwähnten Anforderungen an ein EIP entsprechend, ein Unternehmensportal aufbauen zu können. Die Erweiterung von WebDB wurde in Oracle Portal 3.0 umgetauft.

Im folgenden Kapitel werden kurz die Erneuerungen vorgestellt, die das Produkt Portal 3.0 zu einer Portaldevelopment Software machen. Es werden Komponenten genannt die im Vergleich zur Vorversion hinzugefügt oder verbessert worden sind.

7.1 Verbesserung des Portal User Interface

- Neues attraktives Layout
- Erweiterte Image Unterstützung wie Tabreiter, Banner, Seitenhintergrund
- Komplette neue Navigatorfunktion, mit dem direkte Zugriffe und Änderungen von Datenbankobjekten, Content Areas, Anwendungen und Portalseiten möglich sind. Siehe dazu Kap. 4.2.3

7.2 Erweiterung der Content Bereiche

- Jeder Contentbereich ist jetzt ein individueller Provider
- Das Layout lässt sich flexibler anpassen
- Verschiedene Sicherheitsregeln für einzelne Items sind möglich
- ZIP-Funktion verbessert

7.3 Verbesserung der Portlet Suche und Kategorisierung

- Speicherung der Portlets erfolgt nun in Portal Content Areas. Dadurch ist nun eine Kategorisierung und Suche der Portlets ermöglicht.
- Verbesserte Navigation zwischen den Portlets.

7.4 Neue Tools und Administrationsfunktionen

- Neue Performance- und Benutzer-Überwachungstools (Kap. 4.5.4)
- Benutzersuche mit hierarchischer Darstellungsweise. Die Suche nach einem Portalbenutzer gibt eine Liste aller User aus, auf die das Suchkriterium zutrifft. Weiterhin können Benutzerinformationen direkt oder in einer Organisationshierarchie dargestellt werden. (Manager und seine direkt unterstellten Mitarbeiter)
- Import/Exportfunktion für Content Bereiche und Portalseiten
- Import/Exportfunktion für Anwendungen innerhalb des Portals
- Upload für übergroße Inhalte möglich durch ZIP-Kompression
- Werkzeuge und Skripte zur Erleichterung einer Migration von Komponenten und Webseiten die mit WebDB 2.x und 3.0 Beta erstellt worden sind.

7.5 Erweiterung der Anwendungsentwicklungsfunktion

- Anwendungen besitzen nun eigenen Provider
- Inter Media Suche wird unterstützt. (Siehe Kap. 4.3.1.6)
- Dateien die mit Hilfe von „Oracle Forms“ erstellt wurden, können im Portal verwendet werden.
- Erweiterte Möglichkeiten bei der Erstellung von Diagrammen.

7.6 Partnerprogramm von Oracle

Verschieden Hersteller von Web Services und Portlets beteiligen sich an diesem Partnerprogramm und ermöglichen dem Kunden so eine einfache Integration ihrer Produkte in sein EIP, durch die Einhaltung gewisser Richtlinien bei der Anwendungserstellung. Oracle auf der anderen Seite stellt diese Produkte auf seiner Homepage vor und ermöglicht dem Interessierten Kunden einen direkten Download mit Verlinkung zum Hersteller der entsprechenden Anwendung. [OPAR]

7.7 Änderungen in der Architektur

Die Architektur beim WebDB besteht aus einem eigenen WebDB-Listener und dem WebDB PL/SQL-Gateway über die Zugriffe auf die Datenbank erfolgen.

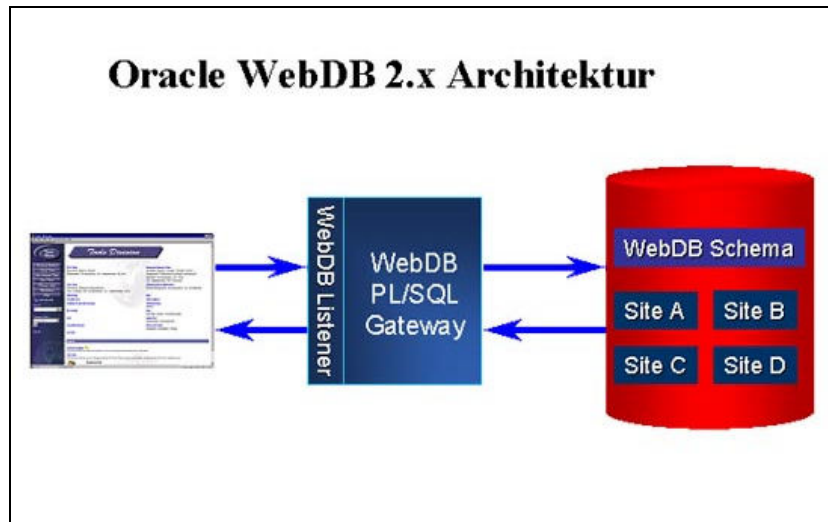


Abbildung 79: Oracle WebDB 2.x Architektur

Änderungen in der Oracle Portal 3.0 Architektur werden in Abb. 80 dargestellt. Es ist zu erkennen, dass eine Erweiterung der folgenden Komponenten erfolgt ist:

