Software-Evaluation für ein Personendaten-Repositorium

von
Fabian Körner
Christoph Plutte
Torsten Roeder
Niels-Oliver Walkowski

Berlin-Brandenburgischen Akademie der Wissenschaften

gefördert durch die DFG – Deutsche Forschungsgemeinschaft

Berlin, April 2010

Working Draft

Inhaltsverzeichnis

1. ZIELE DER EVALUATION

1.1. AUSGANGSSITUATION

Nachdem lange Zeit im Rahmen von Open Access und anderen elektronischen Publikationsinitiativen die Bereitstellung von Dokumenten innerhalb von Repositorien im Zentrum stand, finden nun immer mehr auch die Veröffentlichungen von Forschungsdaten Beachtung. Dies führt nicht nur dazu, dass die Forschung transparenter wird und externe Wissenschaftler ihre eigenen Schlussfolgerungen aus dem Ausgangsmaterial ziehen können, sondern es ermöglicht auch die Aufwertung von dem primären Forschungsziel nachgestellten Material zu eigenständigen, vollwertigen, digitalen Ressourcen mit erheblichem wissenschaftlichen Mehrwert.

In einem solchen Zusammenhang ist die Entwicklung eines von der DFG geförderten Personendaten-Repositoriums an der Berlin-Brandenburgischen Akademie der Wissenschaften zu sehen. Im Rahmen geisteswissenschaftlicher Forschungsprojekte, die in zahlreichen voneinander unabhängigen Vorhaben organisiert sind, entstanden und entstehen umfangreiche Sammlungen von Personendaten. Eine erste Schätzung des Bestandes beläuft sich auf über 150.000 Datensätze historischer Personen. Personendaten stellen damit einen erheblichen Teil der wissenschaftlichen Produktion an der Akademie dar, und ihre Bedeutung für die Forschung, sei es bei der kritischen Edition von Texten oder bei der kulturhistorischen Kartographie einer Epoche, ist essenziell. Dennoch besitzt die Akademie bisher keine zentrale Infrastruktur, in der diese Daten abgelegt, organisiert und weiterverarbeitet werden können. Jedes Vorhaben hat in der Vergangenheit seine eigene Lösung realisiert. Dies ist nicht nur als ein Mangel an Vernetzung zu verstehen, sondern unterstreicht auch die Vielfältigkeit an Blickwinkeln und Zielen, mit der Forscher, in Abhängigkeit von ihrem Forschungsgegenstand, die für sie besten Inhaltsmodelle und technischen Lösungen suchen.

Eine Instanz eines zentralen Repositoriums muss auf jeden Fall in der Lage sein, diese Diversität an Perspektiven abzubilden, will sie qualitativ hochwertige Inhalte zur Verfügung stellen und den mit ihm arbeitenden Forschern einen Mehrwert bieten. Da angestrebt wird, die tägliche Arbeit mit Personendaten in das zu schaffende System zu überführen, wird darauf zu achten sein, dass die Daten auf einfachen Wegen erweiterbar und redigierbar sind. Da das Personendaten-Repositorium von vornherein in Kooperationen eingebunden ist, soll es so konzipiert sein, dass es als Infrastruktur mit allen Anpassungen, die im Rahmen dieses Projektes vorgenommen werden, weitergegeben und auf anderen Servern installiert werden kann, ohne dass es dabei seine Fähigkeit verliert, anpassbar zu sein. Wenn davon auszugehen ist, dass die erarbeitete Infrastruktur später an mehreren Standorten installiert werden kann, so erscheint es wünschenswert, dass die Repositorien verschiedener Institutionen in der Lage sind miteinander zu kommunizieren und von einer Instanz auf den Datenbestand einer anderen Instanz zugegriffen werden kann. Entsprechend der Vielschichtigkeit des Datenbestandes und seiner Hintergründe, aber auch auf Grund des besonderen Charakters von Personendaten, bedarf es einer besonderen Flexibilität im Zugriff auf und bei der Darstellung der Daten an der Benutzeroberfläche. Orts- und Zeitdimensionen von Personeninformation, insbesondere aber auch Beziehungen zwischen Personen, verlangen spezielle Suchalgorithmen und Ergebnisvisualisierungen, um den vollen Informationsgehalt zur Geltung zu bringen. Da sich auch die Berlin-

Brandenburgische Akademie der Wissenschaften, wie viele andere Wissenschaftsinstitutionen, in viele organisatorische Einheiten aufspalten lässt und die Herkunft der Daten innerhalb dieser Organisationsstruktur etwas über ihren Status aussagt, muss das System in der Lage sein, die dahinter liegenden Ordnungen mit aufzunehmen. Anders als Texte stellen Daten sehr viel stärker eine Grundlage für darauf aufbauende Forschung dar. Es besteht ein großes Bedürfnis, Bereiche aus ihnen zu selektieren und zu reorganisieren, um sie an das eigene Forschungsvorhaben anzupassen, sie aber auch mit eigenen Daten zu vergleichen, sie zu erweitern und zu überarbeiten. Zusammengefasst kann man davon ausgehen, dass der Umgang mit Daten wesentlich dynamischer ist als der Umgang mit Dokumenten oder audio-visuellen Dateien. Auf diesen Tatbestand muss ein Datenrepositorium mit besonders komfortablen und weitreichenden Browsing-, Editions- und Aggregationswerkzeugen reagieren. Gleiches trifft auf die Darstellung der Datenressourcen zu. Daten möchte man tendenziell nicht in der atomaren Form betrachten, sondern gruppiert und arrangiert, und das möglichst dynamisch und bedarfsorientiert. Dies ist deutlich mehr als die Ergebnisliste eines Dokumentenservers.

Der Anteil von Daten-Repositorien an Repositorien insgesamt ist nach wie vor sehr klein. Von den 1.682 in ROAR (Registry of Open Access Repositories) registrierten Repositorien sind lediglich 36 als Datenrepositorien klassifiziert (Stand 03/2010).[1] Erfahrungen mit der Nutzung zur Verfügung stehender Repositoriensoftware für den speziellen Fall der Administration von Datensätzen sind daher wenig vorhanden. Eine Evaluation dieser Software im Hinblick auf ihre Nutzbarkeit als Datenrepositorium erscheint umso notwendiger, als aus der beschrieben Situation heraus die Entwickler von Repositoriensoftware den Fokus ihrer Entwicklung in der Verwaltung von digitalen Ressourcen, zumeist Textdokumente, sehen mussten. Der Umgang mit Datensätzen ist aber ein gänzlich anderer, oder sollte ein gänzlich anderer sein, wenn der Informationsraum, welchen die Daten eröffnen, vollständig ausgeschöpft werden soll. Ansätze, in denen die Daten in einzelne Datensätze aufgespalten werden, um diese in Textdateien zu speichern und mit Metadaten zu versehen, die die Grundlage für eine Suche bilden, verschenken viel von dem Potenzial, welches in ihren Daten steckt. Diese Strategie versucht die Datenstruktur in eine Dokumentenstruktur zu überführen, um damit die originäre Arbeitseinheit von Repositorien wieder herzustellen. Anstelle dessen sollte für eine bestmögliche Nutzbarkeit der Daten im Prinzip jeder Datensatz als eigenständige Ressource behandelt werden können.

1.2. BEWERTUNGSKRITERIEN UND ABLAUF

Da in den letzten Jahren bereits eine Reihe an Evaluationen von Repositoriensoftware erschienen sind, soll es nicht darum gehen, den Kreis derjenigen einfach zu erweitern, die sich als Definitionsmacht in Sachen Institutionelle Repositorien positionieren möchten.[2] Ausgehend von der Überzeugung, dass die Benutzbarkeit und Qualität von Softwarelösungen am besten in Hinblick auf ein spezifisches Anwendungsszenario hin beschrieben werden kann, möchten wir einen Überblick über diesen Bereich für

1) http://roar.eprints.org/ (29.03.2010)

2) z.B.: Dobratz, Susanne: Open-Source-Software zur Realisierung von Institutionellen Repositories – Überblick. (Berlin: Humboldt-Universität zu Berlin, Zentraleinrichtung Universitätsbibliothek, 2007) http://nbn-resolving.de/urn:nbn:de:kobv:11-10081380 Repositories Support Project: Repository Software Survey. (2009) http://www.rsp.ac.uk/software/surveyresults. Stand: 2009-08-20. (Archiviert durch WebCite® at http://www.webcitation.org/5jALcy27v)

eine Aufgabenstellung geben, von der wir glauben, dass sie in Zukunft größere Aufmerksamkeit erhalten wird: die Nutzung von Repositoriensoftware für Datenrepositorien.[3]

In diesem Sinne erfolgt auch die Auswahl der von uns angelegten Bewertungskriterien. Für die allgemeinen Anforderungen, welche ein Repositorium erfüllen sollte, gibt es bereits weitreichende Orientierungshilfen durch Initiativen wie DINI[4], Open Access[5], DRIVER[6] oder OASIS[7]. Im vorliegenden Szenario geht es deshalb darum, die in diesen Initiativen proklamierten Anforderungen zu reformulieren. Dabei werden einige Kriterien, wie das Vorhandensein von XmetaDISS-Schnittstellen, wegfallen, da sie für Datenrepositorien wenig bis keine Relevanz haben.[8] Andere, wie ein flexibles Datenmodell, die Grundvoraussetzungen für Datenrepositorien sind, werden hinzukommen. Dieses Papier unterscheidet daher zwischen Kriterien, die von den oben genannten Initiativen mit dem Vorsatz, einen Standard zu definieren, erarbeitet wurden und solchen Kriterien, die sich aus der Benutzung der Repositorien als Personendaten-Repositorium der BBAW ableiten lassen. Um die Mannigfaltigkeit an Kriterien und Orientierungshilfen, die durch die oben genannten Initiativen geschaffen worden sind, zu Gunsten der Prägnanz und Übersichtlichkeit der Evaluation sinnvoll begrenzen zu können, reduzieren wir uns bei der Auswahl der Kriterien ebenfalls auf jene, die für Datenrepositorien in unserem Verständnis eine besondere Tragweite besitzen. Den Kriterienkatalog haben wir schließlich in drei größere Gruppen untergliedert, die durch die Kapitel 2, 3 und 4 repräsentiert sind. In einem allgemeinen Teil (Kapitel 2) betrachten wir zunächst nur den Entstehungshintergrund und die Nutzergemeinde des jeweiligen Repositoriensystems. Der zweite Teil (Kapitel 3) konzentriert sich auf die technischen Aspekte der Software, in dem die Installation, die Benutzerverwaltung und die Anpassbarkeit eingehend getestet wurde. Der dritte Teil (Kapitel 4) konzentriert sich schließlich auf die spezifischen Kriterien von Repositoriensystemen und den Anforderungen des Personendaten-Repositoriums im besonderen.

Wir haben uns dafür entschieden, bei der Evaluation selbst weitestgehend auf quantitative Methoden zu verzichten. Dies hat mehrere Gründe. Zum einen sind in einer Bewertung der Repositorien-Software über ein Punktesystem nie die Hintergründe präsent, aufgrund derer es zu der Bewertung gekommen ist. So bedeutet ein hohe Bewertung für Langzeitarchivierungs-Komponenten nicht zwingend, dass diese auch in jeder infrastrukturellen und organisatorischen Umgebung einer Institution gut anwendbar sind. Wirkliche Vergleichbarkeit kann ein quantitatives Verfahren daher auch nicht herstellen. Umgekehrt zeigt die Arbeit mit Repositorien, dass ein und dieselben Faktoren aufgrund der Vielfältigkeit der Einsatzumgebungen und der damit verbundenen Ziele zu ganz unterschiedlichen Bewertungen führen können. Eine quantitative Bewertung ist daher immer subjektiv, ohne dass diese Subjektivität in der Bewertung wirklich präsent ist. Zu guter Letzt wollen wir, wie bereits betont, mit diesem Papier auch die Dynamik der Diskussion um Datenrepositorien anregen. Dies lässt sich am besten auf der diskursiven Ebene erzielen. Wir grenzen uns

3) Zur Aktualität zur Thematisierung von Datenrepositorien siehe: nestor Arbeitsgruppe Grid/e-science und Langzeitarchivierung, nestor-bericht – Digitale Forschungsdaten bewahren und nutzen – für die Wissenschaft und die Zukunft. (Frankfurt am Main: nestor c/o Die Deutsche Bibliothek, 2009) http://nbn-resolving.de/nbn:de:0008-2009071031.

4) Deutsche Initiative für Netwerkinformation e.V., http://www.dini.de/ (29.03.2010)

5) http://open-access.net (29.03.2010)

6) Digital Repository Infrastructure for European Research, http://www.driver-community.eu

7) Organization for the Advancement of Structured Information Standards, http://www.oasis-open.org/

8) Das Repositorium als ganzes ließe sich vielleicht mit einer URN versehen, die einzelnen Daten zu adressieren macht jedoch aus ersichtlichen Gründen keinen Sinn und würde mit Sicherheit auch nicht akzeptiert werden.

daher auch von der Herangehensweise von Initiativen wie OA-Netzwerk ab, die zur Zeit ein Online-Tool für den Software-Vergleich anhand selektierbarer Eigenschaften entwickeln,[9] ohne die Qualität und Zweckmäßigkeit solcher Projekte für einen Großteil vorgefundener Interessenlagen in Abrede stellen zu wollen.

Jedes Kriterium soll zunächst kurz vorgestellt und unser Verständnis von ihm erläutert werden. Danach folgt für jedes der getesteten Systeme eine Beschreibung, wie weit und auf welche Art und Weise es die durch das Kriterium formulierte Aufgabenstellung erfüllt. Dargestellt werden soll dabei auch immer die Beziehung der Aufgabenerfüllung im Verhältnis zur Situation eines Datenrepositoriums im speziellen. Zur groben Orientierung erfolgt zu jedem Abschnitt ein Aussage darüber, ob eine Software die Aufgabe gar nicht, mit Einschränkungen oder relativ umfassend erfüllt.

1.3. VORAUSWAHL

Die Evaluation verlief in zwei Phasen. Zu Beginn wurde aus dem reichhaltigen Angebot an Repositorien-Software eine übersichtliche Liste mit Applikationen erstellt, die für eine erste intensivere Recherche als geeignet befunden wurden. Für die Vorauswahl wurden von uns einige elementare Kriterien festgelegt, die sich aus den Gegebenheiten an der Akademie der Wissenschaften, einigen grundlegenden Anforderungen im Umgang mit Daten und bestimmten Grundüberzeugungen zusammensetzen. Die Software musste auf jeden Fall als Open Source verfügbar sein. Da unter den oben aufgeführten Voraussetzungen davon auszugehen war, das umfangreiche Anpassungen unabhängig vom verwendeten System notwendig werden würden, sollte der Quellcode zur Verfügung stehen und darüber hinaus auch veränder- und erweiterbar sein. Um das Ergebnis später, wie angestrebt, auch an andere Institutionen weitergeben zu können, muss die verwendete Software unter einer entsprechenden Lizenz stehen. Zu guter Letzt kann der Fokus auf Open-Source-Software auch in Zusammenhang mit der von der Berlin-Brandenburgischen Akademie der Wissenschaften unterzeichneten „Berliner Erklärung über den offenen Zugang zu wissenschaftlichem Wissen“[10] betrachtet werden. Davon ausgehend, dass die Bedeutung und Wirkung von Informationen nie völlig unabhängig von der Umgebung ist, in der sie präsentiert wird, kann die Wahl von Open Source dadurch, dass es ein potenziell transparentes System liefert, einen erweiterten offenen Zugang bieten. Aus diesen Gründen scheiden neben klassisch proprietärer Software z.B. auch Systeme wie Microsofts *Zentity*[11] aus, da dieses zwar kostenlos nutzbar, der Quellcode aber nicht frei verfügbar ist.

