

OPUS Entwicklung und Perspektive

Christof Mainberger (Bibliotheksservice-Zentrum Baden-Württemberg)
Vortrag auf dem 100. Bibliothekartag in Berlin am 10.6.2011

Opus ist eine Software für institutionelle Repositorien – und Bibliografien – und dient der Veröffentlichung, Erschließung, Administration, Recherche sowie Verbreitung elektronischer Publikationen. Es ist über eine Vielzahl standardisierter Schnittstellen in die Bibliothekssysteme und die Netzwerke des Open-Access eingebettet, lässt sich flexibel vom Betreiber anpassen und bietet Interaktionsmöglichkeiten mit lokalen Systemen und Nutzern. Technologisch ist OPUS eine Open-Source-Webanwendung basierend auf Open-Source-Komponenten (PHP, MySQL, Apache, Solr).

Die Frage, was OPUS von vergleichbaren Softwaresystemen unterscheidet und z.B. eine Weiterentwicklung rechtfertigt, wird mit folgendem Credo beantwortet: „OPUS ist aufgrund seiner Architektur ein technisch einfach zu installierendes und zu betreibendes System, das im Produktionsbetrieb sehr stabil und wartungsarm läuft“.

Der niedrige Aufwand und die Robustheit zählen sicherlich zu den Gründen, warum OPUS im nationalen Rahmen nach wie vor die am meisten eingesetzte Software für Anwendungen dieser Art ist: Uns sind aktuell über hundert OPUS-Anwendungen bekannt, wobei gut die Hälfte davon beim BSZ, hbz und KOBV gehostet werden. Diese Möglichkeit, die insbesondere auch kleineren Einrichtungen den Aufbau von institutionellen Repositorien ermöglicht, ist eine weitere Ursache für die weite Verbreitung von OPUS.

OPUS, im Jahr 1997 in einem Entwicklungsprojekt als „OnlinePublikationsverbund Universität Stuttgart“ ins Leben gerufen, blickt zwischen der Version 1.2 aus dem Jahr 1998 und der Version 4.1 von Ostern 2011 auf mittlerweile insgesamt neun Versionen zurück. Ältere Versionen sind dabei parallel zum jüngsten Release noch im Einsatz. Parallel zu OPUS 4, das beim KOBV weiterentwickelt wird, wird auch OPUS 3 von der Universitätsbibliothek Stuttgart weiterentwickelt.

Die OPUS-Ökumene stellt sich somit mittlerweile als weitverzweigter Baum dar: Vereinzelt sind noch OPUS 2 Installationen im Einsatz; das Gros bildet OPUS 3.2, welches selbst in etlichen angepassten Variationen verzweigt ist. Die Hostings beim BSZ arbeiten mit OPUS 3.3, was sich von 3.2 lediglich durch eine neue Oberflächentechnologie unterscheidet und dessen Entwicklungen nachzieht. Schließlich gibt es OPUS 4, ganz aktuell als Version 4.1.1, mit vier produktiven Installationen beim KOBV sowie in Kaiserslautern und weiteren beim KOBV und dem BSZ, die sich im Aufbau befinden.

Auch Qucosa, das in der SLUB Dresden entwickelt wurde, beruht z.T. auf der Zusammenarbeit im OPUS 4 Projekt.

Konsequenz dieser Situation ist, dass die OPUS-Versionen zwar eine gemeinsame konzeptionelle Herkunft teilen und auf gemeinsame langjährige Erfahrungen gegründet sind, es aber keine einheitliche OPUS-Community mehr gibt. Nachdem OPUS 4 eine vollständige



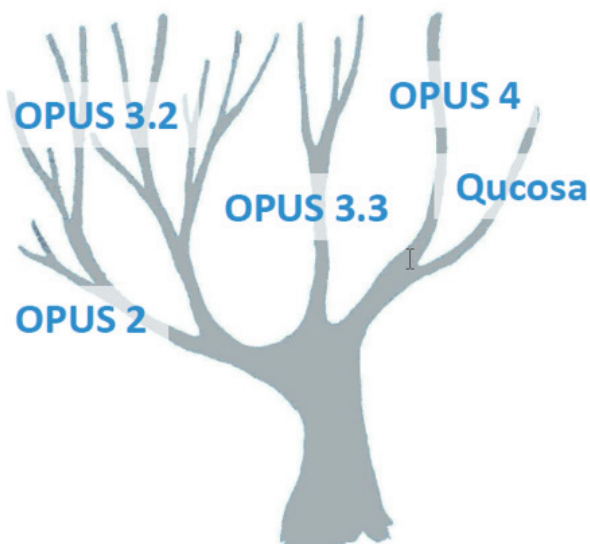
Die Zahl der OPUS-Anwendungen ist auf über hundert gewachsen, wobei über die Hälfte in Verbundzentralen gehostet wird.

