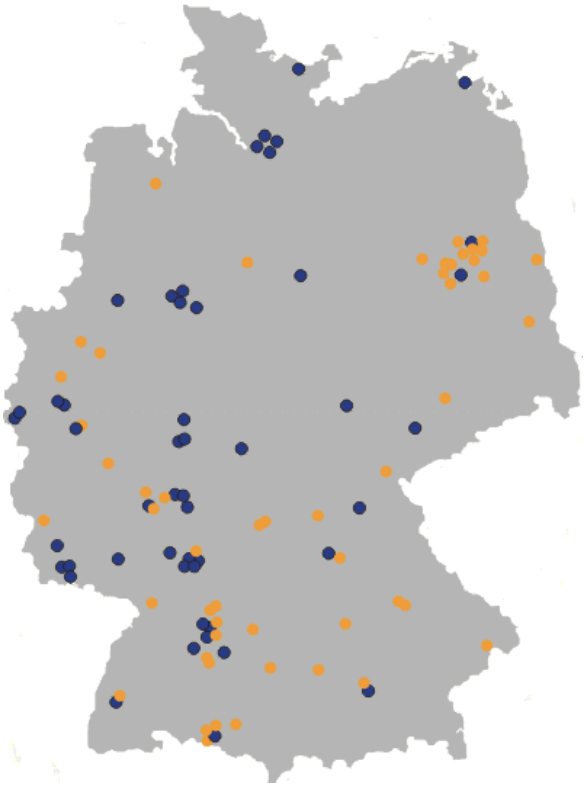
OPUS -
Community- und Softwareentwicklung als Arbeitsgebiet der Verbünde

Christof Mainberger (BSZ)

(Vortrag auf dem Bibliothekskongress in Leipzig am 16. März 2010)

Der Online-Publikationsverbund der Universität Stuttgart – besser bekannt unter dem Akronym OPUS – blickt bereits auf eine lange Geschichte zurück: 1997 als Projekt der Universitätsbibliothek und des Rechenzentrums Stuttgart begonnen, wurden bis zur Version OPUS 3.3 im Jahr 2009 insgesamt sieben Versionen dieser Software veröffentlicht.



Von den hundert OPUS-Installationen werden fünfzig (orange) bei Verbundzentralen betrieben.

Mittlerweile – im März 2010 – zählen wir in Deutschland hundert Installationen dieser Anwendung. Vergegenwärtigt man sich, dass die Anzahl der Institutionellen Repositorien derzeit in der Bundesrepublik auf insgesamt hundertfünfzig geschätzt wird, ist diese Zahl beachtlich.

OPUS ist für die Verbünde das Mittel der Wahl, wenn Bibliotheken sie nach einer Möglichkeit zum elektronischen Publizieren fragen: Sowohl der Bayerische Bibliotheksverbund BVB, das Bibliotheksservice-Zentrum Baden-Württemberg BSZ, das Hochschulbibliothekszentrum Nordrhein-Westfalen hbz als auch der Kooperative Bibliotheksverbund Berlin-Brandenburg KOBV bieten an, OPUS-Anwendungen für Bibliotheken zu installieren und zu betreiben – wobei die Installationen von OPUS-Bayern künftig vom KOBV übernommen werden. Fünfzig der hundert OPUS-Anwendungen werden derzeit in Verbundzentralen gehostet.

Daher erstaunt nicht, dass insbesondere Verbundzentralen an der Entwicklungsgemeinschaft zur OPUS 4 Entwicklung beteiligt sind. Dieses Projekt zur Entwicklung einer neuen Version wurde, gefördert von der Deutschen Forschungsgemeinschaft, 2008 von den Universitätsbibliotheken in Bielefeld, Hamburg-Harburg, Saarbrücken und Stuttgart sowie dem BSZ und dem KOBV begonnen; später traten noch das hbz und zeitweise die Staatsbibliothek Landes- und Universitätsbibliothek Dresden in die Kooperation ein.