Ein weiterer wichtiger Punkt im Umgang mit Daten ist die Möglichkeit ein flexibles und vom Benutzer anpassbares Content- und Metadatenmodell zu nutzen. Daten, im vom uns verwendetem Sinn, sind weder Dokumente noch audio-visuelle Ressourcen, und will man sie nicht als solche behandeln, dann sind die von den meisten Systemen verwendeten Beschreibungsstandards *Dublin Core, MODS* und *METS* nicht angemessen. Hier wäre die nachträgliche Integration des Datenmodells der *Data Documentation Initiative*[12]

9) Das Tool ist bisher noch nicht freigegeben, sondern wurde auf dem Workshop „Vernetzungstage 09“ vorgestellt. http://www.dini.de/fileadmin/workshops/oa-netzwerk-juni2009/vernetzungstage_2009_severiens2.pdf
10) http://oa.mpg.de/openaccess-berlin/berlindeclaration.html (29.03.2010)
11) http://research.microsoft.com/en-us/projects/zentity/ (29.03.2010)
12) http://www.ddialliance.org/ (29.03.2010)

ein gangbarer Weg. Noch schwieriger stellt sich die Situation dar, möchte man, wie in unserem Projekt vorgesehen, Teile oder die gesamten Daten auf der Ebene, in der sonst die Metadaten festgehalten werden, abbilden. Daher scheiden Repositoriensysteme, die kein flexibles Datenmodell mit sich bringen, von vornherein aus. Daneben waren für die Aufnahme in die vorübergehende Liste verbreitete Faktoren wie eine aktive Entwicklergemeinschaft, die Verwendung zukunftssicherer Techniken sowie einer ausführlichen Dokumentation in einer internationalen Sprache ausschlaggebend. Eine Ausnahme bilden hier Systeme aus Deutschland, da die Entstehung des Personendaten-Repositoriums in der deutschen Forschungslandschaft verankert ist und hier auch Anbindung sucht.

In die vorläufige Liste wurden die folgenden sieben Angebote aufgenommen:

- DSpace
- EPrints
- eSciDoc
- Fedora Commons
- Greenstone
- MyCoRe
- OPUS

In dieser ersten Phase wurde die Software (außer im Falle von MyCoRe) noch nicht installiert. Vielmehr ging es darum, durch eine umfangreiche Recherche der Dokumentation, der Evaluation von Benutzerfahrungen sowie durch Entwicklerangaben die Grundlage für eine engere Selektion von vier Systemen zu schaffen, die dann auf einem Testserver zum Einsatz kommen, und in einem Anwendungsszenario beobachtet werden sollten.

1.4. ENGERE AUSWAHL

Aus der oben genannten Liste fallen drei Systeme heraus, welche aus sehr unterschiedlichen Gründen nicht in die engere Auswahl mit aufgenommen wurden. Dies betrifft eSciDoc, myCoRe und OPUS.

eSciDoc ist ein System, welches auf Fedora Commons basiert. Es verfolgt unter anderem das Ziel, die Basisfunktionalitäten von Fedora zu erweitern und dem Entwickler einen Teil der Anpassungsarbeit abzunehmen, die aufgrund des minimalistischen Ansatzes von Fedora erheblich sein kann. eSciDoc teilt sich in zwei Bereiche: die eSciDoc Infrastructure und die eSciDoc Solutions. Die Infrastructure stellt dabei eine Basisarchitektur dar, auf der für spezifische Szenarien die Solutions aufgesetzt werden können. Bisher entwickelte Solutions konzentrieren sich auf den Anwendungsbereich von Dokumenten- und Multimedia-Repositorien. Daher kommt die Nutzung einer Solution für ein Datenrepositorium nicht in Frage. Auch die Entwicklung einer eigenen Solution auf der Basis der Infrastructure wurde von uns verworfen, da letztere bereits Anpassungen in eine Richtung vornimmt, die für ein Datenrepositorium im besten Fall überflüssig und im schlechtesten Fall einschränkend sind. So könnte in unserem Fall der Service *Duplicate Detection* zu Problemen führen, da es gleiche Datensätze geben kann, die sich jedoch in einem anderen Kontext

befinden. Ein weniger erheblicher, aber dennoch wichtig zu erwähnender Punkt ist die etwas dürftige und teils fehlerhafte Dokumentation, was von den Entwicklern bereits erkannt, aber noch nicht behoben wurde.

MyCoRe bietet ein Repositorien-Framework und – abgesehen von der mitgelieferten Beispielanwendung „DocPortal“ – keine Out-of-the-Box-Lösung. Insofern hätte es sich als Vergleichssystem zu Fedora Commons angeboten, da sonst ausschließlich Out-of-the-Box-Systeme evaluiert wurden, und sollte zunächst in die engere Auswahl mit aufgenommen werden. Bei der ersten Installation war es möglich, einen Blick auf DocPortal zu werfen; ein zweitenr Installationsversuch ohne DocPortal blieb leider erfolglos. Weder die Wahl des Installationsmediums (herunterladbare Pakete, svn-Version), noch die exakt reproduzierte Vorgehensweise der ersten Installation, noch die Lektüre der, teilweise veralteten, Dokumentation und des Wikis, noch die Bemühung sonstiger Informationsquellen führten zum Erfolg. Diese Umstände hätten auch bei einem letztendlich funktionierenden System gegen eine detailliertere Evaluation gesprochen, da in unserem Fall die Distribution an andere Institutionen ein zentrales Anliegen ist und sich der damit verbundene Integrations- und Unterstützungsaufwand in klaren, engen Grenzen bewegen muss. MyCoRe bietet darüber hinaus keine große Community, aus der Unterstützung zu ziehen ist. Es wurde daher durch Greenstone ersetzt.

Gegen eine Aufnahme von OPUS sprachen gleich mehrere Gründe. Zum einen stand die Software im Zeitraum der Evaluation nur gegen eine Lizenzgebühr zur Verfügung, was bereits ein kategorisches Ausschlusskriterium war.[13] Zum anderen konzentriert sich OPUS weitestgehend ausschließlich auf die Anforderungen von Bibliotheken und unterstützt daher auch nur die entsprechenden Schnittstellen. Darüber hinaus stand zum Entscheidungszeitpunkt eine Neuherausgabe der Software (Version 4) noch kurz bevor, so dass eine Evaluation der neuen, in Entwicklung befindlichen Version auf keiner stabilen Grundlage hätte stehen können, während eine Evaluation der stabilen, aber alten Version ebenso bald hinfällig gewesen wäre.[14]

In die engere Auswahl fallen daher die Systeme DSpace, EPrints, Fedora und Greenstone (alphabetische Reihenfolge).

13) http://elib.uni-stuttgart.de/opus/doku/opus_sw.php (10.11.2009)
14) OPUS 4 steht derzeit als Entwicklungsrelease zur Verfügung, http://samos.bsz-bw.de/ (31.03.2010)

2. ALLGEMEINE EVALUATION

2.1. ÜBERBLICK

Im Sinne einer Einführung umreißen wir in den folgenden Absätzen die verschiedenen Repositoriensysteme hinsichtlich ihrer bisherigen Geschichte, ihres Standortes und ihrer Entwickler-Kerngruppe. Außerdem stellen wir kurz die aktuellen bzw. von uns verwendeten Versionen und deren Lizenzmodelle vor. Besonderheiten hinsichtlich des Paket-Umfanges, wie z.B. mitgelieferte Frontends, oder spezielle institutionelle Anbindungen erwähnen wir ebenfalls. Abschließend nennen wir die gängigsten Informationsquellen für aktuelle Entwicklungen.

2.1.1. DSpace

DSpace wird seit 2002 entwickelt und ging aus einer gemeinnützigen Initiative von *Hewlett Packard* und dem *Massachusetts Institute of Technology* hervor.[15] Sitz der Entwicklerorganisation *DSpace Foundation* ist Cambridge in Massachusetts. Sie gehört der Dachorganisation *DuraSpace* an, welche weitere Repositorien-Frameworks und Datenbank-Systeme wie Fedora und Mulgara herausgibt.[16] Die Software ist BSD-lizenziert[17] und wird als Out-of-the-Box-System ausgeliefert. Die stabile, von uns getestete Version trägt die Nummer 1.5.2, die aktuelle, jedoch noch in der Entwicklung befindliche, ist die Version 2.0. (Im Gegensatz zum obigen Fall von OPUS steht die Veröffentlichung von DSpace 2.0 allerdings nicht unmittelbar bevor.)

Die Entwicklergemeinde hat ihren Kern an amerikanischen Universitäten,[18] deren hohe Aktivität ein Issue Tracker belegt.[19] Gutzuheißen ist außerdem das Prinzip des Community-Development, entsprechend dem der Apache Foundation, welches für eine noch lang andauernde Aktivität spricht. Auch die Einbindung in übergreifende Entwicklungskontexte wie DuraSpace erscheint hinsichtlich der langfristig erwartbaren System-Interoperabilität günstig. Über aktuelle Entwicklungen informieren der monatliche Newsletter „NewSpace“[20] und der Nachrichtendienst *Twitter*[21].

2.1.2. EPrints

EPrints wird seit dem Jahr 2000 an der *School of Electronics and Computer Science* an der *University of Southampton* entwickelt.[22] Es entstand innerhalb der OAI (Open Archives Initiative) und ist insofern darauf ausgelegt, Bestände aus verschiedenen Archiven aufzunehmen. EPrints ist ein Out-of-the-Box-System und wird für Linux unter der aktuellen GPL-Lizenz herausgegeben,[23] für Windows-Systeme greift hingegen die

15) http://www.dspace.org/ (10.12.2009)
16) http://duraspace.org/ (03.12.2009)
17) http://www.opensource.org/licenses/bsd-license.php (03.12.2009)
18) http://wiki.dspace.org/index.php/DspaceProjects (03.12.2009)
19) Dieser dokumentiert statistisch den Bearbeitungsstand von anstehenden Programmierarbeiten. http://jira.dspace.org/jira/secure/Dashboard.jspa (03.12.2009)
20) http://www.dspace.org/newsletter-newspace/newspace/ (03.12.2009)
21) http://twitter.com/dspacetweets (03.12.2009)
22) http://www.eprints.org/ (10.12.2009)
23) General Public License 3.0, http://www.gnu.org/licenses/gpl-3.0.txt (08.12.2009)

Microsoft Reciprocal License (Ms-RL).[24] Die Tatsache, dass bei der Entwicklung auf EPrints basierenden Software im Hinblick auf Installationen in anderen Serverumgebungen möglicherweise beide Lizenzmodelle gehandhabt werden müssten, könnte dabei zu Komplikationen führen. Die von uns getestete Version trägt die Nummer 3.1.3, welche seit Mai 2009 zur Verfügung steht.[25]

Über den Verlauf der Weiterentwicklung oder ein etwaig geplantes Major Release von EPrints liegen keine öffentlichen Informationen vor. Neuigkeiten über EPrints lassen sich jedoch über die Mailingliste *Eprints-announce*[26] und über den Blog *EPrints News*[27] verfolgen. Öffentlich ist EPrints sowohl auf der *Open Repositores Conference* als auch auf der *European Conference on Digital Libraries* (ECDL)[28] vertreten. Erwähnenswert ist außerdem, dass *EPrints Services* eine Promotion in „Web Science“ an der ECS fördert.[29]

2.1.3. Fedora Commons

Fedora Commons, oder kurz: Fedora (nicht zu verwechseln mit der gleichnamigen Linux-Distribution), ist das Akronym für „Flexible Extensible Digital Object Repository Architecture“.[30] Fedora wird seit 1997 an der *Cornell University* in Ithaca (New York State) entwickelt. Es ist als Repositorien-Framework konzipiert und liefert daher – im Gegensatz zu den anderen evaluierten Systemen – keine Out-of-the-Box-Lösung, was allgemein gesprochen sowohl einen Nachteil als auch einen Vorteil bedeuten kann. Hier muss der Entwicklungsaufwand mit dem Anpassungsaufwand bei den anderen Systemen verglichen werden. Die aktuelle, von uns getestete Version ist das Release 3.2.1 und wird unter der Apache License herausgegeben.[31]

Wie bei DSpace hat die Entwicklergemeinde von Fedora ihren Kern an amerikanischen Universitäten. Auch die Einbindung in übergreifende Entwicklungskontexte wie *DuraSpace* erscheint hinsichtlich der langfristig erwartbaren System-Interoperabilität als begünstigend. Aktuelle Informationen über das Projekt werden über eine Mailingliste[32], den DuraSpace-Blog[33] und über den Nachrichtendienst Twitter[34] verbreitet. Nachrichten in gebündelter Form sind über den Newsletter Hatcheck zugänglich.[35]

2.1.4. Greenstone

Greenstone ist wie *Fedora Commons* ein Repositorien-Framework, welches seit ca. 2001 von der *University of Waikato* in Neuseeland in Kooperation mit der UNESCO und der *Human Info NGO* in Antwerpen

24) http://wiki.eprints.org/w/Installing_Eprints_3_on_Windows (09.12.2009). Der Lizenztext findet sich unter http://www.microsoft.com/opensource/licenses.mspx (09.12.2009)
25) http://www.eprints.org/software/ (08.12.2009)
26) http://mailman.ecs.soton.ac.uk/mailman/listinfo/eprints-announce (09.12.2009)
27) http://eprintsnews.blogspot.com/ (09.12.2009)
28) http://www.ionio.gr/conferences/ecdl2009/ (09.12.2009)
29) http://webscience.ecs.soton.ac.uk/dtc/ (09.12.2009)
30) http://www.fedora-commons.org/ (10.12.2009)
31) http://www.fedora-commons.org/software/licenses (10.12.2009)
32) https://lists.sourceforge.net/lists/listinfo/fedora-commons-users (10.12.2009)
33) http://expertvoices.nsdl.org/duraspace/category/humanities/ (10.12.2009)
34) http://twitter.com/FedoraRepo (10.12.2009)
35) http://www.fedora-commons.org/community/hatcheck (21.12.2009)

entwickelt und herausgegeben wird.[36] Die Einbindung in eine internationale Organisation ist eine Besonderheit von Greenstone, die es den anderen Systemen voraus hat, besonders im Hinblick auf Sprachunterstützung und Diversität der Nutzergemeinde.

Greenstone ist GPL-lizenziert und wird inzwischen in der dritten Generation entwickelt. Das Stable Release, welches von uns getestet wurde, trägt die Nummer 2.83. Ein für Greenstone entwickeltes Frontend, „EmeraldView“, ist als gesondertes Paket verfügbar.[37] Über aktuelle Entwicklungen informieren mehrere Blogs und mehrere Mailinglisten in verschiedenen Sprachen.[38]

2.2. COMMUNITY

Unter dem Begriff „Community“ haben wir unsere Erkenntnisse über das geographische und inhaltliche Spektrum sowie über die Aktivität des Nutzerkreises zusammengefasst. Als begünstigend erachten wir vor allem eine geographisch weiträumige Verteilung bzw. eine international geprägte Nutzergemeinde, einen regen Austausch unter den Nutzern sowie nicht zuletzt eine aktive Mitarbeit der Nutzer an der Weiterentwicklung der jeweiligen Repositoriensoftware. Aussagen über die Verteilung haben wir vor allem über die Herausgeber der Software, aber auch über den Informationsdienst ROAR[39], abgeleitet. Einen Eindruck von der Aktivität gewannen wir indessen vor allem über Austauschplattformen wie z.B. Mailinglisten und FAQs. Außerdem wurde nach Projekten Ausschau gehalten, die ähnliche Anwendungsszenarien wie das Personendaten-Repositorium einschließen.

2.2.1. DSpace

DSpace wird vor allem von Universitäten als Publikationsserver genutzt. Heute kann DSpace etwa 700 Instanzen in 70 Ländern nachweisen, allerdings nur 11 davon im deutschsprachigen Raum.[40] Die größten Nutzergemeinden finden sich laut ROAR in den USA und in Japan.[41]

Die meisten Projekte, auch diejenigen in Deutschland, sind durch ihre Anbindung an Universitäten nicht thematisch, sondern institutionell orientiert. Lediglich ein deutsches Projekt kann als dezentral betrachtet werden, das *Bürgerarchiv zur Geschichte des Alltags in den Hamburger Stadtteilen*.[42]

Die Community von DSpace darf als sehr aktiv eingestuft werden. In den USA und in Europa finden regelmäßig „User Group Meetings“ statt, zuletzt in Göteborg und Atlanta (2009),[43] die mit der jährlichen *Open Repositories Conference* koordiniert wird.[44]

---

36) http://www.greenstone.org/ (17.12.2009)
37) http://emeraldview.tourolib.org/ (17.12.2009)
38) http://wiki.greenstone.org/wiki/index.php/GreenstoneWiki (17.12.2009)
39) http://roar.eprints.org/ (01.02.2010)
40) http://www.dspace.org/whos-using-dspace/Repository-List.html (03.12.2009)
41) http://roar.eprints.org/index.php?action=generate_chart&q=&chart_type=pie&chart_field=country&country=&version=dspace&type=&submit=Filter (08.12.2009)
42) http://stadtteilgeschichten.net/ (03.12.2009)
43) http://www.dspace.org/user-group-meetings/user-group-meetings/ (03.12.2009)
44) http://www.openrepositories.org/ (03.12.2009)

2.2.2. EPrints

Dieses System wird vorzugsweise für die Verwaltung und Veröffentlichung von Bibliotheks- und Archivbeständen vor allem im englischsprachigen Raum, aber auch in den Ländern des europäischen Festlandes verwendet. Aufgrund der einfachen Handhabbarkeit und der Auslieferung als Out-of-the-Box-System erfährt es auch in Deutschland beachtliche Verbreitung und behauptet sich dort laut ROAR als am zweithäufigsten verwendetes Repositoriensystem nach OPUS.[45] Weltweit kann EPrints nach eigenen Angaben 269 Projekte nachweisen,[46] laut ROAR sind es sogar über 350, was es an zweite Stelle nach DSpace rückt.