Neuentwicklung darstellt, gibt es auch keine gemeinsame Softwarebasis und keine koordinierte Fortentwicklung. Nicht ausgemacht ist allerdings, ob diese Vielfalt nun eher einen Fehler darstellt oder nicht vielmehr ein Feature.

Institutionelle Repositorien haben sich aus der Nischen-situation, in der sie sich in den Bibliotheken entwickelt und jahrelang um Resonanz gekämpft haben, an die Schnittstelle zwischen Hochschule, Nutzer und Bibliothek geschoben. Ursachen liegen in einer zunehmenden Integration der IT-Infrastruktur an Universitäten und Hochschulen, in der die Bibliotheken ihre Kompetenz für Literatur reklamieren, hierbei aber nicht die Biblio-

theKssysteme in den Vordergrund stellen, die durch bibliothekarische Geschäftsgänge eng reglementiert sind. Stattdessen werden die institutionellen Repositorien positioniert, die für digitale Dokumente und Objekte besser geeignet sind und deren Datenmodelle und Funktionalitäten leichter in den lokal erforderlichen Kontext eingepasst werden können.

Repositorien finden sich daher in einem Spannungsfeld unterschiedlicher Anforderungen wieder: Zur Einbettung in bibliothekarische Kontexte müssen sie Verbundkatalog, Bibliothekssystem sowie Dienste wie BASE, OAI-



Die OPUS-Entwicklung verzweigt sich in zahlreiche Äste: Fault oder Feature?

ster und DRIVER bedienen; sie müssen zur Erschließung die Verwendung von Normvokabular forcieren, die Pflichtabgabe an die DNB und digitale Langzeitarchivierung ermöglichen und persistentes Referenzieren und damit zuverlässiges Zitieren erlauben.

Die Nutzer erwarten komfortable Web 2.0-Oberflächen für Recherche, Präsentation und insbesondere für die Publikation, die Anbindung an soziale Netzwerke und Literaturverwaltungssysteme sowie die Sichtbarkeit in Google.

Die Hochschule möchte – soweit dies Literatur betrifft – mit dem Repository nach innen und nach außen ihre Forschungs- und Lehrleistung sichtbar machen, also insbesondere eine Bibliografie realisieren, Forschungsinformationssysteme anschließen, primäre Forschungsdaten verwalten und die Aktivitäten auf dem Repository schließlich im Rahmen der leistungsorientierten Mittelvergabe belohnen.

Während die bibliothekarischen Belange über Standards wie XMetaDissPlus, XEpicur, OAI, URN weitgehend einheitlich gestaltet werden können, sind Anforderungen aus dem Hochschulbereich weit weniger geklärt und stellen sich lokal sehr unterschiedlich dar. Als Beispiel sei die Auseinandersetzung aufgeführt, welche Funktionalitäten eigentlich eine Hochschulbibliografie in OPUS 4 ausmachen sollen:

Ein erster Vorschlag stellte fest, dass die Hochschulbibliografie auch Einträge in OPUS erlauben müsse, für

die kein Volltext vorhanden ist, also z.B. zu Aufsätzen, die anderweitig veröffentlicht sind. Diese Charakterisierung genügt allerdings nicht, da natürlich auch Einträge mit Volltext zur Bibliografie gehören können, also sich die Bibliografie nicht als die Einträge ohne Volltext definieren lässt.

Ein weiterer Vorschlag hob darauf ab, dass beim Bibliografieren oft Metadatenelemente, die bei Hochschulschriften zum Beispiel für die Konformität zu DRIVER etc. verlangt werden, nicht zur Verfügung stehen: Ein Abstract oder ein Titel auf Englisch oder Angaben wie z.B. Vornamen aller Autoren liegen dem Mitarbeiter, der die Literaturliste seiner Professorin eingeben muss, oft einfach nicht vor. Das führte zu abweichenden Metadatenschemata für die Bibliografie mit weniger obligatorischen Feldern als im Repository verlangt, also zu uneinheitlichen Beschreibungen von Publikationstypen.