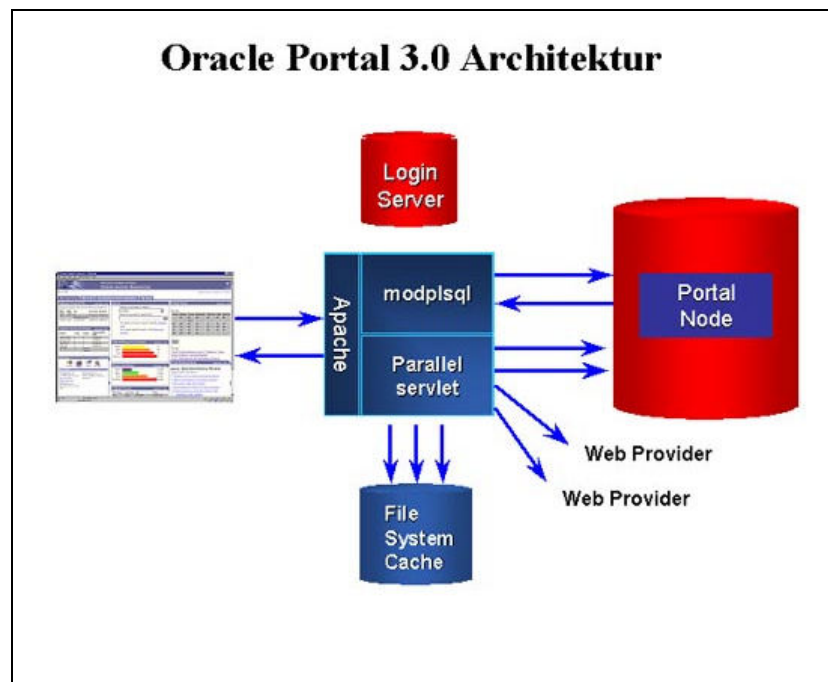


Abbildung 80: Oracle Portal 3.0 Architektur

Server

Alle Funktionen und Inhalte für Oracle Portal 3.0 werden durch ein vereinheitlichtes Schema gemanaged.

Mittel Tier

Es wird kein Oracle eigener WebDB-Listener verwendet, sondern der Apache-Weblistener.

Der WebDB PL/SQL Gateway ist durch „modplsql“ ersetzt worden. Wobei „modplsql“ ähnliche Funktionen vollzieht aber im Apache integriert ist.

Parallel Engine

Die Parallel Engine ist als ein Servlet implementiert und läuft unter der Apache Jserv. Sie fügt den Page Requests eine Serie von Parallel Portlet Requests hinzu.

File System Cache

Der File System Cache wird von der „modplsql“ und der Parallel Engine verwendet um Portlets und Portalseiten zu cachen und somit zur erheblichen Performancesteigerung beizutragen.

7.7.1 Login Server Integration

Mit der Integration des Login Servers in das Oracle Portal 3.0 werden dem Benutzer viele Funktionen geboten.

- Lightweight Authentifizierung
- Datenbank Authentifizierung
- Oracle Applications Authentifizierung
- Externe Authentifizierung am LDAP Server
- Single Sign-On Authentifizierung

7.7.2 Parallel servlet

Durch die „offload“ Portlet Ausführung in der Mittel Tier wird eine verteilte Knoten-Architektur und die Nutzung des Caches ermöglicht. Sie Unterstützt die Verwendung von XML, Web Providern und von Portlets die ihre Informationen an externe Quellen weitersenden können.

7.7.3 Caching

Zwei Arten des Caching werden ermöglicht:

- Validation-Based-Caching
- Expires-Cased-Caching (siehe Kap.4.4.1)

Das Caching kann auf zwei Levels gesteuert werden:

- Portlet caching
- Page caching

8 Zusammenfassung

Ziel dieser Arbeit war es, dem Leser den Begriff und die dahinter stehende Technologie der Enterprise Information Portals (EIP's) näher zu bringen. Anhand von Merkmalen und Fähigkeiten von Oracle Portal und einer kritischen Produktanalyse sollte die Frage beantwortet werden: "Eignet sich Oracle Portal für die Erstellung und zukünftige Verwaltung / Pflege eines Portals für die Fachhochschule Köln, Campus Gummersbach?"

Dazu wurden in den einzelnen Kapiteln dem Leser Informationen gegeben, die bei der Beantwortung der Frage zum Tragen kommen sollten.

EIP's dienen in erster Linie der Integration aller Anwendungen und Informationen eines Unternehmens. Des Weiteren sollen sie auch Anwendungen und Informationen, die das Internet bietet in einer rollenbezogenen, personalisierten Umgebung dem Anwender zur Verfügung stellen können.