Die meisten auf EPrints basierenden Repositorien werden für Publikationen verwendet und sind entsprechend der Intention der Software in vielen Fällen auch institutsübergreifend ausgelegt. Da EPrints für diese Zwecke bereits in der Standardinstallation gute Dienste leistet, sind individuell konfigurierte Projekte nur selten zu finden. Webangebote wie *The Linnean Collections*[47] und *Language Box*[48] demonstrieren jedoch, dass EPrints auch mit nicht-bibliographischen und sogar multimedialen Inhalten umgehen kann und dass eine freie Anpassung der Oberflächengestaltung durchaus möglich ist.

Über eine aktive Entwickler-Community ist nichts bekannt, was für ein Open-Source-Projekt ungewöhnlich ist. Wikipedia zufolge gibt es seitens der Entwickler jedoch auch kein Bestreben oder Interesse, dies zu ändern.[49] Im Zusammenhang mit dem kostenpflichtigen Dienstleister *EPrints Services* positioniert sich EPrints, dem Geschäftsmodell des Open Source entsprechend, als Entwickler einer quelloffenen Software und kommerzieller Dienstleistungsanbieter.

2.2.3. Fedora Commons

Die meiste Verbreitung findet Fedora im englischsprachigen Raum und in Mitteleuropa, wo die meisten Projekte im deutschsprachigen Bereich lokalisierbar sind. Einerseits erscheint dies im direkten Vergleich mit anderen Systemen eher wenig zu sein, andererseits ist es gerade eine beachtliche Zahl, da es sich bei Fedora schließlich um ein Framework handelt, welches im Gegensatz zu z.B. EPrints und DSpace keine fertige Oberfläche mitliefert. Insofern werden nur solche Projekte Fedora verwenden, die über die Ressourcen für die Entwicklung einer eigenen Oberfläche verfügen und eine individuell anpassbare Arbeitsoberfläche benötigen.

Nach eigenen Angaben existierten im Mai 2009 insgesamt 165 Projekte, die Fedora als Framework verwenden.[50] Die große Differenz zu den Angaben bei ROAR (16)[51] und *Repository Maps* (22)[52] ist

45) Registry of Open Access Repositories, http://roar.eprints.org/index.php?action=generate_chart&country=de &version=&type=&chart_field=version&chart_type=pie&submit=Generate+Chart (08.12.2009)
46) http://www.eprints.org/software/archives/ (08.12.2009)
47) http://www.linnean-online.org/ (10.12.2009)
48) http://languagebox.eprints.org/ (10.12.2009)
49) http://en.wikipedia.org/wiki/EPrints (10.12.2009)
50) https://fedora-commons.org/confluence/display/FCCommReg/Fedora+Commons+Registry (10.12.2009)
51) http://roar.eprints.org/?action=home&q=&country=&version=fedora&type=&order=name&submit=Filter (10.12.2009)
52) http://maps.repository66.org/ (10.12.2009)

möglicherweise durch unterschiedliche Kriterien zu erklären, nach denen die Repositorien in die jeweilige Statistik aufgenommen werden.

Projekte wie *eSciDoc*[53] oder und *Islandora*[54] belegen, dass sich auf der Basis von Fedora offene Architekturen für Repositoriensysteme aufbauen lassen. Außerdem zeigen derartige Projekte, dass eine sehr rege und weitgefächerte Community hinter Fedora steht, und dass Fedora auch Unterstützung von offiziellen Trägern erfährt. Das Projekt Islandora kombiniert beispielsweise Fedora mit dem Drupal CMS und spricht damit auch die große Drupal-Community an. eSciDoc hingegen wird vom Bundesministerium für Bildung und Forschung unterstützt.

Fedora Commons fördert zudem die Zusammenarbeit seiner Nutzer. Eine Übersicht aller Community-Ressourcen steht auf der offiziellen Fedora-Webseite zur Verfügung, welche in verschiedene Interessengruppen gegliedert ist.[55] Die Nutzer können über dieses Portal untereinander in Kontakt treten.

2.2.4. Greenstone

Greenstone kann weltweit mehr als 60 Vorhaben nachweisen, die das System verwenden.[56] Eine herausragende Besonderheit der Benutzergemeinde ist, dass sie nicht nur in den westlichen Ländern, sondern in allen Regionen der Erde vertreten ist. Im Jahr 2006 wurde eine Umfrage über die Nutzergemeinde durchgeführt, die leider mangels Aktualität und Repräsentativität keine genaueren Aussagen über die Community zulässt.[57] Auch den Angaben bei ROAR ist angesichts der weit divergierenden Angaben (dort werden lediglich 10 Repositorien gezählt) ebenfalls kein repräsentativer Wert beizumessen.[58]

Die Vielsprachigkeit ist wohl das auffallendste Merkmal der Greenstone-Community. Nicht nur der Nutzerkreis ist international, sondern auch die Internationalisierung der Software selbst wird stark gefördert, was eine Besonderheit unter Repositorien-Systemen ist, die üblicherweise nur auf Englisch verfügbar sind. Die UNESCO unterstützt die Übersetzung des Programmkerns und anderer Bestandteile in die Sprachen Englisch, Russisch, Spanisch und Französisch,[59] etwa 50 weitere Sprachen werden von sogenannten *volunteers* übernommen.[60]

2.3. DOKUMENTATION UND SUPPORT

In diesem Bereich behandeln wir die allgemeine Qualität der vorliegenden Dokumentationen für die Repositorien-Software im Hinblick auf Umfang, Vollständigkeit und Aktualität. In die Betrachtung fließen außerdem auch Support- und Selbsthilfemöglichkeiten wie etwa Mailinglisten oder FAQs, aber auch

53) https://www.escidoc.org/ (01.02.2010)
54) http://islandora.org/ (01.02.2010)
55) http://www.fedora-commons.org/community (10.12.2009)
56) http://www.greenstone.org/examples (17.12.2009)
57) http://www.ils.unc.edu/~sheble/greenstone/survey-report.html (17.12.2009)
58) http://roar.eprints.org/view/software/greenstone.html (01.02.2010)
59) http://wiki.greenstone.org/index.php/Greenstone_language_support (17.12.2009)
60) http://www.greenstone.org/gti/status.html (17.12.2009)

Angebote kommerzieller Dienstleister ein. Besondere Service-Konzepte oder Features, wie z.B. Mehrsprachigkeit, sollen ebenfalls berücksichtigt werden.

2.3.1. DSpace

Zu DSpace liegen ausführliche Dokumentation über alle Versionen vor.[61] Nutzer von DSpace können sich außerdem über drei Mailinglisten austauschen (untergliedert in die Zweige *General*, *Technical* und *Development*).[62] Fragen werden dort üblicherweise innerhalb eines Tages beantwortet. Zudem finden sich auf der DSpace-Seite umfangreiche Ressourcen wie Tutorials, Lehrmaterial, Präsentationen, Vergleiche mit anderen Systemen und ferner auch Links zu Publikationen, die sich mit DSpace auseinandersetzen. Insgesamt darf DSpace damit zu den am besten dokumentierten Repositorien-Frameworks gezählt werden, welches zudem einen beachtlichen Niederschlag in der Forschungsliteratur findet.[63]

DSpace hat eine eigene Support-Initiative, die sich *DSpace Ambassador Program* nennt.[64] In Deutschland operiert lediglich ein „Ambassador".[65] Es gibt außerdem mehrere Service-Provider, die allerdings nicht in Deutschland operieren.[66] Insofern müssen deutsche Nutzer vor allem auf Unterstützung über die oben genannten Mailinglisten zurückgreifen.

2.3.2. EPrints

EPrints bietet neben einer Wiki-Dokumentation[67] einen gut strukturierten, selbstarchivierenden FAQ-Bereich[68] an. Lernmaterial zu den Bereichen Installation, Konfiguration und Endbenutzung stehen ebenfalls zur Verfügung.[69] Außerdem gibt es eine freie Mailingliste für technische Fragen, deren Aktivität für eine rege Nutzergemeinde spricht.[70] EPrints veranstaltet jährlich einen Tages-Workshop für verschiedene Nutzergruppen.[71]

Professioneller, kostenpflichtiger Support wird von der Dienstleistungsgruppe *EPrints Services* angeboten,[72] die außerdem mehrere Training Courses im Jahr anbietet.[73]

2.3.3. Fedora Commons

Fedora bietet neben ausführlichen Dokumentationen in englischer Sprache[74] auch eine sehr umfangreiche Einführung in das Konzept und das Datenhaltungsmodell an. Dies ist zu begrüßen, da Fedora damit zeigt,

---

61) zur Version 1.5.2: http://www.dspacedev2.org/1_5_2Documentation/ (03.12.2009)
62) http://www.dspace.org/Mailing-List-Summary.html (03.12.2009)
63) http://www.dspace.org/publications/publicationsnonfoundation/ (21.12.2009)
64) http://wiki.dspace.org/index.php/DSpace_Ambassador_Program (03.12.2009)
65) Joachim Räth in Hamburg, der auch mit dem o.g. Bürgerarchiv befasst ist.
66) http://www.dspace.org/service-providers/Service-Providers.html (03.12.2009)
67) http://wiki.eprints.org/ (09.12.2009)
68) http://www.eprints.org/openaccess/self-faq/ (09.12.2009)
69) http://www.eprints.org/software/training/ (09.12.2009)
70) http://www.eprints.org/tech.php/ (09.12.2009)
71) http://eprintsnews.blogspot.com/2009/03/free-training-course-for-tecchies.html (09.12.2009)
72) http://www.eprints.org/services/ (09.12.2009)
73) http://www.eprints.org/services/training/ (09.12.2009)
74) http://www.fedora-commons.org/confluence/display/FCR30 (22.12.2009)

dass es weniger einen zweckorientierten als einen allgemeingültigen Ansatz verfolgt, der wie viele Repositorien-Systeme aufgrund ihres Entstehungskontextes nicht nur z.B. auf Literaturdatenverwaltung zugeschnitten ist, sondern sich auch auf andere Sachverhalte übertragen ließe.

Supportmöglichkeiten gibt es vor allem über die rege Community. Dort ist z.B. eine Mailingliste für Anwender zu finden, auf der Fragen schnell beantwortet werden.[75] Besonders interessant ist außerdem die Möglichkeit, sich als Fedora-Benutzer zu registrieren und Kontakt zu anderen Nutzern aufzunehmen, die z.B. mit ähnlichen Projekten befasst sind. Support wird außerdem von kostenpflichtigen Dienstleistern angeboten, die über die Fedora-Website zu finden sind.[76]

2.3.4. Greenstone

Greenstone ist insgesamt sehr umfangreich dokumentiert. Es liegen Handbücher für User und Developer vor, sowie zahlreiche Tutorials und ein Forum mit ca. 15.000 Einträgen. In dem offiziellen Greenstone-Wiki werden Handbücher in sieben Sprachen für das letzte Stable Release angeboten, die allerdings zum letzten Zeitpunkt des Abrufs nicht alle verfügbar waren.[77] Ein innerhalb dieses Wikis eingerichteter, sehr übersichtlich strukturierter FAQ-Bereich ist außerdem direkt über die Projekt-Homepage erreichbar.[78] Abgesehen davon ist Greenstone das einzige Projekt, welches Support in mehreren Sprachen wie z.B. Spanisch[79] und Französisch[80] anbietet. Daneben gibt es noch die „Regional Support Groups“ für das südliche Afrika und für Südasien. In Deutschland oder in deutscher Sprache gibt es bislang keinen Support. Darüber hinaus bieten gleich mehrere kommerzielle Dienstleister ihre Expertise mit Greenstone-Repositorien an, was für hohe Flexibilität und einfache Handhabung des Systems sprechen kann.[81] Mindestens einmal im Jahr findet zudem ein offizieller Greenstone Workshop statt, bislang meist im pazifischen Raum.[82]

75) http://www.fedora-commons.org/community/userlist (22.12.2009)
76) http://fedora-commons.org/confluence/display/SVCPROV/Home (21.12.2009)
77) http://wiki.greenstone.org/wiki/index.php/Manual (17.12.2009)
78) http://wiki.greenstone.org/wiki/index.php/Greenstone_FAQ (31.03.2010)
79) http://greenstone.infonautica.net/ (17.12.2009)
80) http://www.greenstone.fr/ (17.12.2009)
81) http://www.greenstone.org/support (31.03.2010)
82) http://wiki.greenstone.org/wiki/index.php/Greenstone_workshops (17.12.2009)

3. TECHNISCHE EVALUATION

3.1. INSTALLATION

Unter diesem Abschnitt fassen wir zunächst die wesentlichen Anforderungen der Repositorien-Systeme an die Server-Umgebung zusammen. Ferner dokumentieren wir den Verlauf der Installation und ggf. dabei zutage getretene Besonderheiten.

3.1.1. DSpace

DSpace stellt seine Software zum Download über Sourceforge[83] zur Verfügung und bietet außerdem einen direkten Zugang zu den Quelltexten über einen Subversion-Server[84] an. Die erstgenannte Quelle stellt ebenfalls vorkompilierte Versionen bereit, auf die hier, wie bei den anderen Kandidaten dieser Evaluation auch, aus Gründen der allgemeinen Vergleichbarkeit, zurückgegriffen wurde.

Voraussetzungen

Für den Betrieb von DSpace (1.5.2) benötigt man ein UNIX(-Derivat) oder Microsoft Windows mit installiertem Java JDK ab Version 5. Möchte man die Software lokal kompilieren, benötigt man laut Dokumentation für die Erzeugung eines Installationspaketes Apache Maven ab Version 2.0.8[85] sowie Apache Ant ab Version 1.6.2.[86] Maven wird im übrigen auch dann benötigt, wenn man das Installationspaket nicht lokal kompilieren möchte.

Als RDBMS können Oracle ab Version 9 oder PostgreSQL ab Version 7.3 verwendet werden. Wegen des vergleichsweise geringen Aufwandes der Installation und Konfiguration und mit Blick auf ein ggf. produktives System wurde in der vorliegenden Evaluation PostgreSQL in der Version 8.3.8 verwendet.[87] Da sich das RDBMS auf demselben System befindet wie die anderen DSpace-Komponenten, genügt es, der Dokumentation zu folgen und zwei kleine Änderungen in den PostgreSQL-Konfigurationsdateien vorzunehmen. Bei einer Verteilung der Komponenten auf mehrere Server können diese Änderungen, abhängig von der Netzwerkstruktur, etwas komplizierter ausfallen. In unserem Fall ist das Datenbanksystem nach einem Neustart für den weiteren Installationsprozess von DSpace vorbereitet.

Als Application Server kommen Apache Tomcat ab Version 4.x, Jetty oder Caucho Resin in Frage. In dieser Evaluation wird Tomcat (5.5.28) verwendet. Im Gegensatz zu den beiden anderen Möglichkeiten sind dazu zwei kleine (und gut dokumentierte) Einstellungen nötig, welche die Größe des für die virtuelle Java Maschine verfügbaren Arbeitsspeicherbereiches und die Zeichenkodierung der verwendeten URIs betreffen.