Als vorläufiges Ergebnis erlaubt die Version 4 nun einfach die Zuordnung eines OPUS-Eintrags zur Bibliografie per Häkchen, wobei langfristig sicher die Zuordnung zu unterschiedlichen Bibliografien realisiert werden sollte – neben der Hochschule auch z.B. eines Fachbereichs etc. – und der Gesichtspunkt unterschiedlicher Masken mit unterschiedlichen Pflichtfeldern auch zu berücksichtigen ist.

OPUS 3 folgt einem sehr pragmatischen Programmierstil, der im Wesentlichen einem prozeduralen Paradigma folgt, in das nur wenige modulare und objektorientierte Ansätze einfließen. Diese Einfachheit macht sicherlich einen wichtigen Moment des Erfolgs dieser Software aus: Der Quellcode kann mit sehr elementaren Programmierkenntnissen verstanden und an lokale Erfordernisse angepasst werden. Die Programmierung von OPUS 3.3, die hauptsächlich entsprechend dem MVC-Pattern die Oberflächengestaltung von der Programmlogik abtrennt, wurde am BSZ von einem Systembibliothekar ohne spezielle Informatikvorkenntnisse ausgeführt.

Gleichzeitig ist der Quellcode durch seine lange Genese inzwischen ziemlich verwachsen, z.B. wird von fast jedem einzelnen Skript auf die Datenbank zugegriffen, und die unterschiedlichen Programmierer haben bei der Renovierung einzelner Bereiche deutlich ihre Handschriften hinterlassen.

Dennoch hat die Universitätsbibliothek Stuttgart begonnen, OPUS 3.2 weiterzuentwickeln. Erste Ergebnisse sind eine Reaktivierung von PrintOnDemand, was nach der Einstellung des ProPrint-Dienstes eingestellt worden war, sowie die Erhöhung der Sichtbarkeit in Google Scholar. Sehr wichtig für die zahlreichen laufenden OPUS 3 Installationen ist insbesondere die Implementierung der Version 2.0 von XMetaDissPlus, da über diese Schnittstelle zukünftig die DNB die Pflichtablieferung abwickeln wird.

Weitere Vorhaben der Universitätsbibliothek Stuttgart betreffen das DINI-Zertifikat 2010, Forschungsdatenanbindung, OpenAIRE/DRIVER sowie die Bibliografiefunktion. Langfristig ist auch eine objektorientierte Überarbeitung vorgesehen. Die Universitätsbibliothek Stuttgart folgt bei diesen Arbeiten dem eigenen Bedarf, stellt die Resultate allerdings allen Anwendern zur Verfügung.

OPUS 4 wurde von vornherein mit Blick auf eine moderne Softwarearchitektur konzipiert und ist daher streng in Ebenen und Module gegliedert. Über einer technologischen Basis, die vom Apache HTTP-Server, MySQL, PHP und Solr gebildet wird, werden vom ZENDFramework Basisdienste bereitgestellt, die eine Reihe von Kernkomponenten nutzen, welche wiederum in den Anwendungsmodulen verwendet werden.

Das Ziel dabei ist, Erweiterungen des Systems einfach durch Konfiguration zu ermöglichen, etwa um neue Publikationstypen oder Metadatenelemente. Für weitergehende Anpassungen werden feste APIs bereitgestellt, die eine einheitliche Programmierung und Einbindung von zusätzlichen Modulen gewährleisten.

Sollen dadurch die Probleme von OPUS 3 vermieden werden, sind nun sehr anspruchsvolle Kenntnisse zur Anpassung des Programmcodes erforderlich, die sich bis auf nicht gerade landläufige Technologien wie ZENDFramework, Solr, XSLT erstrecken. Es wird sich als wesentlicher Erfolgsfaktor von OPUS 4 herausstellen, ob die bereitgestellten Schnittstellen ausreichend dokumentiert und das zu ihrer Programmierung erforderliche Personal gefunden werden kann.

Die Fortentwicklung von OPUS 4 hat der Kooperative Bibliotheksverbund Berlin Brandenburg übernommen. Die Version 4.1.1, die er kurz vor dem Bibliothekartag 2011 veröffentlicht hat, enthält alle wesentlichen Funktionalitäten von OPUS 4 und bildet eine stabile Basis für die Migration von OPUS 3-Anwendungen. Die nächsten Ziele liegen nun in der Konsolidierung und Fehlerbearbeitung.