Im zweiten Kapitel wurden die hohen Anforderungen an ein EIP's erläutert. Wobei der individuelle Informations- und Ressourcenbedarf eine wichtige Rolle dabei spielt. Der Anwender soll seine benötigten Informationen und Anwendungen nicht an vielen verschiedenen Stellen suchen müssen, sondern sie in einer personalisierbaren Seite über den Browser aufrufen können. Woher diese Ressourcen letztendlich kommen und wie sie für das EIP aufbereitet werden müssen ist nur für Portalentwickler und -administratoren von Bedeutung. Dabei könnten sie sich, wie aus diversen Studien hervorgeht, verschiedener Techniken bedienen. Darunter fallen die Web Services, deren Eigenschaften sich hervorragend für den Portaleinsatz eignen. Dieser Einsatz bedarf jedoch eines noch nicht offiziell verabschiedeten Standards, dessen Funktionsweise und Zusammenspiel ebenfalls in diesem Kapitel angesprochen wurde.

Des Weiteren bieten EIP's Suchdienste und eine einfache Navigationsstruktur an. Sie sind skalierbar und ermöglichen einen sicheren „rund um die Uhr“ Betrieb, wobei sie von jedem Internetfähigem PC aufgerufen werden sollten. Die dafür notwendige Sicherheit, während des Kommunikationsprozesses zwischen User und System aber auch durch die Benutzer-Authentifizierung, muss stets gewährleistet sein.

Der Nutzen den ein EIP dem Unternehmen bringt, liegt in der Erhöhung der Produktivität und Effektivität seiner Mitarbeiter. Diese und weitere Portaleigenschaften gehen aus Auszügen verschiedener, in diesem Kapitel aufgeführter, Studien hervor.

Im Hinblick auf die anfangs gestellte Frage des Einsatzes von Portaltechnologie innerhalb der Fachhochschule, werden im Kapitel 3 einige Verwendungsmöglichkeiten vorgestellt. Dabei wurde die bisherige Webseite der FH analysiert und um neue Einsatzbeispiele theoretisch erweitert. Im Rahmen dieser Arbeit lässt sich der hier beschriebene Portaleinsatz nicht in die Praxis umsetzen. Die Beispiele sollen jedoch auf die Fähigkeiten und Verwendungsmöglichkeiten eines EIP's hinweisen und ggf. eine Inspiration für zukünftige Projekte der Fachhochschule Köln, Campus Gummersbach sein.

Im Kapitel 4 wird das Produkt Oracle Portal vorgestellt. Dieses Produkt wurde für die Entwicklung des praktischen Beispiels eines Informationsportals für die Fachhochschule ausgewählt. Diese Produktvorstellung soll ebenfalls als eine Art Anleitung dienen, die es dem Anwender oder Entwickler des FH-Portals erleichtern soll, sich mit den Funktionen und Möglichkeiten von Oracle Portal vertraut zu machen.

Dabei lässt sich zum Produkt zusammenfassend sagen, dass es sich bei Oracle Portal um eine browser-basierte Softwarelösung handelt, die der Erstellung und Verwaltung von EIP's dient. Sie bietet erweiterbare Frameworks, die auf Portlets basieren. Portlets sind universell verwendbare Komponenten, die vielseitig Informationen und Anwendungen darstellen können. Für ihre Integration in die Portalseiten lässt sich ihr Layout und Design an die Portalseite anpassen. Dabei werden die Portalseiten durch das Seitenlayout, den Seitenbanner, den Seiteneinhalt und die Seitensicherheit definiert.

Ihre Layouts und Stile lassen sich mit Oracle Portal erstellen und anpassen. Die Sicherheit innerhalb der Portalseite wird durch Privilegienzuordnung gewährleistet. Für die Veröffentlichung von Informationen stellt Oracle Portal Content Areas und Components (Komponenten) bereit.

Bei Content Areas handelt es sich um Webseiten, auf denen verschiedene Inhalte über Assistenten veröffentlicht, verwaltet, mit Zugriffsrechten versehen und natürlich angezeigt werden können. Die Elemente, aus denen eine Content Area besteht, sind Ordner, Items und die Navigationsleiste. Weitere Elemente, die sie beinhaltet, sind Suchdienste, Klassifikation und Stil.

Components dagegen sind dynamisch erzeugte Komponenten wie Berichte, Diagramme, Formulare, Kalender, dynamische Seiten und Menüs. Sie können alle mit Hilfe von Assistenten erstellt und als Portlets innerhalb des Oracle Portal dargestellt werden.

Oracle Portal besteht aus 4 Hauptkomponenten, die in einer 3-Schicht-Architektur integriert sind. Es gibt mehrere Dienste, die der Sicherheit dienen. Damit können Portal-Elemente verwaltet und Portal-Anwender authentifiziert werden. Weitere Dienste im Oracle Portal

sind das Single Sign-On, Inhaltsklassifikation, Gruppen, -Anwender, Datenbankverwaltung, Suche und Überwachungsdienste.