Da aus Sicherheitsgründen eine Server-Anwendung niemals als `root`-Benutzer mit allen damit verbundenen Rechten ausgeführt werden sollte, haben wir zunächst einen Nutzer `DSpace` angelegt. Die Pakete von

83) http://sourceforge.net/projects/dspace/files/ (13.01.2010)
84) http://scm.dspace.org/svn/repo/dspace/tags/dspace-1.5.2/ (13.01.2010)
85) http://maven.apache.org (13.01.2010)
86) http://ant.apache.org (13.01.2010)
87) http://www.postgresql.org/ (13.01.2010)

DSpace und Tomcat können dann beispielsweise direkt in dessen `home`-Verzeichnis entpackt und installiert werden.

Der Übersichtlichkeit wegen, kann (oder sollte) man auch das vom Betriebssystem für die Installation zusätzlicher Software vorgesehene Verzeichnis `/opt` in Betracht ziehen. In diesem Fall muss dort ein Unterverzeichnis mit allen Berechtigungen für den Nutzer `DSpace` erstellt werden. Außerdem muss für ihn eine Datenbank `DSpace` auf dem PostgreSQL-Server angelegt werden.

Installation

Bevor die Installationsroutine gestartet wird, müssen für den Nutzer `DSpace` einigeUmgebungsvariablen gesetzt werden (`JAVA_HOME`, `JAVA_BIN`, `PATH`, `JAVA_OPTS`, `CATALINA_HOME`, `DSPACE_SRC`). Der Einfachheit halber lassen sich die notwendigen Befehle gut in der Datei `~/.profile` oder einem anderen Skript unterbringen, das bei jedem Login als `DSpace` ausgeführt wird.

Für den Fortgang der Installation müssen einige grundlegende Informationen in die zentrale Konfigurationsdatei eingetragen werden.[88] In der Datei selbst finden sich mit Beispielen versehene Beschreibungen. Es folgt die Erstellung eines Installationspaketes mit Hilfe eines Maven2-Skripts, in dessen Verlauf weitere notwendige Software-Komponenten heruntergeladen werden. Die nächsten Schritte des Installationsprozesses werden von einem Ant-Skript geleistet. Ein schlichtes `ant fresh_install` bewirkt, dass eine Grundinstallation ausgeführt wird. Die abschließende Ausgabe gibt über die verbleibenden Schritte Auskunft, bevor ein Zugang mittels Browser über die ebenfalls angegebenen URLs möglich ist.

Das Einrichten eines funktionsfähigen DSpace-Servers geht dank einer in Sachen Installation und Konfiguration recht ausführlichen Dokumentation schnell von der Hand. Bei der Installation für diese Evaluation sind uns keinerlei Probleme aufgefallen. Administratoren mit grundlegendem Wissen über alle beteiligten Komponenten sollten eine Grundinstallation innerhalb weniger Stunden durchgeführt haben. Der Erfolg lässt sich anschließend sofort durch Aufrufen und Ausprobieren der JSP- bzw. XML-basierten Nutzerschnittstellen überprüfen. Außerdem lohnt wie immersich ein Blick auf die Liste installierter Applikationen des Tomcat-Managers.

3.1.2. EPrints

EPrints stellt auf seiner Homepage vier verschiedene Möglichkeiten zur Auswahl, seine Software in Augenschein zu nehmen.[89] Die Live-CD und eine Windows-Version sind für einen ersten interessierten Blick auf die Fähigkeiten der Software sicherlich brauchbar, nicht jedoch für den produktiven Server-Betrieb. Gerade für den Betrieb auf Linux-Servern sind die verbleibenden zwei Angebote, nämlich fertige Software-Pakete für die Installation von EPrints, zu begrüßen, da man damit beispielsweise Abhängigkeiten von anderer Software meist unproblematisch auflösen und diese ggf. automatisch nachinstallieren lassen kann. Auch Konfigurationsschritte wie das Anlegen eines EPrints-Benutzers mit notwendigen Verzeichnissen, zugehörigen Berechtigungen usw. können hiermit schnell und komfortabel erledigt werden. Leider wird der

88) `${DSPACE_SRC}/dspace/config/dspace.cfg`
89) http://www.eprints.org/software/ (13.10.2010)

Redhat- bzw. Fedora-Nutzer gleich nach dem Klick auf den entsprechenden Link leicht vor den Kopf gestoßen, da dort nicht die erwarteten Installationspakete zu finden sind, sondern als notwendiger Zwischenschritt eine Anleitung, nach der die versprochenen Pakete erst erzeugt werden müssen.

Installation

Für die Installation auf einer Debian- oder einer damit verwandten Linux-Distribution versteckt sich hinter dem zugehörigen Link eine knappe, aber brauchbare Anleitung. Eventuelle Abhängigkeiten werden dabei vom Paket-Verwalter aufgelöst und ggf. nachinstalliert. Mit Hilfe von `apt-get` ist EPrints auch lokal kompiliert installierbar. Um die Vergleichbarkeit mit anderen Repositorien-Systemen nicht zu gefährden, indem eventuelle Besonderheiten der verwendeten Plattform Einfluss nehmen könnten, haben wir hier von dieser Möglichkeit abgesehen; bei der Installation auf einem Produktionsserver sollte man dies jedoch in Betracht ziehen, um systemspezifische Möglichkeiten oder Besonderheiten zu berücksichtigen.

Möchte man an den Quellcode der EPrints-Software kommen, sucht man auf der Website des Projektes vergebens nach einer direkten Verknüpfung. Dennoch stehen laut Dokumentation mehrere Wege zur Verfügung. Zwei Fliegen mit einer Klappe schlägt man durch eine Installation auf Basis der Quelltexte. Danach sollte eine Datei mit der Erweiterung `.tar.gz` und den Quellen als Inhalt im aktuellen Verzeichnis zu finden sein.[90]

Als RDBMS werden MySQL und Oracle unterstützt. Es ist keine weitere Vorbereitung nötig. Im Verlauf der Einrichtung eines Archivs müssen lediglich die üblichen Zugangsinformationen gegeben werden, woraufhin das Programm alle DB-bezogenen Schritte übernimmt.

3.1.3. Fedora Commons

Fedora Commons bietet seine Software in zwei verschiedenen Paketen zum Herunterladen an, einmal als Installer und einmal als Quellcode. Aus den bereits genannten Gründen der Vergleichbarkeit soll auch hier die Installer-Variante genutzt werden. Ihr Download erledigt sich leicht mit `wget`.[91] Für die Überprüfung der Integrität der heruntergeladenen Datei steht eine Prüfsumme (MD5) zur Verfügung.

Voraussetzungen

Fedora Commons benötigt das Java SE Development Kit (JDK) 5 oder 6; auf dem Zielsystem für die Evaluation ist eine Java Laufzeitumgebung in der Version 1.6.0_18 (= JDK 6) vorhanden.

Als RDBMS kommen das mitgelieferte Derby SQL Database 10.4.2 oder eines von MySQL, Oracle oder PostgreSQL infrage. Derby wird auf der Homepage von Fedora Commons nur für die Verwendung zu Evaluations- und Entwicklungszwecken empfohlen. Es wird ausdrücklich davon abgeraten, es für Produktions-Repositorien einzusetzen. Da aber unserer Meinung nach eine Evaluation gerade den Zweck

90) Sollte man erst nach einer Installation auf die Idee gekommen sein, einen Blick in den Quellcode werfen zu wollen, hilft das Kommando `apt-get -d source eprints`. Mit diesem Befehl weist man `apt-get` an, das Source-Paket von EPrints zu holen, ohne jedoch eine Installation durchzuführen. Auch in diesem Fall befindet sich anschließend ein Archiv mit den Quelltexten im aktuellen Verzeichnis. Es sind keine root-Rechte erforderlich, um diesen Befehl ausführen zu können.

91) `wget -c http://downloads.sourceforge.net/fedora-commons/fedora-installer-3.2.1.jar`

verfolgen sollte, die Tauglichkeit eines Systems und seiner Komponenten für den Produktionsbetrieb zu ergründen, setzen wir uns im weiteren über diesen Hinweis hinweg und nutzen für die Evaluation einen bereits auf dem Zielsystem existierenden MySQL Server in der Version 5.0.67.

Als Application Server können wiederum der mitgelieferte Apache Tomcat 5.5.26 oder (laut Installationsanleitung) auch einer von den ebenfalls frei verfügbaren Application Servern Jetty oder JBoss eingesetzt werden. Für die Evaluation setzen wir den von Fedora Commons bevorzugten Apache Tomcat ein, allerdings in der aktuellen Version 5.5.28.

Als Build-Tool wird laut Installationsanleitung Apache Ant in der Version 1.7 oder höher vorausgesetzt. Ob das auch bei einer Installation über das Installer-Paket notwendig ist, wurde nicht überprüft.

Installation

Bevor die Installationsroutine gestartet wird, müssen einige Umgebungsvariablen gesetzt werden (`JAVA_BIN`, `JAVA_OPTS`, `PATH`, `CATALINA_HOME`, `FEDORA_HOME`, `FEDORA_BIN`). Die notwendigen Befehle lassen sich am günstigsten in der Datei `~/.profile` oder in einem anderen Skript unterbringen, das bei jedem Login ausgeführt wird.

Das bereits heruntergeladene Installationsprogramm kann direkt aus der Kommandozeile ausgeführt werden (`java -jar fedora-installer-3.2.1.jar`). Die Eingabe der Werte aller Parameter, die für die Konfiguration notwendig sind, erfolgt textbasiert und nacheinander, und ist einigermaßen übersichtlich in Abschnitte gegliedert. Im Folgenden erwähnen wir nur zwei Werte, die von den Vorgabe des Installationsprogramms abweichen: die Verfügbarkeit über SSL, die Servlet Engine und die Datenbank (hier werden die bereits existierenden Apache Tomcat und MySQL benutzt). Erwähnenswert ist an dieser Stelle der Fedora Ressource Index, der auf einer Mulgara-Datenbank[92] basiert und die Indizierung von Beziehungen zwischen Objekten und ihren Komponenten ermöglicht, was für die Problemstellung eines Personendaten-Repositoriums sehr interessant ist, und was auch als möglicher Vorteil gegenüber anderen Produkten bewertet werden kann.[93]

Um einen leichteren Einblick in die Möglichkeiten von Fedora zu bieten, stellt das Projekt einige Daten zur Verfügung, die mittels Skript in das Repositorium eingepflegt werden können. Das Vorgehen hierfür ist gut dokumentiert. Die Demo-Objekte stehen anschließend für den direkten Zugriff oder für Suchanfragen zur Verfügung.

Fedoras Installationsprozess ist sehr geradlinig gestaltet. Administratoren mit grundlegenden Kenntnissen der genutzten Plattform, des Java Application Servers, verteilter Systeme und des gewählten RDBMS sind problemlos in der Lage, die einzelnen Komponenten auch abweichend von der Standardkonfiguration zu einem funktionierenden Ganzen zusammenzufügen. Installation und Konfigurationsmöglichkeiten sind Teil

92) http://www.mulgara.org (13.01.2010)

93) Hat man, der Vorgabe entsprechend, Fedoras XACML policy enforcement aktiviert, muss man nach Abschluss der Installationsroutine eine kleine Änderung in der Datei `${FEDORA_HOME}/data/fedora-xacml-policies/repository-policies/default/deny-apim-if-not-localhost.xml` vornehmen, um den entfernten Zugriff auf die API-M zu ermöglichen. Die Zeile `<Rule RuleId="1" Effect="Deny">` ist hierfür einfach in `<Rule RuleId="1" Effect="Permit">` abzuändern.

der durchaus als umfangreich zu bezeichnenden Dokumentation des Fedora-Projekts. Fehlerhafte oder unzureichende Informationen sind im Rahmen dieser Evaluation nicht aufgefallen. Nach Abschluss einer Installation lässt sich der Erfolg sehr schnell mit Hilfe einiger beiliegender Beispieldaten überprüfen. Das für deren Import auszuführende Skript zeigt seinerseits, wie man Daten ggf. aus dem lokalen Dateisystem in das Repositorium einpflegen kann. Weitere Applikationen und Schnittstellen stehen sofort nach der Installation zur Verfügung und erlauben einen Blick ins Repositorium und seinen Datenbestand.

3.1.4. Greenstone

Für Greenstone stehen drei Installationspakete für den Betrieb unter Windows, MacOS und Unix/Linux zur Verfügung. In dieser Evaluation haben wir den letzten Stable Release 2.82 unter Windows installiert.

Voraussetzungen

Dank des Installationsprogramms waren keine Voreinstellungen wie Vergabe von Systemvariablen erforderlich.

Installation

Das Installationspaket wurde heruntergeladen und auf einer Workstation ausgeführt. Im ersten Dialog des Installationsprogramms kann die Installationssprache ausgewählt werden, zur Auswahl stehen Englisch, Französisch, Spanisch, Deutsch, Russisch, Chinesisch und Arabisch. Durch die folgenden Installationsschritte wie Lizenzzustimmung und Installationsverzeichnis leitet das Installer-Programm.

Die Greenstone-Applikation vereint alle für das Repositorium erforderlichen Dienste wie auch die eines Servers, so dass kein eigenständiger Server installiert wird. Nach Abschluss der Installation kann unter Windows das Repositorium direkt vom Startmenü aus gestartet werden. Der Server stellt sämtliche Funktionalitäten standardmäßig unter http://localhost:1025 zur Verfügung. Für einige Server-Einstellungen steht eine minimale Benutzeroberfläche zur Verfügung. Diese Oberfläche ist eine Besonderheit des Greenstone-Repositoriums, die die lokale Installation erleichtert. Beim Betrieb auf einem Server ohne grafische Benutzeroberfläche scheint das Kommandozeilen-Interface zu fehlen.

Zudem wurden vom Installationsprogramm das Graphical Librarian Interface (GLI) zum Bearbeiten und Verwalten des Repositoriums sowie der Metadata-Set Editor zum Erstellen neuer Metadaten-Schemata installiert.

3.2. BENUTZERVERWALTUNG

In der Regel besitzen Repositorien-Systeme ein Benutzerverwaltungssystem, welches bereits passend zugeschnittene Benutzerrollen anbietet. Wir untersuchen, inwieweit die angebotene Benutzerverwaltung den vorliegenden Zwecken gerecht wird und inwieweit das System Möglichkeiten zur Anpassung bietet bzw. ob Implementierungsaufwand einzuplanen ist.

3.2.1. DSpace

DSpace bietet eine sehr weitreichende und praxistaugliche Benutzerverwaltung. Das Repositorium ist in `Communities`, `Sub-Communities` und `Collections` organisiert, für deren Benutzer und Reviewer man jeweils separate Berechtigungen zuweisen kann. Dadurch ist das System besonders gut einsatzfähig für die Handhabung verschiedenartiger Datenbestände, die von ganz unterschiedlichen Benutzergruppen bearbeitet werden. Diese Einteilung kann einen Nachteil bedeuten, da sie sowohl auf einen institutionellen Rahmen als auch auf bestimmte Objekttypen ausgerichtet sind. Der Workflow fließt erst auf der untergeordneten Ebene mit ein.

3.2.2. EPrints

EPrints bietet, ähnlich wie DSpace, eine Benutzerverwaltung, die bereits für die Nutzergruppen eines Dokumenten-Repositoriums ausgelegt ist. Es unterscheidet zwischen den Gruppen `Admin`, `Editor` und `User`: Ein `Admin` kann neue Benutzer anlegen, ein `User` kann Daten und Dokumente ins Repositorium eingeben, welche einem Review durch einen `Editor` unterliegen, bevor sie öffentlich einsehbar werden. Diese Benutzergruppen entsprechen vom Prinzip her genau den notwendigen Anforderungen.[94] Allerdings könnte sich dies in anderen Anwendungsfällen auch als Nachteil offenbaren. Hier liegt im Prinzip der umgekehrte Fall zu DSpace vor.

3.2.3. Fedora Commons

Da Fedora keine Endbenutzersoftware anbietet, bildet es unter den Repositorien-Systemen eine Ausnahme. Es wird daher auch nicht zwischen verschiedenen Benutzergruppen unterschieden, sondern es gibt lediglich Administratoren. Benutzergruppen müssen also in einer auf dem Repositorium aufbauenden Anwendung implementiert werden. Dies ist ein Nachteil des Systems, da die Entwicklung eines Benutzer- und Rechteverwaltungssystems nicht trivial ist. Ob es separate Module gibt, die eine Benutzerverwaltung implementieren, wurde nicht recherchiert.