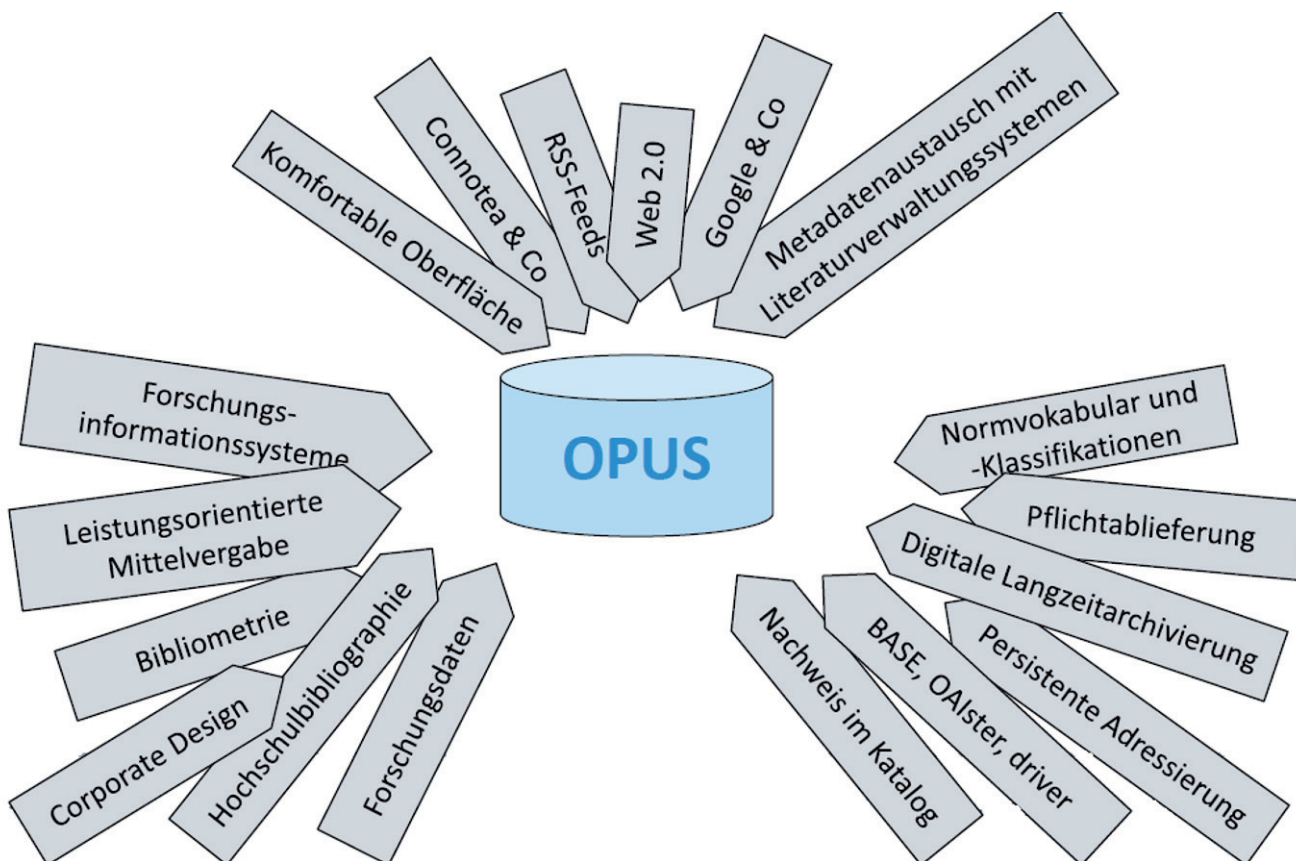
Das Release von OPUS 4 durch den KOBV markiert das glückliche Ende einer langwierigen Entwicklungszeit, in der als Projektpartner das BSZ, das hzb, der KOBV, die Universitätsbibliothek Bielefeld, die Bibliothek der Technischen Universität Hamburg-Harburg, die Saarländische Universitäts- und Landesbibliothek Saarbrücken, die Universitätsbibliothek Stuttgart sowie die Sächsische Staatsbibliothek Landes- und Universitätsbibliothek Dresden zusammengearbeitet haben.