Im fünften Kapitel wurde die Umsetzung des im dritten Kapitel vorgestellten Portals für die Fachhochschule Köln, Campus Gummersbach anhand einiger Beispiele realisiert. Dabei wurde ein Konzept für das Design, Benutzer, Benutzergruppen und Inhalte vorgestellt. Anhand dieses Konzeptes wurden drei Portalseiten, deren Layout und Beispiel-Benutzer erstellt. Das hier Aufgebaute Portal soll als Grundgerüst für ein voll einsetzbares Hochschul-Portal sein. Deshalb wurden hier einige Anwendungen und Inhalte integriert, die vorstellen sollen wie das FH-Portal zukünftig eingesetzt werden könnte.

Im Kapitel 6 wurde die Stärken und Schwächen von Oracle Portal, mit Bezug auf den vorangegangenen praktischen Einsatz analysiert. Dabei lässt sich zusammenfassend folgendes festhalten: Oracle Portal ist ein relativ einfach zu bedienendes Portalentwicklungs-Tool, welches die Integration von Informationen und Anwendungen jeglicher Form ermöglicht. Dazu bedient sich der Portalentwickler der Portlet-Technologie. Eine Personalisierung ist mit dem Produkt möglich. Bei der Erstellung von mehrsprachigen EIP's stellt sich Oracle Portal allerdings als bedingt einsetzbar heraus. Die Content Areas ermöglichen eine einfache Veröffentlichung von Informationen und Inhalten. Funktionen wie Email, Verteilerlisten und Diskussionsforen sollten in einem Portalprodukt als Standardfunktionen mitgeliefert werden. Dies ist bei Portal 3.0 nicht gegeben. Die Navigationsmöglichkeiten innerhalb des Portals stellten sich als einfach und übersichtlich heraus. Die integrierten Suchdienste ermöglichen eine Portal-interne Suche, berücksichtigen allerdings nicht alle Portal-Komponenten. Die Skalierbarkeit stellt für EIP's mit enormem Datendurchsatz sehr viele Möglichkeiten der Lastenverteilung, sofern die Oracle 8i oder 9i Datenbank verwendet wird und sorgen somit auch für einen „Rund um die Uhr“ Einsatz. Im Praktischen Beispiel wies der Apache-Server allerdings Mängel auf, die diesen „24/7 – Einsatz“ verhindern.

Zur Erweiterung der Funktionalitäten des EIP's bietet Oracle Portal eine PL/SQL- und eine Java-Schnittstelle. Diese sollten für die Integration weiterer Anwendungen völlig ausreichend sein. Besonders in Bezug auf die Portlet-Standardards, sobald diese von den jeweiligen Gremien offiziell verabschiedet sein werden. Dienste wie Single Sign-On und LDAP tragen zum Sicherheitsaspekt von Oracle Portal bei, bringen aber auch Mängel mit sich. Für die Administration bietet Oracle Portal Tools, mit denen sich die Verwaltung sämtlicher Portaldienste als benutzerfreundlich erwiesen hat.

Abschließend lässt sich feststellen, dass Oracle Portal für den Aufbau und Einsatz als Hochschulportal gut geeignet ist. Durch die einfache Bedienung und relativ kurze Einarbeitungszeit, ist es möglich verschiedene Administratoren mit der Pflege des Portals zu beauftragen. Die einfache Entwicklung von Anwendungen ermöglicht ein rapides Füllen des FH-Portals mit sinnvollen, selbst erstellten Anwendungen und Informationen. Durch die Möglichkeit Rechte auf einzelne Objekte vergeben zu können, ist eine Anwenderabhängige-Content-Area schnell realisiert. Die *Java-API* ermöglicht zudem die Integration portalfremder Anwendungen, was im Beispiel des FH-Portals ein SAP-System sein könnte, welches den Studenten für Praktika zur Verfügung stehen würde.

Ein weiteres Kriterium für den Einsatz eines EIP's ist oft der Anschaffungspreis bzw. Lizenzgebühren. Im Fall einer Realisierung des in dieser Arbeit vorgestellten FH-Portals in der Fachhochschule Köln, Campus Gummersbach, mit der Oracle Software, könnte die Fachhochschule von einem speziellen Programm profitieren.

Durch die OAI (Oracle Academic Initiative) würden sich die Kosten für die Lizenz des 9iAS Portals auf eine jährliche Gebühr von EUR 543,- netto belaufen. Dieser Preis, der noch weitere Leistungen bietet, kann als ein besonders günstiges Angebot angesehen werden. Besonders im Vergleich zu den User- oder Prozessor-abhängigen Lizenzen anderer EIP-Software-Anbieter.

Als Ausblick kann in einer weiteren Arbeit die vollständige Umsetzung des EIP's betrachtet werden. Auch das Thema Web Services, welches im Rahmen dieser Arbeit nur kurz erwähnt wurde, bietet eine Basis. Denn Web Services werden sicherlich in Zukunft die IT-Branche revolutionieren, was in einer weiteren Arbeit vertieft werden könnte. Die Standards für Web Services und Portlets bieten ebenfalls eine hervorragende Grundlage für eine Analyse der einzelnen Portal-Software-Hersteller und die Bemühungen zur Einhaltung dieser Standards in ihren Produkten.