3.2.4. Greenstone

In Greenstone können Datenobjekte und komplette Collections nur mittels des Graphical Librarian Interface angelegt bzw. hinzugefügt werden. Es gibt somit keine weitere Differenzierung zwischen Administratoren und Endbenutzern. Es fehlt z.B. eine deutlich umrissene Gruppe von Editoren, die neue Datenobjekte anlegen und diese in das Repositorium einspielen können, ohne dass damit Rechte einhergehen, welche Veränderungen an den Grundeinstellungen des Systems erlauben würden. Hilfestellung für die Benutzer – etwa durch einen anpassbaren Workflow – ist ebenfalls im System nicht angelegt. Die Rechteverwaltung ist in Greenstone daher als eingeschränkt zu beurteilen.

---

94) Ein Nachteil hat sich beim Anlegen neuer Benutzer gezeigt. Benutzer können angelegt und gespeichert werden, ohne dass ihnen ein Passwort zugewiesen wurde. Dadurch entstehen Benutzer, die wegen fehlendem Passwort nicht aufgerufen werden können. Ein Passwort kann später nicht zugewiesen werden. Nicht getestet wurde, ob Benutzern ohne Passwort automatisch das Admin-Passwort zugewiesen wird, unter dem sie erstellt wurden.

3.3. CUSTOMIZING

Dieser Abschnitt beziehen sich auf die individuelle Anpassbarkeit der Repositoriensoftware. Aufgrund der Verschiedenartigkeit der Repositoriensysteme sind die Ausführungen nur schwer miteinander vergleichbar; allerdings konnten wir allgemeine Beobachtungen dazu festhalten, inwieweit und an welchen Stellen das jeweilige System eine spezialisierte Konfiguration wie z.B. bei unserem Vorhaben unterstützt oder erschwert.

3.3.1. DSpace

DSpace bietet eine Reihe von effizienten Möglichkeiten, die Eingabemasken für Metadaten im Webfrontend anzupassen. Hier können Workflows für alle oder für einzelne Collections angelegt werden und aus den mitgelieferten Eingabeseiten (Describe, Verify, Licence, Upload usw.) beliebige ausgewählt und in beliebiger Reihenfolge hintereinander geschaltet werden. Für Metadaten, wie wir sie im PDR anlegen werden, genügen die Möglichkeiten der Inputseite „Describe“. Der Umfang und die Auswahl von Eingabefeldern auf einer oder mehreren „Describe“-Seiten kann auf der Grundlage der importierten Metadaten-Schemata beliebig definiert werden. Icons, Style-Sheets, Header und Footer können ebenfalls leicht angepasst werden.

Umfangreichere Anpassungen müssen allerdings in den JSP-Seiten vorgenommen werden, was mit deutlichem Mehraufwand verbunden wäre und damit außer Frage steht. Für Entwickler ist jedoch bereits vorgesehen, DSpace in Eclipse als Projekt zu importieren, damit es dort umfassend bearbeitet werden kann. Hierzu finden sich Dokumentationen über die DSpace-Webseiten.[95] Die Möglichkeit, DSpace in Eclipse zu bearbeiten, erweitert seine Anpassbarkeit deutlich, was als besonderer Vorteil zu werten ist.

3.3.2. EPrints

Erweiterungen können als Plugins geschrieben und implementiert werden. Laut Dokumentation ist dies besonders dazu geeignet, Daten für den Export für die Darstellung in GoogleEarth oder TimeLine vorzubereiten oder sie aus den genannten Formaten zu importieren.

Änderungen an der Konfiguration der Datenbank, der Menge der Datentypen, den Feldern, den Workflows, den Feldverknüpfungen, den Citations sowie den Phrasen können auf zwei Ebenen vorgenommen werden. Zunächst ist dies auf der Ebene der Konfigurationsdateien möglich, was im EPrints-Wiki dokumentiert ist.[96] Beim Test kam es jedoch häufig zu schweren Fehlern und Problemen, da die Dokumentationen entweder nicht ausreichend oder fehlerhaft waren. Leichter und sicherer waren Konfigurationsänderungen auf der Admininistrator-Ebene zu handhaben. Ein Nachteil ergab sich dennoch, denn die Änderungen wurden direkt in der Datenbank umgesetzt, ohne dass sie z.B. in einer Konfigurationsdatei dokumentiert werden. So sind Anpassungen nicht mehr auf einen Blick sichtbar bzw. werden dadurch nicht mehr genau nachvollziehbar.

Als deutlicher Mangel hat sich die eingeschränkte Gestaltungsmacht mittels der Konfigurationsdateien herausgestellt. Die default.xml des Workflows[97] führt beispielsweise nicht alle Felder eines Datentyps auf, die

95) http://wiki.dspace.org/index.php/IDE_Integration:_DSpace,_Eclipse_and_Tomcat
96) How-tos für die Konfiguration finden sich hier: http://wiki.eprints.org/w/Category:Howto
97) `/eprints3/archives/test/cfg/workflows/eprint`

unter „Details“ angezeigt werden. Das bedeutet, dass vorgegebene Felder wie z.B. „Title“, „Abstract“ oder „Creator“ nicht ausgeblendet werden können. Auf der anderen Seite kann unter dem Reiter „Details“ lediglich eine Auswahl der vorgegebenen Eingabefelder angezeigt werden. D.h. benutzerdefinierte Eingabefelder können nicht unter „Details“ eingefügt werden, sondern werden in einem separaten Reiter unter „Misc.“ angezeigt.

Auf Grund dieser Einschränkungen bei der Konfiguration der grafischen Oberfläche und des Workflows kann das Webfrontend von EPrints nicht ausreichend an die Anforderungen des PDR angepasst werden. Ferner führte der Feldtyp „itemref“, welcher der Verknüpfung von Datensätzen des Repositoriums dient und ein wesentliches Funktionselement für das PDR wäre, wiederholt zu Fehlern.

3.3.3. Fedora Commons

Fedora bietet selbst keine Benutzeroberfläche. Lediglich zwei Admin-Oberflächen stehen für die Verwaltung von Daten im Repositorium zur Verfügung. Die ältere Oberfläche ist als Java-Applikation realisiert, mit der Objekte angelegt, gelöscht sowie im- und exportiert werden können. Seit Fedora 3.2 gibt es für die Administration eine browserbasierte Nutzerschnittstelle. Sie ist, ein installiertes Flash-Plugin vorausgesetzt, per Browser zugänglich und bietet ähnliche Funktionalität wie die Java-Applikation. Beide Oberflächen sind jedoch nur für die Verwaltung des Repositoriums durch Administratoren ausgelegt und sind nicht für etwaige eingeschränkt berechtigte Benutzergruppen oder Anpassungen eines Workflows vorgesehen.

3.3.4. Greenstone

Greenstone stellt drei Module bzw. User Interfaces zur Verfügung: Webfrontend, Graphical Librarian Interface (GLI) und Metadata Set Editor.

Das Webfrontend dient ausschließlich der Darstellung von und der Suche inveröffentlichten Collections. Seine Benutzeroberfläche kann durch die Erstellung von Makros, Stylesheets etc. angepasst werden. Das Webfrontend dient weder der Eingabe von Daten und Dokumenten, noch dem Hinzufügen oder Erstellen von Collections.

Das Graphical Librarian Interface (GLI) ist eine Java-Applikation zur Erstellung von Collections, Verwaltung vorhandener Collections, Eingabe von Metadaten und der Einstellung der zu indexierenden Felder. Dies ist das zentrale Eingabewerkzeug von Greenstone. Es ist gut dokumentiert, jedoch nur für die Benutzung durch Administratoren geeignet, da es sehr umfangreiche Einstellungsmöglichkeiten bietet und sich sein Workflow nur geringfügig anpassen lässt.

Der Metadata Set Editor ist eine Besonderheit von Greenstone. Mit diesem lassen sich Metadaten-Schemata (sowohl flache als auch hierarchische) erstellen und in mehreren Sprachen kommentieren. Die so erstellten Metadaten-Schemata können anschließend als Standard-Metadaten-Schema einer Collection in das GLI geladen werden und stehen dort für die Eingabe von Metadaten zur Verfügung.

4. SPEZIFISCHE EVALUATION

4.1. METADATEN-SCHEMA

In diesem Abschnitt beurteilen wir die Flexibilität des Metadaten-Schemas der verschiedenen Repositorien-Systeme. Wichtig ist dabei vor allem die Möglichkeit, ein eigenes hierarchisch strukturiertes Datenmodell einfließen zu lassen.

4.1.1. DSpace

Metadaten-Schema

Metadaten werden in DSpace defaultmäßig nach Dublin Core kodiert. In diesem Schema legt DSpace die Daten in der relationalen Datenbank ab. DSpace bietet weitreichende Optionen, die Metadaten-Eingabemasken auf dem Webfrontend anzupassen. Hier können Workflows für alle oder für einzelne Collections angelegt werden. Es ist möglich, die mitgelieferten Eingabeseiten (Describe, Verify, Licence, Upload etc.) in beliebiger Reihenfolge hintereinanderzuschalten.

Die unter DSpace zur Verfügung stehenden Typen für Eingabefelder genügen den Anforderungen des PDR. Allerdings konnte eine benutzerfreundliche Umsetzung der Objektrelationen – z.B. Verknüpfungen von Aspekten mit Personen – nicht mit den Bordmitteln umgesetzt werden, sondern muss diesem Eindruck nach mit einer eigenen JSP-Seite ausgeführt werden.

Erstellung eines eigenen Metadaten-Schemas

Im folgenden beschreiben wir den Versuch, neue Metadata-Schemata mit benutzerdefinierten Datenfeldern zu erstellen. Für Metadaten, wie wir sie im PDR anlegen werden, genügen die Möglichkeiten der Inputseite „Describe“. Hier können dem DC-Schema benutzerdefinierte Datenfelder hinzugefügt und für die Endanwendung beliebig angeordnet werden. Der Umfang und die Auswahl von Eingabefeldern auf einer oder mehreren „Describe“-Seiten kann auf der Grundlage der importierten Metadaten-Schemata beliebig definiert werden. Alle neu hinzugefügten Datenfelder werden allerdings dem DC-Namensraum zugeordnet, auch wenn sie gar nicht dem DC-Standard entsprechen. Ein Auszug aus den FAQ beschreibt dies wie folgt:

> *In this context support for a given metadata schema means that metadata can be entered into DSpace, stored in the database, indexed appropriately, and made searchable through the public user interface. This currently applies mainly to descriptive metadata, although as standards emerge it could also include technical, rights, preservation, structural, and behavioral metadata. Currently DSpace supports only the Dublin Core metadata element set with a few qualifications conforming to the library application profile (see DSpace Metadata). The DSpace team hopes to support a subset of the IMS/SCORM element set (for describing education material) in the coming year. HP and MIT also have a research project called SIMILE that is investigating how to support arbitrary metadata schemas using RDF as applied by the Haystack research project in the Lab for Computer Science and some of the Semantic Web technologies being developed by the W3C.*[98]

98) http://www.dspace.org/faq/FAQ.html

Bei der Frage nach der Erstellung eines benutzerdefinierten Metadaten-Schemas sind die Dokumentationen nicht eindeutig. Angedeutet wird, dass solche Schemata auf Admin-Ebene im Web Frontend sowie in den Konfigurationsdateien erstellt werden können. Der Versuch, ein eigenes Schema zu implementieren, führte zu wiederholten Fehlermeldungen. Zwar konnten wir ein Schema „pdr“ auf Admin-Ebene erstellen, jedoch konnte es nicht in den Workflow für die Dateneingabe eingebunden werden. Entsprechende Implementierungen in der input-forms.xml – /dspace/config/ – führten wiederholt zu Fehlermeldungen. Der Versuch, dieses Problem durch die Erstellung einer neuen Schema-Datei `pdr-types.xml` nach dem Muster der `dublin-core-types.xml` zu umgehen, schlug ebenfalls fehl, da es beim Import dieser Datei zu einer `ClassNotFoundException` kam. Wie sich herausstellte, ist in der aktuellen DSpace-Distribution die Methode `SchemaImporter`, die, der Dokumentation zufolge, den Import von Metadata-Schemata durchführt, nicht implementiert. Ausgeführt werden konnte nur der `MetadataImporter`, welcher jedoch nur Dublin Core und benutzerdefinierte Erweiterungen des Dublin Core akzeptiert. Für `pdr-types.xml` zeigte `MetadataImporter` keinerlei Effekt.

Demnach unterstützt DSpace bisher nur flachhierarchische Metadaten-Schemata, welche dem Dublin Core entsprechen. Das SMILE-Projekt, eine Initiative von Hewlett Packard und dem MIT, arbeitet derzeit an einer Erweiterung von DSpace für die Unterstützung von weiteren Metadata-Schemata, die auf RDF aufbauen. Es ist jedoch noch nicht absehbar, wann entsprechende Plugins oder neue Releases mit dieser Funktionalität zu erwarten sind.

Bei der Untersuchung von ca. 50 Repositorien von Bibliotheken, Universitäten und Forschungseinrichtungen, die DSpace verwenden, wurde kein Beispiel-Repositorium gefunden, bei dem ein anderes Metadata-Schema als Dublin Core verwendet wird. Nach der bisherigen Evaluation zeigte sich daher, dass die Implementierung von benutzerdefinierten Metadata-Schemata weder gängige Praxis ist noch bisher ohne größere technische Eingriffe in die Geschäftslogik von DSpace möglich ist.

Crosswalks

Im Zusammenhang mit den Problemen, auf die die Erstellung eines eigenen Metadata-Schemas stieß, muss auch auf die Funktion von Crosswalks in DSpace eingegangen werden. Da DSpace intern auf dem Dublin Core-Standard aufbaut, sind die sogenannten Crosswalks dazu gedacht, die Transformation von Datensätzen in andere Formate wie z.B. MODS zu automatisieren. Solche Crosswalks basieren auf XSLT-Skripten und können in den OAI-Support integriert werden. In der aktuellen Release enthält DSpace Crosswalks für MODS, METS und QDC (Qualified Dublin Core). Dadurch können ohne großen Aufwand Transformationen zwischen dem internen Metadata-Schema DC und externen Schemata durchgeführt werden, die für die OAI und für die SOAP-Schnittstelle zur Verfügung stehen.[99] Für die Anforderungen des PDR ergäbe sich dadurch die Möglichkeit, das interne DC-Schema an das PDR-Datenformat anzupassen und entsprechende Crosswalks so zu implementieren, dass die PDR-Daten im PDR-Datenformat über die SOAP-Schnittstelle für Suchanfragen bereitgestellt werden können.

99) Näheres zu den Schnittstellen im Abschnitt 4.4.

Bei diesem Ansatz ergaben sich jedoch zunächst zwei Probleme: Die Anpassung des internen Schemas ergibt zwangsläufig eine DC-Kodierung. Dem DC-Namespace werden dadurch Elemente hinzugefügt, die nicht zum Dublin Core gehören. Somit werden DC-fremde Elemente unter DC einsortiert. Des weiteren erfordern Crosswalks auf der Grundlage von XSLT zusätzliche Rechenkapazität, und zwar nicht nur bei jedem Import von neuen Daten, sondern bei jeder Suchanfrage. Da das PDR auf sehr große Datenmengen ausgerichtet ist, ergäbe dieser Workaround aufgrund des zusätzlichen Transformationsaufwandes sehr wahrscheinlich eine deutliche Performance-Einbuße des Gesamtsystems.

Controlled Vocabulary

Diese Funktion ermöglicht die Erstellung einer hierarchischen Liste, aus welcher der Benutzer Begriffe auswählen kann, um sie in ein bestimmtes Feld eintragen zu können. So können z.B. Klassifikationen hierarchisch zusammengestellt werden, aus denen der Benutzer einzelne auswählen kann, um sie einzelnen Aspekten zuzuordnen. Die Funktion eignet sich ggf. auch für das Taggen von Aspekten, wie sie im PDR verwendet werden sollen. Wichtig ist, dass in diesem Fall dem Nutzer erlaubt werden muss, selbst eigenen Text in das Eingabefeld einzugeben. Hierzu muss in der Definition des entsprechenden Datenfeldes in der input-forms.xml das Attribut `closed` des Elements `vocabulary` den Wert `false` haben (`<vocabulary closed="false">`). Dann kann der Nutzer sich entweder an den vorhandenen Klassifikationen orientieren oder dem Vokabular eine neue Klassifikationen hinzufügen.