Am 10. März 2010 wurde mit einem Entwickler-Release ein erstes Ergebnis des Projekts veröffentlicht. Entwickler-Release bezeichnet dabei eine noch unfertige Version, in der fast alle Funktionalitäten bereitstehen; diese sind allerdings noch nicht eingehend getestet und möglicherweise noch fehlerbehaftet. Außerdem wurden noch nicht alle Textelemente formuliert und übersetzt, Dokumentation und Hilfetexte fehlen noch.

Trotz der langen Geschichte und breiten Verbreitung von OPUS ist es erforderlich, seine wesentlichen Charakteristika deutlich zu machen: OPUS ist eine Anwendung zum Betrieb von institutionellen und fachlichen Repositorien. Zunehmend wird OPUS darüber hinaus auch für Hochschulbibliographien verwendet. OPUS dient insofern der Veröffentlichung, Erschließung, Administration, Recherche und Verbreitung elektronischer Publikationen; als Bibliographie eingesetzt, sind auch Einträge ohne beigefügten Volltext bzw. auch zu nicht elektronisch veröffentlichten Titeln möglich.

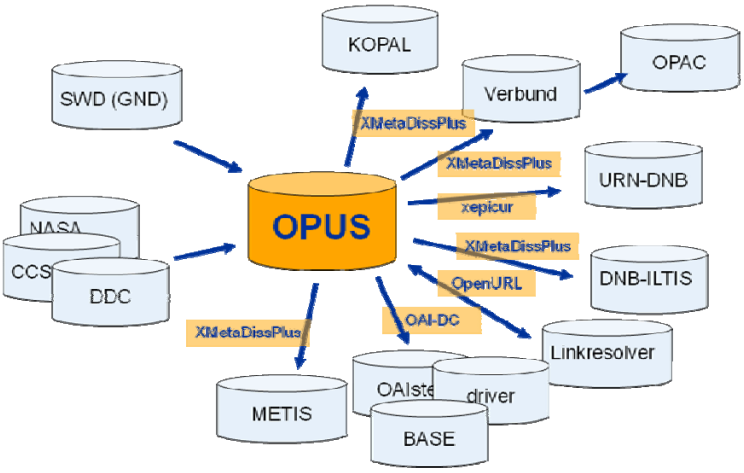
OPUS ist über eine Vielzahl standardisierter Schnittstellen in lokale und regionale Bibliotheksysteme und die nationalen und internationalen Netzwerke des Open-Access eingebettet. Umgekehrt lässt sich OPUS flexibel vom Betreiber anpassen und bietet eine Vielzahl Interaktionsmöglichkeiten mit nicht bibliothekarischen Anwendungen und für die Nutzer.

Nicht zuletzt ist OPUS Software, eine Webanwendung, die auf Open-Source-Komponenten wie z.B. Php5, MySQL, Apache beruht und selbst unter der GNU-Public-Lizenz als Open Source gestellt ist. Dabei wird Wert darauf gelegt, dass OPUS leicht zu installieren und mit geringem Aufwand technisch zu betreiben ist.

Eine zentrale Idee von institutionellen Repositorien wie OPUS liegt darin, dass Endnutzer selbst sowohl Publikationen als auch Metadaten zu diesen Publikationen in das System einpflegen. Dazu werden gegenüber den bibliothekarischen Formaten vereinfachte Masken angeboten. Trotz der Vereinfachung sollen die dabei erfassten Daten möglichst hohe Qualität aufweisen, um die Übertragung in die Verbundsysteme und Bibliothekskataloge zu erleichtern. Als ein Aspekt solcher Qualität erlaubt es OPUS Klassifikationen wie z.B. die DNB-Sachgruppen der DDC ins System zu importieren und bindet online die Schlagwortnormdatei (SWD) der Deutschen Nationalbibliothek ein. Analoges ist selbstverständlich auch für die GND geplant, sobald diese online vorliegt.