Für das Verwenden, des im Rahmen dieser Diplomarbeit erstellten FH-Portals (siehe Kap.5) wäre die Erweiterung dieses mit Tools und Anwendungen, aus Kap 3 oder die Anbindung von Web Services denkbar. Zum Beispiel könnten PL/SQL- und Java-Anwendungen im Rahmen von Praktika und Projekten als Aufgaben an die Studenten vergeben werden.

9 Anhang

9.1 Screenshots

Fachhochschule Köln
University of Applied Sciences Cologne
Campus Gummersbach

November 12, 2002

FH-Köln Allgemein Studium Informatik Elektrotechnik Maschinenbau

Grundstudium

Das Grundstudium umfasst bei den Präsenzstudiengängen die ersten zwei, beim Verbundstudiengang Wirtschaftsformatik die ersten fünf Semester. Der Studienplan des Grundstudiums enthält vorwiegend Grundlagenfächer, wie Mathematik, Theoretische Informatik, Algorithmen und Vorlesungen zu Betriebswirtschaft und Volkswirtschaft. Entsprechend der Ausrichtung des jeweiligen Studiengangs sind die wirtschaftswissenschaftlichen oder die technisch-physikalischen Inhalte stärker ausgeprägt. Im Vergleich zum Hauptstudium enthält das Grundstudium weniger praktische Elemente (zu Medieninformatik vgl. 4.6).

Querschnittsqualifikationen

Eine intensive Vermittlung von Querschnittsqualifikationen oder auch Schlüsselqualifikationen (engl. soft skills) gehört heute in jedes moderne Ausbildungskonzept einer Hochschule. Die Studierenden sind nicht länger, wie häufig noch in der sekundären Ausbildung, eher rezeptive und passiv angesprochen, sondern sollen die Ausbildung aktiv mitgestalten. In Projekten wird ihre Fähigkeit zu Teamarbeit, in Übungen und Seminaren und schließlich im Diplomkolloquium werden ihre Wortbeiträge in Form von Statements und Referaten gefordert. Wenn auch manche dieser Fähigkeiten auf Begabungen beruhen, können sie doch alle durch theoretische Vermittlung und vor allem durch praktische Übungen hinreichend trainiert und entwickelt werden. Hierzu macht der Fachbereich in allen Studiengängen schon sehr frühzeitig im Studium entsprechende Pflichtangebote, die wir mit Querschnittsqualifikationen I und II (○○ I und ○○ II) bezeichnen.

Studienverlauf

QQ I:
Hierunter fallen Lehrveranstaltungen mit weitgehend seminaristischem Charakter und begrenzter Teilnehmerzahl, häufig in Form von Blockveranstaltungen angeboten, zu folgenden Themen:

Rhetorik, Moderation und Präsentation, Verhandlungsführung, schriftlicher Ausdruck/Dokumentation, Teamarbeit, Kreativitätstraining, Personalführung, Bewerbungstraining, Karriereplanung, technologieorientierte Existenzgründung, Ableitung von wissenschaftlichen Arbeiten, Technikphilosophie, Wissenschaftsetik, Fremdsprachen.

Der Studienumfang umfasst formal 4 SWS, es müssen mindestens 2 Veranstaltungen belegt werden, um den Leistungsnachweis zu erhalten. Eine Ausnahme liegt vor, wenn durch eine besondere Vielfalt in der Wissensvermittlung und praktischen Übung die geforderten 4 SWS über ein volles Semester zustande kommen. Hier reicht 1 Lehrveranstaltung aus, und dies wird durch den entsprechenden Fachknotenplatzwert nachgewiesen.

Neuigkeiten und Informationen

Termin am - 16.10.2002
14:00 Uhr Fests Festkom Weyde

Fachbereiche werden eine Fakultät - 12.07.2002
Statt der drei Fachbereiche Maschinenbau, Elektrotechnik und Informatik gibt es zukünftig am Standort Gummersbach der FH Köln nur noch eine Fakultät "Informatik und Ingenieurwissenschaften". Das reichhaltige Studienangebot bleibt erhalten, es gibt aber ein neues Dekanat und eine neue Struktur für den Campus Gummersbach. Umgesetzt wird das neue Modell mit dem Beginn des Wintersemesters am 1. September 2002.

Gummersbacher Industrieforum - 06.06.2002
Industrielle Kommunikation und verteilte Automatisierung: Aktuelle Trends

Studenten pro Studiengang

TI	2
AI	4
WI	8

Links

- [Hochschulbibliothek](#)
- [Prüfungsamt](#)
- [SLZ](#)
- [Physiklabor](#)
- [Oberberg-Online](#)

Abbildung 81: Fertige FH-Startpage

November 11, 2002

Abgemerkt Projekte Personal **Praxissemester** WPF's

Personal im Informatikbereich

Personenverzeichnis Professoren

Der Fachbereich Informatik verfügt über 25 Planstellen für Professoren. Im Unterschied zu Universitäten wird für die Berufung eines Professors an eine Fachhochschule neben entsprechender fachlicher, wissenschaftlicher und pädagogischer Qualifikation eine mindestens fünf Jahre berufliche Praxis gefordert, davon mindestens drei Jahre außerhalb der Hochschule. Zu den Aufgaben eines Professors gehören neben seiner Lehrverpflichtung von 18 Semesterwochenstunden Forschungstätigkeit und Mitarbeit in der Selbstverwaltung der Hochschule. Die Namen aller Professoren des Fachbereichs sind nachfolgend aufgeführt.