4.1.2. EPrints

Über die Admin-Seite des Web Frontends können Metadatensätze in EPrints verwaltet werden. Hier können neue Datenfelder erstellt, deren Anzeigename bestimmt und ihr Datentyp (Date, String, Text etc.) definiert werden. Benutzerdefinierte Datenfelder können anschließend in den Workflow integriert werden. Dadurch lässt sich das Metadaten-Schema in EPrints weitgehend an die verschiedenen Bedürfnisse von Dokumenten-Repositorien anpassen. Die Konfiguration bzw. das Hinzufügen neuer Datenfelder wird jedoch direkt in der relationalen Datenbank des Repositoriums ausgeführt. Dies macht das Löschen und anschließende Rekompilieren des Repositoriums erforderlich. Neue Datenfelder können also nicht oder nur sehr aufwendig im laufenden Betrieb erstellt werden.

4.1.3. Fedora Commons

In Fedora werden Metadaten zu einem Dokument oder Datenobjekt in einer XML-Datei je Datenobjekt im Fedora-Format FOXML abgelegt. Dabei wird zwischen obligatorischen, systemeigenen Metadaten eines Objekts (persistent identifier, event history, etc.) und den Metadaten eines Objektes, welche dieses beschreiben und die selbst wie Content bzw. Datastream behandelt werden, unterschieden. Zu den deskriptiven Metadaten in Fedora zählen Daten im Format DC, der AUDIT-Trail, der die Änderungen verzeichnet, sowie der RELS-EXT-Datastream, in dem Verknüpfungen mit anderen Objekten nach RDF beschrieben werden. Die Daten des DC enthalten in Fedora mindestens die obligatorischen Einträge dc:creator und dc:title, die sich nicht deaktivieren lassen.

Weitere benutzerdefinierte Metadaten können innerhalb der FOXML-Metadaten in Fedora abgelegt werden. Da die benutzerdefinierten Metadaten als Datastream behandelt werden, bestehen keine Einschränkungen bei der Definition eigener Metadaten, die als valide XML-Daten in der FOXML-Datei eingefügt werden. Auf diese Weise lassen sich hier die Daten einzelner Datenobjekte des PDR verwalten. Fedora bietet die größte Flexibilität im Umgang mit benutzerdefinierten Metadaten im Vergleich mit den anderen evaluierten Repositorien. Anzumerken ist, dass Fedora die Metadaten jedoch nicht selbst indiziert. Eine Suche ist daher nicht ohne eine eigene Implementierung möglich.

4.1.4. Greenstone

Wie bereits erwähnt, bietet Greenstone die Besonderheit eines Metadata-Set-Editors zur Erstellung eigener Metadata-Schemata. Der Editor ist als Java-Applikation realisiert und gut dokumentiert. Durch das klare und übersichtliche Menü ist es leicht, ein flaches oder hierarchisch strukturiertes Schema zu erstellen und die einzelnen Datenfelder mit Definitionen und Kommentaren zu versehen. Anschließend kann das Metadata-Schema problemlos im GLI importiert werden. Dadurch wird in Greenstone die Unabhängigkeit von Metadaten-Schemata wie Dublin Core oder METS ermöglicht. Hervorzuheben ist hier wieder die internationale Ausrichtung von Greenstone, denn Kommentare und Definitionen können in über 100 verschiedenen Sprachen angelegt werden.

Auf Schwierigkeiten stieß jedoch der Versuch, Metadaten unabhängig von Dokumenten bzw. Dateien abzuspeichern. Greenstone ist wie jede Repositorien-Software auf Verwaltung, Speicherung und Bereithaltung von Dateien jeder Art ausgerichtet (Dokumente, Bilder, Audio, etc.). Anders als bei den anderen evaluierten Repositorien-Systemen fällt der zu verwaltenden Datei in Greenstone jedoch eine zentrale Rolle bei der Erstellung von neuen Objekten zu. Im GLI werden Objekte in einer *Collection* durch das Hinzufügen von Dateien erstellt. Erst nach dem Hinzufügen können zu dieser Datei Metadaten eingetragen werden. Es ist daher nicht möglich, Metadaten ohne Datei zu verwalten, was die Verwendung als reines Datenrepositorium – wie vom PDR angestrebt – deutlich erschwert.

4.2. DATENMANAGEMENT

4.2.1. DSpace

Deaktivierung der Aufforderung, eine Datei zu einem Eintrag hinzuzufügen, lässt sich leicht über den Workflow bzw. die submission-processes steuern.

Verknüpfung von Datensätzen

Für benutzerdefinierte Datenfelder stellt DSpace keinen Datentyp zur Verfügung, der Verknüpfungen mit anderen Daten des Repositoriums vorsieht. Verknüpfungen werden daher nicht direkt über die DSpace-Eingabemaske unterstützt, sie können nur manuell als Zitate von verknüpften Identifizierern von Datensätzen eingetragen werden.

4.2.2. EPrints

Die von EPrints vorgegebenen Datentypen können durch Eingriffe in die Konfigurationsdateien in eingeschränktem Rahmen modifiziert und ausgeblendet werden. Auch neue Datentypen können erstellt werden. Dabei gilt jedoch die Restriktion, dass alle Datentypen als Metadaten von Dateien behandelt werden, zu denen jeweils ein Feld für den Verweis auf eine Datei (Dokument etc.) vorgesehen ist, das nicht ausgeblendet werden kann.

Für die Klassifikation von Dokumenten nach Themengebieten unterstützt EPrints sog. Subject-Listen, wie sie z.B. von der Library of Congress erstellt wurden. Für die Anforderungen des PDR und zur Klassifikation der Aspekte kann die *Subjects*-Liste und deren Hierarchie entsprechend angepasst werden.[100] Die Anpassung und Erweiterung der Liste kann auf Admin-Ebene als auch in den Konfigurationsdateien vorgenommen werden.

In der Konfiguration können auch Labels und Erläuterungen zu Feldern frei angepasst, neue Datenfelder erstellt, benannt und in einen Workflow integriert werden.[101] Das grundsätzliche Problem, dass für die Aktivierung der Feldänderungen das gesamte Repositorium zunächst gelöscht werden muss, konnte dabei nicht umgangen werden. Dieser Umstand macht Feldänderungen im produktiven Betrieb im Prinzip unmöglich, es sei denn, man arbeitet mithilfe von SQL-Befehlen direkt an der Datenbank.

EPrints verfügt außerdem über Versionierung, welche neue Versionen eines Objektes auf der Grundlage einer exakten Kopie des bisherigen Objektes anlegt. Damit kann leicht eine Änderungshistorie sichtbar gemacht werden.

Strukturierung und Verknüpfung von Datensätzen

Für die Verknüpfung von Datensätzen bietet EPrints den Datentyp „itemref“ an. Bei der Evaluation führte die Implementierung dieses Datentyps jedoch zu wiederholten, nicht näher identifizierbaren Fehlermeldungen, wenn versucht wurde, ein neues Datenobjekt mit diesem Datentyp zu erstellen. Für die Strukturierung und Ordnung der Daten bietet EPrints zusätzlich die Option, Dokumente nach Fakultäten und Abteilungen von Universitäten zu sortieren. Diese Strukturierung ist für den universitären Betrieb ausgelegt und erleichtert die hierfür nötigen Anpassungen. Für den Betrieb im PDR wäre jedoch eine freie Sortierung von Daten in Collections geeigneter.

---

100) Beispiel für eine hierarchische Liste des PDR:
Person
├ Geburt
└ Tod
Beruf etc.

101) Der Versuch, ein neues Feld durch Modifikationen in den Konfigurationsdateien zu erstellen und in den Workflow zu integrieren, meldete das System den Fehler „Failed to retrieve item with ID 1“, sobald ein neues Item angelegt werden sollte. Es stellte sich heraus, dass die Syntax für die Datei eprint_field.pl, welche im Tutorial benutzt wird, nicht mit der tatsächlich verwendeten Syntax übereinstimmt. Offenbar ist das Tutorial in diesem Punkt nicht auf dem neuesten Stand.

4.2.3. Fedora Commons

Fedora legt Metadaten in seinem internen FOXML-Format ab. Zwar ist das System darauf ausgelegt, verschiedene Formate für diese Aufgabe zu unterstützen, aber laut Dokumentation ist dies gegenwärtig nur mit den Formaten FOXML 1.0/1.1, METS 1.0/1.1 und ATOM 1.1 und ATOM Zip 1.1[102] möglich. Eigene, benutzerdefinierte Metadaten-Felder, die nicht zu Dublin Core gehören, können als separater Datastream eingebunden und innerhalb der FOXML-Metadaten abgelegt werden. Sie können dann über einen zusätzlichen Suchdienst wie beispielsweise die Fedora Generic Search mittels Lucene, Solr[103] o.Ä. durchsuchbar gemacht werden.[104]

Datenmodell

Inhalte werden in FEDORA in Form von Objekten abgelegt. Es gibt grundsätzlich 4 verschiedene Objekttypen. Für Content sind das Cmodel Objekt und das Data Objekt relevant. Ein Objekt besteht aus Streams. DC, AUDIT RELS-EXT sind vordefiniert. Der Aufbau eines DataObjects wird durch das Cmodel beschrieben, der MIMEType der Streams ist völlig beliebig. Jedweder Inhalt ist integrierbar und durch jedwede Form von Metadaten beschreibbar. Das Content Model muss für jedes Anwendungsszenario selber entwickelt werden, was sich in Abhängigkeit von seiner Komplexität als durchaus aufwändig erweisen kann.

ServiceDefinitions und ServiceDependencies beschreiben Verarbeitungsanwendungen als Objekt. Durch diese Eigenkonfigurierbarkeit kann potenziell jeder Datentyp vor einer Auslieferung auf jede Art und Weise verarbeitet werden. Das ermöglicht die Bildung von Schnittmengen und den gezielten Zugriff auf Teile eines Datenobjekts.

Indexiert werden standardmäßig im Resource-Index nur die RDF-Triples aus dem RELS-EXT-Stream und im Search-Index nur der DC-Stream. Außerdem kann die Indexierung und das Durchsuchen durch einen eigenen Service von eigenen Metadata-Streams realisiert werden.

Strukturierung und Verknüpfung von Datensätzen

In Fedora können Objekte erstellt werden, die eine Collection repräsentieren. Zugehörigkeit zu einer solchen, durch ein Objekt repräsentierten Collection, kann mittels eines RDF-Triples ausgedrückt werden. Dadurch können Inhalte auf verschiedene Art und Weise gruppiert werden.

Die Verknüpfung von Objekten spielte bei der Entwicklung von Fedora eine wichtige Rolle. Sie wird durch Fedora Digital Object Relationships realisiert.[105] Die Formate dieser Relationships werden auf der Grundlage

---

102) http://www.fedora-commons.org/confluence/display/FCR30/Introduction+to+FOXML (25.01.2010)
103) http://lucene.apache.org/solr/ (25.01.2010)
104) Dazu folgende Foreneinträge:
http://www.fedora-commons.org/confluence/display/DEV/mail/8749992 (25.01.2010)
http://www.fedora-commons.org/confluence/display/DEV/mail/11504859 (25.01.2010)
http://www.fedora-commons.org/confluence/display/FCKB/mail/8751652 (25.01.2010)
http://www.fedora-commons.org/confluence/display/FEDORACREATE/XML+Metadata+Editing (25.01.2010)
105) http://fedora-commons.org/confluence/display/FCR30/Digital+Object+Relationships (25.01.2010)

von RDF definiert[106]. Jede Relationship besteht aus zwei Objekten und einer qualitativ definierten Beziehung zwischen diesen beiden, also aus einem Tripel von subject, property und object. Derartige Verknüpfungen werden in Fedora sowohl für die Gruppierung von Datensätzen als auch für die Klassifikation von Datensätzen verwendet. Die Daten über die Verknüpfungen eines Objektes werden im RELS-EXT-Datastream des Objektes gespeichert.[107] Die Digital Object Relationships machen den Eindruck eines mächtigen und gut entwickelten Werkzeugs für Strukturierung, Klassifikation, Verknüpfung, Rechte-Administration und weitere Aufgaben.

4.2.4. Greenstone

Grundsätzlich wird für jedes Objekt in Greenstone eine Datei angelegt. Um nun reine Datenobjekte ohne Dateien zu erstellen, können bzw. müssen leere Dummy-Dateien mit dem Suffix .nul erzeugt und eingefügt werden. Diese können anschließend Träger von Metadaten werden.[108] Die eingegebenen Metadaten werden in einer XML-Datei in einem Greenstone-internen Schema abgespeichert. Zu jedem Datenobjekt wird eine XML-Datei angelegt, die neben den Metadaten auch den Text eines Werkes, der im Repositorium zugänglich gemacht und angezeigt wird, enthält. Für die Metadatenfelder stellt Greenstone keine nähere Spezifizierung von Datentypen zur Verfügung. Im GLI können Daten verschiedener Datentypen in ein Metadaten-Feld eingegeben werden, was hohe Flexibilität bietet, jedoch eine unterstützende Fehlervermeidung durch Prüfung der Syntax bei der Eingabe vermissen lässt.

Strukturierung und Verknüpfung von Datensätzen

Greenstone sortiert Datenobjekte in Collections, bietet darüber hinaus aber keine besonderen Funktionen zur Verknüpfung von einzelnen Datenobjekten.

4.3. SUCHFUNKTION

In diesem Abschnitt untersuchen wir die Suchmechanismen der verschiedenen Repositorien-Systeme, einschließlich der verwendeten Indexer, der Schlagwortlisten und der Volltextsuche.

4.3.1. DSpace

In DSpace kann zunächst anhand vorgegebener Kategorien (wie z.B. Autor, Titel, Jahr oder Thema) in den Datensätzen geblättert werden. Außerdem kann durch eine freie Suche auf alle Datenfelder – auch auf benutzerdefinierte – zugegriffen werden. Dies ist allerdings nur für Metadaten möglich, für die Dokumente selbst stellt DSpace keine Volltextsuche zur Verfügung. Diese kann jedoch durch die Implementierung von

106) Das Modell findet sich unter http://www.fedora-commons.org/definitions/1/0/fedora-relsext-ontology.rdfs (25.01.2010)
107) http://fedora-commons.org/confluence/display/FCR30/Digital+Object+Relationships (25.01.2010)
108) Die Antwort auf eine entsprechende Frage auf der Mailingliste bestätigt dies: http://tinyurl.com/25qurnf (22.04.2010)

Lucene[109] geschaffen werden. Es besteht außerdem die Möglichkeit, die Suchanfragen auf bestimmte Communities oder Collections zu beschränken.

4.3.2. EPrints

EPrints bietet allgemeine Suchfunktionen für das Suchen in den Metadaten, sowie benutzerfreundliche Blätterfunktionen für Autor, Datum, Abteilung und Subject. Eine Volltextsuche für Textdokumente wie z.B. PDF wird ebenfalls bereitgestellt. In der Konfigurationsdatei des Indexers[110] können die Listen der Stop-words, Trennzeichen sowie Akronyme und Synonyme angepasst werden. Laut Tutorial können eigene Suchformulare definiert werden. Dies wurde jedoch nicht getestet.

4.3.3. Fedora Commons

Für einfache Suchen im Datenbestand eines Repositoriums stellt Fedora ein Webinterface zur Verfügung. Falls man sich während der Installation für die Verwendung des Mulgara Triple Stores entschieden hat, findet sich eine ebenfalls browserbasierte Schnittstelle zum sogenannten Fedora Resource Index Query Service. Schnittstellen, über welche andere Anwendungen Zugang zu den Daten eines mit Fedora realisierten Repositoriums erhalten können, bestehen neben den oben genannten Suchmöglichkeiten über die sogenannten API-A(ccess) und API-M(anagement). Diese bieten funktional getrennt eine Vielzahl von Methoden, die als Web Services erreichbar und mittels WSDL dokumentiert sind. Außerdem stellt Fedora eine OAI-2.0[111] Schnittstelle bereit.