Von den ersten Konzeptionen im Jahr 2006 bis zum Release der Version 4.1 im Jahr 2011 dauerte es dabei fast fünf Jahre. In dieser Zeit mussten Release-Termine immer wieder verschoben werden, bis OPUS 4 auf dem Bibliothekartag 2010 als Entwicklungsrelease und nach einer abschließenden Entwicklungsphase beim KOBV im November 2010 in der Version 4.0 veröffentlicht wurde. Aber erst die Version 4.1, die der KOBV zu Ostern 2011 veröffentlichte, wurde wirklich als Ersatz für eine bestehende OPUS 3-Instanz eingesetzt.

Dies wirft die Frage auf, was zu der – gemessen an der Dynamik des Internets – zu großen Verzögerung geführt hat?

Zum einen war die OPUS-Community nicht in der Lage, fachliche Anforderungen ausreichend konsistent und vollständig zu formulieren und hat die Entwickler zu sehr sich selbst überlassen. Als Beispiel wurde bereits oben die Diskussion um die Bibliografiefunktion dargestellt.

Zweitens ist es trotz Einsatz einer Entwicklungsplattform und entsprechender Kommunikationsmittel nicht gelungen, die zusätzlichen Schwierigkeiten einer verteilten



Repositories befinden sich in einem Spannungsfeld unterschiedlicher Anforderungen: Sie müssen die Bibliotheksstandards bedienen, sich in die IT-Infrastruktur ihrer Institution einbetten und dem webversierten Nutzer gefallen.

Entwicklung organisatorisch zu bewältigen. Zugesagte Kapazitäten konnten beispielsweise – insbesondere nach der eigentlichen Förderphase – immer weniger eingehalten werden, da die Entwickler bei den Projektpartnern zu anderen Arbeiten abgezogen werden mussten. Das abschließende Release ist daher nicht verteilt entstanden, sondern gelang erst, nachdem die Entwicklung bei dem KOBV zusammengefasst worden war.

Schließlich muss auch überlegt werden, ob das LAMP-Paradigma der Aufgabe gewachsen war: Relationale Datenbanken wie MySQL stellen für die semistrukturierten Inhalte von Repositorien keine ideale Lösung mehr dar und auch PHP ist nicht das Mittel der Wahl, wenn leistungsfähige Frameworks und aktuelle Design-Patterns gefragt sind.

Festzustellen ist, dass die Nachfrage nach Repositorien bei kleineren Hochschulen und Instituten aktuell deutlich zunimmt; während sich die Universitäten bereits seit längerer Zeit versorgt haben, erreicht diese Einrichtungen dieser Bedarf erst jetzt in größerem Ausmaß. Daher ist zu erwarten, dass das Hosting in Zukunft noch an Bedeutung gewinnen wird, weil solche Einrichtungen für den eigenständigen Betrieb eines Repositoriums nicht ausgestattet sind.

Es wird sich dabei eine Ausdifferenzierung der funktionalen Anforderungen an die Repositorien ergeben: große, international vernetzte Universitäten werden an ihre Repositorien und deren Verbindung mit Forschungsplattform und Primärdaten andere Anforderungen stellen müssen als kleine Fachhochschulen, die sich erstmals auf das Feld der elektronischen Veröffentlichung wagen.

Vor dem Hintergrund der Einbettung der Repositorien in die Hochschulinfrastrukturen müssen diese ihre sehr monolithische, integrierte Struktur aufgeben und sich modularer aufstellen, so dass z.B. Publikation, Suche, Kollektionen und Präsentation in unterschiedlichsten Kontexten und unter verschiedensten Oberflächen auch getrennt voneinander eingesetzt werden können.

Das Bibliotheksservice-Zentrum Baden-Württemberg (BSZ) wird im Hosting institutionelle Repositorien auch weiterhin mit OPUS realisieren, und es wird dazu die Version 3 einsetzen, sofern keine besonderen Anforderungen eine Abwandlung der Datenmodelle erfordern. Falls eine solche Anpassung nachgefragt wird, wird es das Repository mit OPUS 4 aufbauen.

Zu beiden Versionen will das BSZ Informationen auf www.opus-repository.org zusammenführen. Zum einen soll dies die Auswahl zwischen beiden Varianten erleichtern, zum anderen soll das konzeptionelle Auseinanderdriften von OPUS gebremst und möglicherweise ein Ansatz für eine weitere Version geschaffen werden.

Angesichts der oben dargestellten Herausforderungen an Repositorien in Gegenwart und Zukunft ergeben sich offensichtlich noch Raum und Bedarf für weitere Versionen von OPUS.