- [Drescher](#), Prof. T.
- [Ehres](#), Prof. Dr. E.
- [Faeskorin-Woyke](#), Prof. Dr. H.
- [Fischer](#), Prof. Dr. K.
- [Gömel](#), Prof. Dr. P.
- [Günther](#), Prof. Dr. H.
- [Holland-Monte](#), Prof. Dr. E.
- [Jochum](#), Prof. Dr. F.
- [Karsch](#), Prof. Dr. rer. nat. S.
- [Klocke](#), Prof. Dr. H.
- [Küttel](#), Prof. Dr. F.
- [Koch](#), Prof. Dr. H.
- [von Landsberg](#), Prof. Dr. G.
- [Lening](#), Prof. Dr. A.
- [Pieper](#), Dr. B.
- [Pflaßmann](#), Prof. Dr. G.
- [Schwanenberger](#), Prof. Dr. P.
- [Somogyi](#), Prof. Dipl.-Kameramann T.
- [Stahl](#), Prof. Dr. H. L.
- [Stenzel](#), Prof. Dr. H.
- [Victor](#), Prof. Dr. F.
- [Westenberger](#), Prof. Dr. H.

Teilnehmer am Einführungs-Seminar

Tag	Datum	Teilnehmer
		Langner, WI D
		Waldorf, WI M
Montag	07.10.02	Bauer, WI A
		Schnitt, WI K
		Lutz, K WI
		Java, O WI

Linksammlung Informatik

- [SAP](#)
- [Oracle](#)
- [SUN](#)
- [IBM](#)
- [Oracle Consulting](#)
- [Kienbaum Consulting](#)

Abbildung 82: Fertige Informatik Homepage

November 11, 2002

zurück **Homepage von Prof. Dr. Heide Faeskorin-Woyke**

Allgemeines **Lehre** Diplom Projekte Allgemeines Logout

TEST TEXT

Woyke_Studenten

Insert Query Reset

Name

Vorname

Studiengang

Geschlecht

Insert Query Reset

XML Teilnehmer_WI

Tag	Datum	Teilnehmer
		Langner, D WI
		Waldorf, M WI
Montag	10.10.02	Bauer, A WI
		Schnitt, K WI
		Lutz, K WI
		Java, O WI

Document: Done (2.553 secs)

Abbildung 83: Fertige Professoren Homepage

9.2 Glossar

Apache: siehe Webserver

API: Anwendungs- und Programmierschnittstelle

Java: Eine von der Firma SUN entwickelte Programmiersprache

LDAP: Dieses Protokoll bietet einen Standard für die Kommunikation mit Datenbanken im Internet. Per LDAP können beispielsweise Email-Verzeichnisse nach Email-Adressen oder Namen durchsucht werden. Das Lightweight Directory Access Protocol ist eine vereinfachte Form des Directory Access Protocol. Es soll die Nutzung von Directory Services für Thin Clients erleichtern. Einträge in ein LDAP-Verzeichnis erlauben system- und ortsunabhängige Auffindung und Nutzung aller Objekte in einem Netzwerk wie: Rechner, Verzeichnisse, Dateien, Dienste oder Personen.

Load-Balancing: Durch das Weiterleiten einer Anforderung an einen beliebigen Server in einer Gruppe identischer Server kann die Arbeitsbelastung unter den Servern aufgeteilt werden. Auf diese Weise wird der Durchsatz clientseitiger entfernter Methodenaufrufe verbessert. Die Anforderungen können gleichmäßig auf die Server verteilt werden. Dadurch wird eine ungleiche Arbeitsbelastung vermieden, bei der ein oder mehrere Server im Leerlauf oder mit geringer Auslastung arbeiten, während andere überlastet sind.

SOAP: Sprach- und plattformunabhängiges Kommunikationsprotokoll für den Datenaustausch zwischen Anwendungen aus dem Hause Microsoft. Der Standard definiert, wie Transaktionen via Internet und XML getätigt sowie dynamische Web Services über verteilte Netzwerke genutzt werden können.

SQL: Sprache zur Abfrage und Manipulation relationaler Datenbanken. SQL wurde in den 70er Jahren von IBM entwickelt und ist seit 1987 ISO-Standard. Die letzte Erweiterung des Standards ist von 1992 und nennt sich SQL2. SQL ist eine mengenorientierte, nicht-prozedurale und keine eigenständige Programmiersprache. Um SQL-Befehle in Datenzugriffsmethoden umzusetzen, ist ein Interpreter nötig. Dafür wurden Schnittstellen wie ODBC, BDE oder JDBC entwickelt.