Für die Übertragung der Metadaten in andere bibliothekarische Systeme werden Standardformate verwendet. Für Suchmaschinen wie BASE oder OAIster wird oai-dc angeboten, Linkresolver werden über OpenURL angesprochen – umgekehrt kann auch OPUS per OpenURL eingebunden werden –, URNs werden via xepikur an die DNB gemeldet und Verbundsysteme sowie DNB-ILTIS werden mit XMetaDissPlus versorgt. Dieses Format wird auch für den Export aus OPUS in Systeme der Langzeitarchivierung verwendet und zum Transport der Zählpixel an die VG-Wort.

Die Verwendung von bibliothekarischem Vokabular und die Bedienung bibliothekarischer Schnittstellen integriert OPUS in die Systeme der wissenschaftlichen Informationsversorgung und erfüllt entsprechend auch eine Forderung des DINI-Zertifikats.



OPUS ist über eine Vielzahl standardisierter Schnittstellen in die Bibliotheksysteme und die nationalen und internationalen Netzwerke des Open-Access eingebettet.

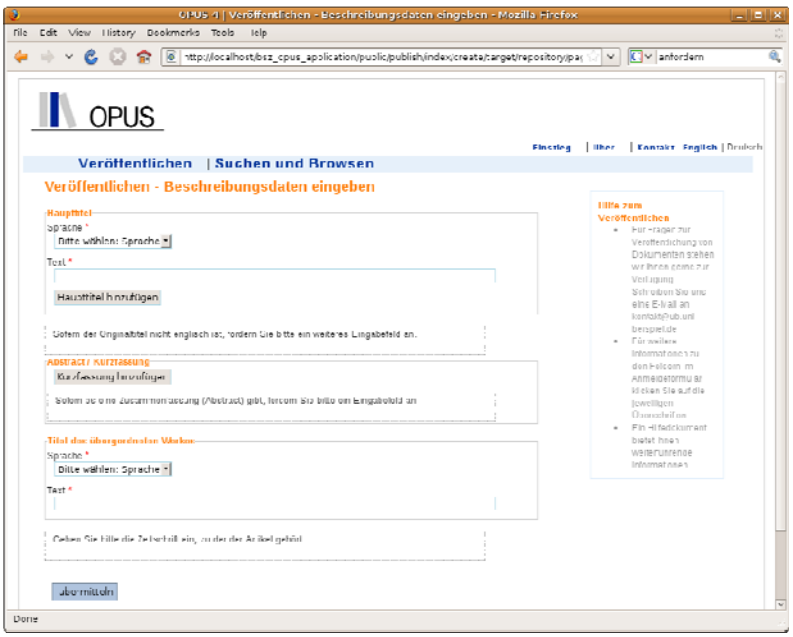
Zunehmend sind Repositorien allerdings mit der Anforderung konfrontiert, mit weitaus weniger standardisierten Systemen zu kommunizieren: Hier sind insbesondere Literaturverwaltungssysteme zu nen-

nen, in die Wissenschaftler Metadaten aus OPUS exportieren und aus denen sie umgekehrt Metadaten in die Masken importieren wollen. Immer häufiger werden Wege zur Verbindung mit Forschungsinformationssystemen gesucht, die z.B. andere Metadatenmodelle aufweisen. Auch das Gebiet der Primär- bzw. Forschungsdaten ist zu nennen, das gegenwärtig erst Ansätze zur Spezifikation der erforderlichen Schnittstellen und Formate aufweist. Google und das Web 2.0 steuern weitere, wenig stabile Schnittstellen bei. Über Bibliometrie sollen die in OPUS eingestellten Publikationen zur leistungsorientierten Mittelvergabe herangezogen werden. Schließlich fallen solche Anforderungen von Anwendung zu Anwendung oft stark unterschiedlich aus, z. B. gehen die Vorstellungen über eine Bibliographie doch von Anwender zu Anwender auseinander.