Bei Fedora können Suchanfragen (wie bei DSpace) auf bestimmte Collections eingeschränkt werden. Jedoch kann auch Fedora Search nur die systeminternen Metadaten (FOXML) durchsuchen. Ein Zugriff auf die Data-Streams ist mit Fedora Resource Index Search nicht möglich. Eine Lösung bietet der Fedora Generic Search Service, welcher auf der Grundlage von Lucene arbeitet und eine einfache Anpassung von Suchfeldern durch XSLT-Transformation der Metadaten erlaubt.[112] Mittels dieses Suchdienstes, können alle Data-Streams innerhalb des FOXML nach benutzerdefinierten Schemata indiziert und durchsucht werden.

4.3.4. Greenstone

Die Indexer von Greenstone MG und MGPP benutzen GDBM-Datenbanken zur internen Verwaltung. Diese sind entscheidend für den Leistungsumfang der Suchfunktionen von Greenstone. In Greenstone kann zunächst wie z.B. auch in DSpace nach vorgegebenen Kategorien (wie Autor, Titel, Jahr oder Thema) in den Datensätzen geblättert werden. Eine freie Suche erlaubt den Zugriff auf alle Datenfelder der Metadaten,

109) http://lucene.apache.org/ (02.02.2010)
110) /eprints3/archives/test/cfg/cfg.d/indexing.pl
111) http://www.openarchives.org/ (25.10.2010)
112) http://www.fedora-commons.org/confluence/display/FCSVCS/Generic+Search+Service+2.2 (01.02.2010)

mittels der MG (Managing Gigabytes) auch auf benutzerdefinierte. Mit Hilfe der mitgelieferten Erweiterung von MG, genannt MGPP, können auch Volltextsuchen durchgeführt werden. Hierzu muss die Collection mit MGPP erstellt und indexiert werden. MGPP bietet zur Verfeinerung der Suche neben den üblichen booleschen Operatoren auch Optionen für Case Sensitivity, Stemming, Wortabstand und Wortgewichtung. Für Greenstone gibt es eine in C++ und eine in Java geschriebene Version von MGPP. Neben MG und MGPP bietet Greenstone außerdem standardmäßig die Verwendung von Lucene als Indexer an. Weder Lucene noch MGPP unterstützen Linkstrunkierung innerhalb von Greenstone.

4.4. SCHNITTSTELLEN

Das Thema Schnittstellen besitzt für ein Datenrepositorium eine besondere Bedeutung. Dies liegt daran, dass aus unserer Perspektive für eine sinnvolle Verarbeitung von Datensätzen einige der in Repositorien verwendeten Schnittstellen nicht sinnvoll und andere zu implementieren sind. So arbeitet eine Initiative zum Beispiel an einem Directory Interchange Format[113] zum Austausch von wissenschaftlichen Datensätzen. Eine Z39.50 Schnittstelle wird hingegen nicht benötigt. Wünschenswert wäre deswegen vor allem die Möglichkeit auf der Basis vorhandener Schnittstellen eigene problemspezifische Services entwickeln zu können.

4.4.1. DSpace

DSpace bietet eine OAI- und eine SOAP-Schnittstelle sowie RSS. Außerdem erlaubt sein SWORD-Interface („Simple Web-Service Offering Repository Deposit“)[114] den Austausch mit anderen Repositorien wie z.B. Fedora und Greenstone. Des weiteren bietet DSpace mit dem Lightweight Network Interface (LNI) eine sehr gut dokumentierte[115] Schnittstelle. Sie kapselt, ihrem Namen leicht widersprechend, mehrere Möglichkeiten Daten einzuspeisen bzw. auf von DSpace verwaltete Daten und Metadaten zuzugreifen. Zum einen ist ein eigenständiger Kommandozeilen-Client in der Dokumentation verlinkt; gleichzeitig stellt dieser Dienst aber auch eine WebDAV-Schnittstelle zur Verfügung, auf die z.B. von der Kommandozeile mit gängigen Programmen wie z.B. cadaver[116] zugegriffen werden kann. Drittens bietet das LNI auch noch eine SOAP-Schnittstelle. Die zugehörige WSDL-Datei findet sich in der JAR-Datei des oben erwähnten Client-Programms.

Für den gemeinsamen Betrieb mehrerer DSpace-Instanzen oder den Datenaustausch mit anderen Systemen kann eine Java-API für die Programmierung entsprechender Erweiterungen genutzt werden. Beim Datenexport in XML fiel auf, dass Daten lediglich in ein voreingestelltes, nicht anpassbares Format exportiert werden können.

113) http://gcmd.gsfc.nasa.gov/User/difguide/difman.html
114) http://www.swordapp.org/ (13.10.2010)
115) http://wiki.dspace.org/index.php/LightweightNetworkInterface (13.10.2010)
116) http://www.webdav.org/cadaver/ (13.10.2010)

4.4.2. EPrints

SWORD

Laut Homepage des SWORD-Projektes[117] ist eine SWORD-Schnittstelle vorhanden. In der Dokumentation von EPrints gibt es aber einen Hinweis, dass sie erst ab Version 3.2 zu einer Kernkomponente wird.[118]

OAI

EPrints unterstützt den OAI-Harvester für Metadaten (OAI 2.0), zunächst unter der Einschränkung, dass sich diese in METS, MODS oder Dublin Core ausdrücken lassen.[119] Mit Hilfe eines hauseigenen XSLT-Skripts[120] zur Transformation nach HTML ist die entsprechende Ausgabe auch mit einem Browser gut benutzbar.

Im- und Export über das Webfrontend

EPrints bietet über das Webfrontend den Import aus den Formaten XML, bibTeX, DOI, PubMed und PubMed XML an. Der Import von Metadaten im XML-Format wurde getestet und funktionierte fehlerfrei. Jedoch konnten bei der bisherigen Recherche noch keine Konfigurationsmöglichkeiten für das XML-Importformat gefunden werden. Weder auf Admin-Ebene noch in den Konfigurationsdateien ließen sich Einstellungsmöglichkeiten für die Anpassung des Importformats an die Ausgangsdatei oder die Zuweisung von Datenfeldern finden. Dem bisherigen Eindruck nach können solche Anpassungen wahrscheinlich nur auf einer tieferen Ebene, vermutlich nur innerhalb der Geschäftslogik vorgenommen werden, was mit erhöhtem Aufwand verbunden wäre. Die fehlende Anpassungsmöglichkeit des Importformats an die Bedürfnisse des PDR ist daher als ein beachtenswerter Nachteil von EPrints zu bewerten.

Daten in EPrints können nach MODS, METS, Dublin Core und DL, in die Literaturformate BibTeX, refer, EndNote und nach GoogleEarth sowie Smilie TimeLine exportiert werden.

Laut einer alternativen Repositorien-Evaluation sind auch Massen-Im- und Exporte möglich[121], was in dieser Evaluation jedoch nicht getestet wurde.

Web Services

EPrints bietet derzeit keine Unterstützung für eine Web-Service-API wie z.B. SOAP oder REST an[122], allerdings ist für die Version 3.2, deren Termin noch völlig offen ist, ein REST-Interface geplant. Ähnliche Hinweise gibt es zur Entwicklung einer SOAP-API für EPrints.[123] Für die Zwecke des PDR ist das Fehlen von Web-Service-Schnittstellen ein entscheidender Nachteil von EPrints, da es ohne derartige Schnittstellen den absehbaren Anforderungen nicht genügen kann.

---

117) http://www.swordapp.org (29.03.2010)
118) http://wiki.eprints.org/w/SWORD (29.03.2010)
119) Dazu wird in einer Repositorien-Evaluation bemerkt, dass EPrints (gegenüber DSpace) hier erweitert werden könne, vgl. http://www.documanager.de/magazin/artikel_851-print_open_source_software_archivierung.html (25.01.2010)
120) http://www.eprints.org/software/xslt.php (13.10.2010)
121) vgl. http://www.dini.de/fileadmin/workshops/oa-netzwerk-februar2009/schmitz.pdf (25.01.2010)
122) vgl. Technical Evaluation of selected Open Source Repository Solutions, S. 24.
123) vgl. http://wiki.eprints.org/w/Web_Services (25.01.2010)

4.4.3. Fedora Commons

Web Services

Mittels Apache Axis bietet Fedora Web Services für den Zugang zu API-M und API-A an. Direkt nach der Installation findet man unter `http://hostname:port/fedora/servlet/AxisServlet` eine Liste der Dienste und ihrer Methoden sowie Verknüpfungen zu den entsprechenden WSDL-Dateien. Eine ausführliche Beschreibung findet man in der Fedora Repository Dokumentation.[124] Desweiteren werden der Apache FOP[125] (Formatting Objects Processor), Saxon[126] (XSLT and XQuery Processor) und ein auf ImageJ[127] basierender Dienst zur Verarbeitung von Bildern bereitgestellt. Weitere Schnittstellen sind:

- Schnittstellen für den Zugriff auf Objekte
  - Fedora Access Service (SOAP) – API-A
  - Fedora Access Service (REST) – API-A-LITE
- Schnittstellen für die Administration, das Einstellen und das Löschen von Objekten
  - Fedora Management Service (SOAP) – API-M
  - Fedora Management Service (REST) – API-M-LITE
- Weitere Schnittstellen
  - Fedora REST API – Ein einfache Kombination aus den Schnittstellen oben
  - Basic Search - Repository Registry Query (REST)
  - Basic OAI - Simple OAI-PMH Provider (REST)
  - RISearch - Resource Index Search (REST)

(fk)

4.4.4. Greenstone

Greenstone bietet eine OAI-, sowie eine Z39.50-Schnittstelle für den Austausch von Metadaten. Für Web-Services stellt es eine SOAP-Schnittstelle zur Verfügung, durch die ohne größeren Aufwand Verknüpfungen zwischen Greenstone und DSpace-Repositorien möglich sind. Collections können im METS-Format exportiert ebenso wie massenhaft in diesem Format importiert werden. Zusätzlich sind Importe in benutzerdefinierten Metadaten-Formaten möglich. Zum Import und zur Transformation von Daten in

124) Fedora Repository 3.2.1 Documentation, Abschnitt 1.5.8 Web Service Interfaces.
125) http://xmlgraphics.apache.org/fop/ (25.01.2010)
126) http://saxon.sourceforge.net/ (25.01.2010)
127) http://rsb.info.nih.gov/ij (25.01.2010)

folgenden Formaten stehen Plugin-Erweiterungen zur Verfügung: XML, MARC, CDS/ISIS, ProCite, BibTex, Refer, OAI, DSpace, METS.

5. ERGEBNISSE

5.1. PRO UND CONTRA

5.1.1. DSpace

1.) Pro:

1. DSpace bietet umfangreiche Funktionen für die Verwaltung von Dokumenten und anderen Datendateien; die Funktionen und Datentypen können ohne besonderen Aufwand flexibel angepasst werden.
2. DSpace liefert eine umfangreiche und übersichtliche Verwaltung von Benutzern und Dokumentengruppen wie Communities und Collections.
3. Anpassungen des Layout sind bei DSpace auf verschiedenen Ebenen sehr praktisch und sehr weitreichend möglich.
4. Die Anpassung des Workflows geschieht mittels benutzerfreundlicher XML-Konfigurationsdateien.
5. Die Konfiguration des Workflows, die Verwaltung von Datengruppen und die grafische Anpassung von DSpace sind gut dokumentiert.
6. Durch die Funktion „Controlled Vocabulary“ bietet DSpace nützliche Funktionen zur Klassifikation von Datensätzen auf der Grundlage von anpassbaren hierarchischen Listen von Subjects.
7. Es ist erweiterbar durch Plugins.
8. Die intern im Dublin Core-Format gespeicherten Daten können durch sog. Crosswalks DSpace-intern in externe Datenformate transformiert werden.
9. DSpace bietet neben OAI auch eine SOAP-Schnittstelle.

2.) Contra:

1. Das Repositorium setzt auf Dublin Core auf und erlaubt keinen Austausch des internen Metadata-Schemas durch ein benutzerdefiniertes Schema. Dadurch kann DSpace nicht, bzw. nur behelfsmäßig an andere Datenformate wie das des PDR angepasst werden.
2. Die vorgegebenen Importformate können nicht auf Admin-Ebene oder in den Konfigurationsdateien angepasst werden.
3. Die Dokumentation für die Erstellung und Anpassung von Metadata-Schemata ist sehr knapp und missverständlich. Eine Klasse für den Import eines Schemas fehlt in der Release-Version.

4. Datenfelder, die auf der Admin-Ebene in einem benutzerdefinierten Metadata-Schema erstellt wurden, können in den Konfigurationsdateien nicht in den Workflow integriert werden. Hier entstand bei der Evaluation der falsche Eindruck, ein eigenes Metadata-Schema könnte in DSpace importiert und anstelle von Dublin Core verwendet werden – was jedoch zu wiederholten Fehlermeldungen führte.

5.1.2. EPrints

1.) Pro:

1. EPrints bietet umfangreiche Funktionen für die Verwaltung von Dokumenten und anderen Datendateien; die Funktionen und Datentypen können ohne besonderen Aufwand flexibel angepasst werden.
2. EPrints bietet eine sinnvolle und nützliche Funktion zur Klassifikation von Daten durch das Hinzufügen von Subjects aus einer anpassbaren hierarchischen Liste.
3. EPrints liefert eine praktische und übersichtliche Benutzerverwaltung.
4. Anpassungen des Layout sind bei EPrints gut umsetzbar.
5. Gute Dokumentationen zu Konfiguration und grafischer Anpassung.
6. Erweiterbarkeit durch Plugins.

2.) Contra:

1. EPrints 3.1.x bietet nur eine OAI-Schnittstelle, jedoch keine Web-Services wie REST oder SOAP.
2. Das Repositorium ist auf die Verwaltung von Metadaten und Dateien mit Inhalten spezialisiert, daher ist die Erstellung und Anpassung von Datentypen nur in einem zu diesem Zweck beschränkten Rahmen möglich.
3. Der Workflow ist nur begrenzt anpassbar.
4. Die vorgegebenen Importformate können nicht auf Admin-Ebene oder in den Konfigurationsdateien angepasst werden.

5.1.3. Fedora Commons

1.) Pro:

1. Fedora ist sehr flexibel und erlaubt sehr weitreichende Anpassungen.
2. Die Software ist gut dokumentiert und es gibt einen breiten Anwenderkreis.
3. Fedora ist auf eine hohe Skalierbarkeit und die Verwaltung von mehr als 1 Mio. Datenobjekten ausgerichtet.

4. Fedora bietet neben OAI auch eine SOAP und REST-Schnittstelle.

2.) Contra:

1. Das Repositorium setzt auf Dublin Core auf und erlaubt den Austausch des internen Metadata-Schemas durch ein benutzerdefiniertes Schema nur unter zur Hilfenahme von Erweiterungen wie Fedora Generic Search.
2. Die Dateien des Repositoriums werden nicht in einer Datenbank, sondern nur im Dateisystem gespeichert, was möglicherweise und in starker Abhängingkeit von den zu verwaltenden Daten mit Schwierigkeiten bei der Performance verbunden ist.
3. Zu Fedora gibt es keine mitgelieferten Endbenutzern-Interfaces.

5.1.4. Greenstone

1.) Pro:

1. Greenstone bietet gute und nur wenig eingeschränkte Möglichkeiten, eigene Metadatenschemata zu erstellen und anzupassen.
2. Greenstone besitzt eine umfangreiche Verwaltung von Dokumentengruppen in Collections.
3. Anpassungen des Layout sind bei Greenstone auf verschiedenen Ebenen sehr praktisch und sehr weitreichend möglich.
4. Standardmäßig ist Greenstone mit zwei Volltextindexern MGPP und Lucene ausgerüstet.
5. Greenstone bietet neben OAI auch eine SOAP-Schnittstelle.
6. Ausgesprochen umfangreiche Dokumentation sowie sehr gute Sprachunterstützung.

2.) Contra:

1. Für jedes Datenobjekt muss eine Datei in das Repositorium geladen werden, selbst wenn ein Datenobjekt nur Metadaten enthält, muss eine leere `*.nul` Datei referenziert werden.
2. Greenstone bietet keine Benutzerschnittstelle zur Erstellung neuer Datenobjekte, dessen Workflow stark anpassbar und für Benutzergruppen einschränkbar wäre.