Thin-Client: Im Gegensatz zur Fat-Client-Variante einer mehrschichtigen Anwendung, entfallen alle vorzunehmenden Einstellungen auf der Client-Seite. Die gesamte Programmlogik befindet sich auf dem Server.

Webbrowser: Programme, die Daten aus dem World Wide Web (WWW) aufrufen und auf einem beliebigen Computer (Client) verarbeiten und anzeigen können. Neben Text beherrschen moderne Browser auch die Anzeige von Grafiken, Videoclips und weiteren Datenformaten.

Webserver: Server, der HTML-Dokumente und andere Internet- / Intranet-Ressourcen speichert und über HTTP versendet bzw. entgegennimmt (auch HTTP-Server genannt).

9.3 Abkürzungsverzeichnis

.NET	Microsoft XML Web Services Plattform
AI	Allgemeine Informatik
API	Application Program(ming) Interface
AS	Application Server
B2C	Business to Customer
BI	Business Intelligence
CMS	Content Management System
CRM	Customer Relationship Management
DAD	Data Access Descriptor
DBA	Database Administration
DNS	Domain Name Server/Service
ERP	Enterprise Resource Planning
FIBU	Finanz-Buchhaltung
HTML	Hyper Text Markup Language
HTTP	Hyper Text Transfer Protocol
HTTPS	Hyper Text Transfer Protocol Secure sockets
IP-Adressen	Internet Protokoll Adresse
ISV	Independent Software Vendor
J2EE	Java 2, Enterprise Edition
JCP	Java Community Process
JPDK	Java Portlet Development Kit
JSP	Java Server Pages
JSR 168	Java Specification Request # 168
KM	Knowledge Management
LDAP	Lightweight Directory Access Protocol
LoV	List Of Values
NLS	National Language Support
OASIS	Organization for the Advancement of Structured Information Standards
OLAP	Online Analytical Processing
PDA	Personal Digital Assistant
PDF	Portable Document Format (Adobe Acrobat)
PL/SQL	Procedural Language/Structured Query Language
QBE	Query By Example

R&D	Research & Development
RDBS	Routing Data Base System
SAP	Systeme, Anwendungen, Produkte in der Datenverarbeitung
SOAP	Simple Object Access Protocol (XML protocol)
SQL	Structured Query Language
SSO	Single Sign-On
TI	Technische Informatik
UDDI	Universal Description, Discovery and Integration
UI	User Interface
URL	Uniform Resource Locator
W3C	World Wide Web Consortium
WAP	Wireless Application Protocol
WI	Wirtschaftsinformatik
WML	Website Meta Language, Wireless Markup Language
WSDL	Web Services Description Language (IBM / Microsoft)
WSIA	Web Services for Interactive Applications
WSRP	Web Services for Remote Portals
XML	eXtensible Markup Language
XSLT	Extensible Stylesheet Language Transformations

9.4 Literaturverzeichnis

- [RGJM01] Oracle 9iAS Portal Bible
Rick Greenwald, James Milbery, 2001
Hungry Minds Verlag
ISBN 0-7645-4749-6
- [SVKC02] Oracle 9i Application Server Portal Handbook
Steve Vandivier, Kelly Cox, 2002
McGraw-Hill/Osborne, Oracle Press
ISBN 0-07-222249-2
- [XMLMA] XML & Web Services Magazin, 1.2002
"Java Web Services Developer Pack im Überblick", A. Holubeck

URL und Whitepapers

- [AMAWS] www.simple.ch/webservice
- [ASCII] <http://www.ascii.cl/htmlcodes.htm>
- [BCME01] Brian Cheney, Metagroup
"Portals Now and The Future", 2001
<http://www.mmltd.com/presentations/Meta%20Group%20OLA%203.pdf>
- [BDID01] Brian McDonough; IDC, 2001
"EIP- Software Market Revised Forecast and Analysis, 2001-2005"
<http://www.idc.com/getdoc.jhtml?containerId=26243>
- [CLFI00] Clive Finkelstein , *Information Engineering Services*
"The Enterprise: The Potential of Enterprise Portals", 2000
<http://www.dmreview.com/master.cfm?NavID=198&EdID=1847>

- [ECM02] E-Commerce Magazin
 „Enterprise-Portal-Markt wächst jährlich um 35 Prozent“, 2002
http://www.e-commerce-magazin.de/print.php?page=news-show_neu.php3&naechster=3769
- [EC-WS] ECIN
 “Webservices wollen Web erobern”, 2001
<http://www.ecin.de/technik/webservices/>
- [FGFO01] Frank Gillett, Forrester
 Making Enterprise Portals Pay, 2001
<http://www.plumtree.com/pdf/MakingEnterprisePortalsPay-aug.pdf>
- [FHPOR] http://lwis01.inf.fh-koeln.de:7778/servlet/page?_pageid=51&_dad=portal30&_schema=PORTAL30
oder
 “Mod_plsql Configuration Menu” →
 “Home”-Button → LOGIN (link)
- [GFTH99] Geoffrey Fox, Tom Haupt, NPAC
 Java for Computational Science, 1999
<http://www.npac.syr.edu/users/haupt/WebFlow/papers/JGtutorial/gcf/sld035.htm>
- [GIGA01] Giga Information Group
<http://www.gigaweb.com/homepage/>
- [GLOSSAR] <http://www.glossar.de>
<http://www.acronymfinder.com/>
<http://whatis.techtarget.com/>
- [GPGA01] G. Phifer, Gartner Group
 2H01 Portal Products Magic Quadrant, 2001
<http://www.sybase.es/productos/ebusiness/2H01PortalQuadrant.pdf>

- [GPRV02] G. Phifer, R. Valdes, D. Gootzit, Gartner Group
Big Change Evident in 2H02 Horizontal Portal Product MQ
[http://www.magnasistemas.com.br/magnasistemas/sbs1_3p.nsf/fb519bf0213f796003256893006f1ff0/612e771e481748de83256c0500686373/\\$FILE/2002HorizontalPortalProductMQ.pdf](http://www.magnasistemas.com.br/magnasistemas/sbs1_3p.nsf/fb519bf0213f796003256893006f1ff0/612e771e481748de83256c0500686373/$FILE/2002HorizontalPortalProductMQ.pdf)
- [JCP] <http://www.jcp.org/en/home/index>
- [JMF199] Joseph M. Firestone, Ph.D. (EIS Inc.)
“Defining the Enterprise Information Portal “, 1999
<http://www.hpcwire.com/dsstar/00/0822/102054.html>
- [JSBT99] John Schroeder, *Brio Technology*
“Enterprise Portals: A New Business Intelligence Paradigm”, 1999
<http://www.dmreview.com/master.cfm?NavID=55&EdID=1371>
- [LDAP1] <http://linuxline.epfl.ch/Doc/rhl-rg-de-7.0/ch-ldap.html>
- [LOTU] Lotus Notes Portlet Download
<http://portalstudio.oracle.com/pls/ops/docs/FOLDER/COMMUNITY/PDK/SAMPLES/WEBPORTLETS/LOTUSNOTESPORTLET/LOTUS.ZIP>
- [META01] Mete Group, 2001
„Der Markt für Knowledge Management in Deutschland“
http://www.community-of-knowledge.de/pdf/KM_Studie_metagroup_Kurzinfo.pdf
- [META02] Meta Group, Deutschland
<http://www.metagroup.de>
- [MIBU00] Michael P. Burwen, *PAMG*
„EIP: Bringing BI to the Masses”, 2000
<http://www.dmreview.com/master.cfm?NavID=198&EdID=2548>

- [O9PO02] Oracle Whitepaper April 2002
Oracle9iAS Portal Release 2 Portal Overview
<http://www.oracle.com/portals/index.html?intro.html>
- [OAI] <http://www.oracle.com/global/de/oai>
- [OASIS] <http://www.oasis-open.org>
- [OPAR] Oracle Portal Partner Initiative
<http://www.oracle.com/portals/partners/>
- [OPF300] Oracle Product Features
Oracle Portal 3.0 - Production Release, 2000
http://technet.oracle.com/products/iportal/pdf/oportal_feature_doc_10_10.pdf
- [OPSGS] Oracle Parallel Server Getting Started
<http://otn.oracle.com/doc/windows/server.804/a55925/toc.htm>
- [OTN02] <http://otn.oracle.com>
- [PAEN00] Paul Encarnación, Oracle
Oracle Portal Security Overview
http://technet.oracle.com/products/iportal/pdf/oportal_security_oview.pdf
- [PAMI00] Pankaj Mita, Oracle, 2000
“Building Scalable and Performant Portal Solutions Using Portal”
http://technet.oracle.com/products/iportal/pdf/build_scale_perform_portal.pdf
- [PDK02] 9iAS Portal Center
“Oracle Portal Development Kit”
<http://portalstudio.oracle.com/>
- [PLUM02] Plumtree Survey, 2002
“The Corporate Portal Market in 2002”
<http://dagda.shef.ac.uk/inf201/plumtreesurvey.pdf>

- [PORST] <http://portalstudio.oracle.com>
- [TSSH02] Thomas Schaeck, Stefan Hepper, *The Server Side*
“Portal Standards”, 2002
http://www2.theserverside.com/resources/article.jsp?l=Portlet_API
- [URLS] <http://www.w.fh-giessen.de/>
<http://www.gladbeck.de>
<http://wlmpportalprod.wlmp.com/>
<http://www.stadtbibliothek.pforzheim.de>
- [WAEC99] Wayne Eckerson
“15 Rules for Enterprise Portals”, 1999
<http://www.oracle.com/oramag/oracle/99-Jul/49ind.html>
- [WSCGE] Cap Gemini Ernst & young
“Web-Services: viel Gerede um nichts?”, 2002
<http://www.ecin.de/marktbarometer/studie-webservices/>
- [WSRP02] developer Works, IBM
Web Services for Remote Portals (WSRP), 2002
<ftp://www6.software.ibm.com/software/developer/library/ws-wsrp.pdf>