(cp)

5.2. SCHLUSSBETRACHTUNG

Wie eingangs beschrieben, vollzog sich unsere Evaluation der zur Verfügung stehenden Repositoriums-Software unter den Vorzeichen einer immer stärker werdenden Diskussion über Infrastrukturen für das Management und die Publikation von Forschungsdaten im Allgemeinen, sowie dem Bedarf eines konkreten Systems für Personendaten aus der geisteswissenschaftlichen Grundlagenforschung an der Berlin-

Brandenburgischen Akademie der Wissenschaften im Besonderen. Insofern erscheint es angebracht, am Ende der Evaluation die Erkenntnisse über Eigenschaften, Verhalten und Tauglichkeit von Repositoriensystemen in Verbindung mit einem Personendaten-Repositorium zurück in die allgemeine Debatte selbst zu bringen und auszuweiten. Es ist zu hoffen, dass mit ähnlichen Beiträgen gerade auch der Geisteswissenschaften das Profil der Anforderungen geschärft wird, welches forschende Einrichtungen an Systeme zur Publikation von Forschungsdaten stellen, damit Entwickler von entsprechender Software in die Lage versetzt werden, darauf zu reagieren. Eine Realisierung des Potenzials, das in der Onlinepublikation von Forschungsdaten steckt, hat gerade erst begonnen.

Einer der sensibelsten Punkte der Evaluation war es, die Nutzbarkeit der Objektbehandlung und der Inhaltsmodelle für die Abbildung komplexer und heterogener Daten zu testen. Nicht nur entscheidet sich an dieser Stelle, ob ein Datenbestand überhaupt in einem bestimmten Repositorium abgebildet werden kann, die Flexibilität und Form der Abbildungsstruktur entscheidet auch später darüber, wie die Daten genutzt werden können. Grundsätzlich haben sich zwei Möglichkeiten dafür angeboten, Datensätze in einem Repositoriensystem festzuhalten: als Dokument oder in den Metadaten. Die dokumentenzentrierte Fokussierung einiger Repositoriensysteme führt dazu, dass man nicht bei jeder Software die Wahl besitzt: So sahen EPrints und Greenstone den Import eines Dokuments für ein Objekt im Repositorium vor bzw. machten es zur Voraussetzung[128]. Eine Schwierigkeit eines solchen Ansatzes ist es, dass die Volltextsuche, die von den meisten Systemen zur Verfügung gestellt wird, bei Daten nur bedingt brauchbar ist. Denn entscheidend bei der Suche in Datensätzen ist nicht nur, in welchem Datensatz ein bestimmtes Datum zu finden ist, sondern auch wo innerhalb der Daten. Gerade bei abstrakten Werten, die leicht an verschiedenen Stellen auftreten können, wird dies zum Problem[129]. Es ist deshalb zur Realisierung einer sinnvollen Indexierung von Daten noch erstrebenswerter als bei der Suche über textuelle Ressourcen, nicht nur die Daten selbst, sondern auch Ihren Kontext – ihre Position im Verhältnis zu anderen Daten – zu indizieren[130]. Dies funktioniert in den von uns getesteten Systemen lediglich bei den Metadaten. Information Retrieval bedeutet in Repositorien bisher fast ausschließlich Document Retrieval. Dieser Bereich muss um Aspekte des Data Retrievals erweitert werden. Der zweite Weg – die Daten als Metadaten zu behandeln und im Repositorium ein reines Metadatenobjekt zu kreieren – besitzt den Vorteil, dass so eine feldspezifische Suche eher möglich ist, er verlangt aber von der Repositoriensoftware eine weitreichende Flexibilität im Umgang mit Metadatenschemata. Darüber hinaus werden so Daten und Metadaten zusammengeworfen, was unter Administrations- und Datenpflegeaspekten inkonsequent erscheint[131]. Insbesondere ist dies unter dem Aspekt zu berücksichtigen, dass es für die Beschreibung und den Austausch von Datensätzen eigene Metadatenstandards gibt, die es gesondert zu implementieren gilt[132]. Auch liegt bei fast allen

128) EPrints propagiert selbst diese Lösung, indem es das Label Datensatz als mögliche Dokumenttyp-Deklaration zur Verfügung stellt.

129) Als Beispiel kann ein Datensatz dienen, in dem ein Graph beschrieben ist. Eine klassische Volltextsuche kann nur herausfinden, in welchem Datensatz ein bestimmter Wert vorkommt, aber nicht, welcher Datensatz einen Graphen beschreibt der durch den Punkt x=3.5 und y=4.6 läuft

130) Ein mögliche Lösung wäre es in Zukunft neben der Volltextsuche auch eine Suche auf die interne XML Struktur vom XML Dokumenten anzubieten. Datensätze ließen sich dann in XML einpflegen und sinnvoll durchsuchen. Dieser Ansatz würde allerdings dazu zwingen die Daten in XML abzulegen.

131) Fedora ist in der Lage dieses Problem dadurch zu lösen, dass Metadaten selbst nur Datenströme wie Dokumente sind und jedes Objekt so viele Datenströme besitzen kann wie es notwendig ist. So ließe sich ein Datenstrom für die Daten und einer für die Metadaten erstellen.

132) Bisher arbeiten solche Inititiaven eher fachbezogen. So haben sich Metadatenstandards und -spezifikationen z.B. für die

Repositoriensystemen, die wir getestet haben, der notwendige Grad an Flexibilität noch nicht vor oder kann nur über Umwege erreicht werden[133].

Ein weiterer Aspekt, der unserer Meinung nach mit der Veröffentlichung von Forschungsdaten an Gewicht gewinnt und den wir daher untersucht haben, war die Möglichkeit, Daten und Datensätze durch die Definition von Beziehungen mit internen und externen Ressourcen zu verknüpfen. Heterogene oder komplementäre Datensätze innerhalb eines Repositoriums lassen sich so zusätzlich strukturieren und es lassen sich Abhängigkeiten voneinander beschreiben. Mehr noch lässt sich in hybriden Repositorien das Ziel der Enhanced Publication[134] verfolgen. Notwendig erscheint eine solche Fähigkeit des Systems ebenfalls in Zusammenhang mit den Zielen der Linked Data Initiative. Die Verknüpfung eigener mit fremden Datenbeständen und Ressourcen kann als erheblicher Multiplikator für die Rezeption und die Qualität der eigenen Datenbestände wirken. Um so engagierter ist deshalb auf die Defizite hinzuweisen, die die getesteten Systeme auf diesem Gebiet besitzen[135]. Eine nachdrücklich hervorzuhebende Ausnahme stellt Fedora dar. Für jedes Fedora-Objekt lassen sich unbegrenzt viele Beziehungen deklarieren. Dies geschieht noch dazu durch die Verwendung des Resource Description Framework (RDF) unter Verwendung einer Technik, die von Linked Data selbst propagiert wird und auf dem Gebiet der semantischen Erschließung von im Web zugänglichen Ressourcen einen Standard darstellt[136]. An diesen Komplex schließt sich direkt die Notwendigkeit der Erweiterung der Schnittstellen von Repositorien an, damit die neuen Informationstypen und zum Teil auch datenspezifischen Erschließungen auch in einer globalen Repositorien-Infrastruktur sichtbar sind und abgefragt werden können[137].

Insgesamt lässt sich feststellen, dass die Administration und Publikation von Forschungsdaten in Repositoriensystemen trotz eines überall formulierten Bedarfs noch erhebliche Schwierigkeiten mit sich bringt. Solange dies der Fall ist, erscheint es um so wichtiger, dass die Repositorien auf substantieller Ebene leicht anpass- und erweiterbar sind. Nicht zuletzt auch aus diesem Grund haben wir uns am Ende der Evaluation für Fedora Commons entschieden, da es sich auf Kernkomponenten beschränkt und dem Entwickler große Freiheiten lässt[138]. Bei der Evaluation wurden unzureichende Dokumentation und Unzugänglichkeit sowie Unübersichtlichkeit des Codes daher besonders hervorgehoben.

Geowissenschaften (Content Standard for Digital Geospatial Metadata – http://www.fgdc.gov/metadata/csdgm/), die Sozialwissenschaften (Data Documentation Initiative – http://www.ddialliance.org) und für quantitative Daten (Directory Interchange Format – http://gcmd.gsfc.nasa.gov/User/difguide/difman.html) entwickelt.

133) Greenstone und FEDORA heben sich von diesem Gesamteindruck ab, wobei beide die Freiheit im Anlegen eigener Datenschemata dadurch einbüßen, das bei dem einen ein Objekt mit einem Dokument verbunden sein muss und bei dem anderen die Verwendung von Dublin Core auch unter der Verwendung eines eigenen Datenschemas obligatorisch ist.

134) Gemeint ist mit dem Begriff unter anderem die Verfügbarmachung und Verknüpfung verschiedener Materialisierungen eines Forschungsprozesses wie etwa Forschungsdaten und die darauf aufbauende Publikation. Sierman, Barbara, Birgit Schmidt und Jens Ludwig. Enhanced Publications : Linking Publications and Research Data in Digital Repositories. Surf / EU-Driver. Amsterdam: Amsterdam University Press, 2009.

135) DSpace und Greenstone sehen keinerlei Mittel vor, Beziehungen zu definieren. Bei EPrints erwies sich die Implementierung als fehleranfällig.

136) http://www.fedora-commons.org/confluence/display/FCR30/Digital+Object+Relationships (Stand 08.02.2010)

137) Für Fedora liegt die Erweiterung *oreprovider* vor, die die REST-EXT Schnittstelle in die Lage versetzt Resource Maps unter Verwendung von OAI-ORE zur Verfügung zu stellen.

138) Nicht ungewürdigt soll an dieser Stelle auch die schon erwähnte leichte Integrierbarkeit des DSpace-Kerns in Eclipse bleiben, die alle Voraussetzungen für eine schnelle Anpassung des Repositoriums an die eigenen Bedürfnisse mitbringt.

Die Schwierigkeiten bei der Verwendung von Repositoriensoftware für ein Datenrepositorium waren in der Evaluation nicht unerheblich. Viele davon, wie z.B. der Umgang mit selbst entwickelten Metadaten-Schemata, traten in der einen oder anderen Weise in jedem der getesteten Systeme auf. Andere besaßen ihren Ursprung in der Fokussierung auf Dokumente als Ressourcen. Zusammengefasst lassen die erläuterten Probleme die Frage zu, inwieweit ein Repositorium das richtige System für die Lagerung und Publikation von Forschungsdaten darstellt oder ob nicht intuitivere Lösungen wie eine Datenbank oder ein „Data Warehouse“ zu bevorzugen sind. Sicherlich gehört es im Gegensatz zu Repositorien zur genuinen Aufgabe einer Datenbank fallspezifische Datenmodelle abbilden zu können, doch besitzen Repositoriensysteme gegenüber diesen nicht zu vernachlässigende Vorteile, die sie für eine Publikation von Forschungsdaten interessant machen. Datenbanken stellen an sich noch keine Infrastruktur dar, sondern finden ihre Anwendung als Teilkomponente größerer zusammenhängender Systeme zur Erfüllung bestimmter Aufgaben. Dieser Anforderungshorizont ist bei Repositoriensystemen schon wesentlich konkreter formuliert und führt dazu, dass Repositoriensoftware in dem meisten Fällen ein integriertes Produkt aus verschiedenen Komponenten liefert. Ein zentraler Aspekt dieses Aufgabenhorizonts ist der der Publikation. Repositorien sind daher darauf ausgelegt, dass eine wissenschaftliche oder interessierte Öffentlichkeit in einer spezifischen, webbasierten Art und Weise auf sie zugreifen will. Daher bringen viele Repositoriensysteme bereits bestimmte Filter zur Verarbeitung und Oberflächen zur Präsentation der Inhalte mit. Auch zielen Repositorien auf eine breitere Nutzergruppe ab. Als System, das als Bestandteil eines institutionellen Alltags konzipiert ist, unterstützt es Nutzer ohne große Nähe zu technischen Fragestellungen beim Einstellen und Verwalten von Inhalten. Dies geschieht ebenfalls durch die Bereitstellung von Oberflächen, aber auch durch die Konfigurierbarkeit von Workflows. Darüber hinaus umfasst der Publikationsbegriff im Sinne gegenwärtiger Online-Publikation nicht nur das Sicherstellen der Sichtbarkeit für den Menschen in Form von Portalen. Repositorien werden auch als Bestandteil einer übergeordneten Informationsinfrastruktur verstanden, in die es sich zu integrieren gilt.[139] Die spezifischen Schnittstellen, mit denen dies geschieht, wie OAI-PMH oder Z39.50, sind, auch wenn sie wie gezeigt nicht ausreichen, immerhin schon Kernbestandteil dieser Systeme. Bei Datenbanken müssten solche Schnittstellen auf der Basis der datenbankeigenen Schnittstelle erst entwickelt werden. Die oben bereits angesprochene Informationsinfrastruktur formiert sich primär unter der Leitidee der Dezentralität, die die Voraussetzung für einen ungehinderten Informationsfluss und Transparenz in den Wissenschaften bildet. Auf informationstechnologischer Ebene zeigt sich dies durch die immer stärker werdende Fokussierung auf verteilte Systeme und Grid-Strukturen[140]. Da Repositoriensysteme diesem Kontext entsprungen sind, bringen sie auch hervorragende Voraussetzungen mit, diesem Anspruch auf dezentrale Vernetzung zu entsprechen. Konkret spiegelt sich dieses Problem beim Personendaten-Repositorium in dem Ziel wider, dass auch andere Institutionen diese von uns entwickelte Infrastruktur für ihren Themenbereich benutzen können sollen und gleichzeitig eine Suche im gesamten Personendaten-Repositorien-Netz trotz des Fehlens eines zentralen Knotenpunktes angestrebt wird. Dies ist nötig, da zum jetzigen Zeitpunkt noch gar nicht absehbar

139) Siehe: Juling, W. “Vom Rechnernetz zu e-Science.” PIK – Praxis der Informationsverarbeitung und Kommunikation 32, no. 1 (2, 2009): 33-36.

140) Siehe: Schwiegelshohn, U. “Grids als neue Komponenten des Integrierten Informationsmanagements.” PIK – Praxis der Informationsverarbeitung und Kommunikation 32, no. 1 (2, 2009): 29-32. oder: “DINI: Informations- und Kommunikationsstruktur der Zukunft.” BIBLIOTHEK Forschung und Praxis 33, no. 2 (9, 2009): 209-215.

ist, wo und wie viele künftige Personendaten-Repositorien es geben wird. Insbesondere die von einigen Repositoriensystemen zur Verfügung gestellten SOAP und REST Schnittstellen und ihre Konfigurierbarkeit erleichtern die anstehenden Aufgaben für eine effiziente Vernetzung erheblich[141].

Zusammenfassend sind dies auch die Gründe, weshalb wir uns für ein Repositoriensystem zur Speicherung und Administration unser Datenbestände entschieden und nicht auf eine stärker datenbankorientierte Lösung wie z.B. ein Data Warehouse zurückgegriffen haben.

Wir fühlen uns trotz aller aufgezeigten Schwierigkeiten in dieser Entscheidung bekräftigt, da sowohl fördernde als auch geförderte Institutionen die Publikation von Forschungsdaten immer stärker in den Repositorienkontext rücken. Zu erwähnen sei hier insbesondere die kürzlich erfolgte Ausschreibung Informationsinfrastrukturen für Forschungsdaten von der Deutschen Forschungsgemeinschaft[142] sowie das Positionspapier Enhanced Publications[143] der DRIVER Initiative, in dem explizit eine Erweiterung des Publikationshorizonts von Repositorien auch auf Forschungsdaten befürwortet wird. Die Evaluation zeigte einige Schwierigkeiten auf, die es auf der Wegstrecke hin zu einer Publikationsinfrastruktur für Forschungsdaten in Repositorien zu bewältigen gilt.

Fabian Körner
Christoph Plutte
Torsten Roeder
Niels-Oliver Walkowski

141) An dieser Stelle erweist sich auch die Möglichkeit der Definition von eigenen Services unter Fedora als ausgesprochen fruchtbar.
142) DFG. "Informationsinfrastrukturen für Forschungsdaten," 2010. http://www.dfg.de/foerderung/info_wissenschaft/info_wissenschaft_10_02/index.html. (Stand 01.2010)
143) Sierman, Barbara, Birgit Schmidt, und Jens Ludwig. "Enhanced Publications : Linking Publications and Research Data in Digital Repositories." Surf / EU-Driver. Amsterdam: Amsterdam University Press, 2009. http://dare.uva.nl/document/150723.