Mit institutionellen Repositorien wie OPUS reklamieren Bibliotheken die Aufgabe der Literaturverwaltung für sich, die scheinbar so viel einfacher im Content-Management der Hochschule, dem Arbeitsplatzrechner jedes Wissenschaftlers oder irgendwo im Web 2.0 erledigt werden kann. Im Gegenzug und weil das Ziel ist, den Endnutzer auch für die aktive Mitarbeit am Repositorium zu gewinnen, muss die Anwendung die Einfachheit und den Komfort nachzeichnen, die die Nutzer mittlerweile gewohnt und nicht mehr zu entbehren bereit sind. Andererseits sollen elektronische Publikationen ganz selbstverständlich in den Katalogen nachgewiesen sein und dort den entsprechenden Standards genügen – ein durch die Gegebenheiten geforderter veritabler Spagat.

OPUS muss also einerseits in die durch Standards geregelte Bibliothekswelt blicken und andererseits – gleichzeitig – die sehr dynamischen, vielfältigen und selten schon ausreichend spezifizierten Anforderungen seiner Nutzer respektieren. Wie meistert die neue OPUS 4 Version diese Spannung?

Diese Herausforderung beantwortet OPUS 4 insbesondere durch ein flexibles Datenmodell, das die Ausgestaltung der Publikationstypen in ihrer Feldzusammensetzung sowie die Definition zusätzlicher Publikationstypen nach dem lokalem Bedarf erlaubt. Dabei ist es möglich, aus einem Pool von Feldern auszuwählen und festzulegen, wie oft ein Datenelement wiederholt werden kann, bzw. ob es obligatorisch oder fakultativ ist. Jeder Publikationstyp kann unabhängig für das institutionelle Repositorium sowie die Bibliographie-Funktion definiert werden, so dass für die Bibliographie z.B. auf die Angabe eines Abstracts verzichtet werden kann, das im gleichen Publikationstyp für Hochschulschriften obligatorisch ist. Die Beschreibung erfolgt über XML-Dokumente, die einer einfachen Syntax folgen und vom Betreiber bearbeitet werden können.



Aus den Publikationstypdefinitionen werden automatisch, ggf. wiederholbare Eingabefelder für das Publikationsformular generiert.

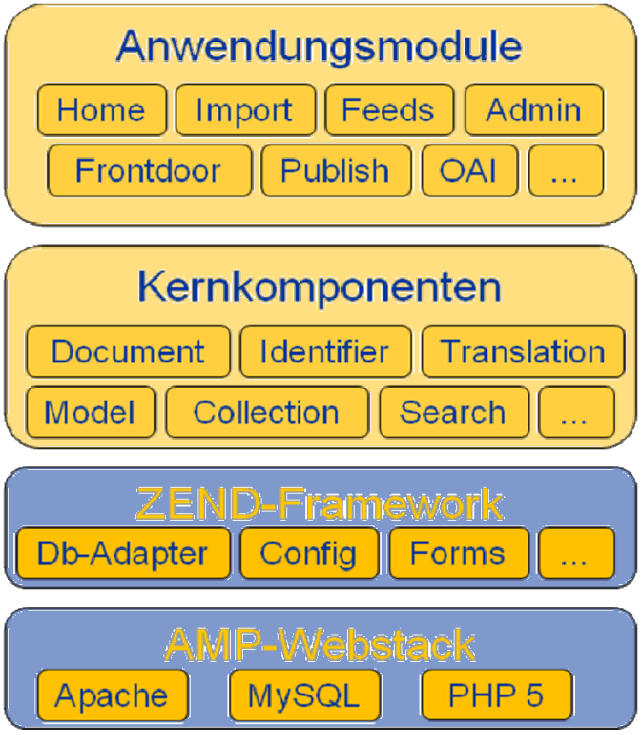
Aus den Publikationstyp-Definitionen generiert die Software automatisch sowohl die Anbindung an die unterlegte MySQL-Datenbank zur Metadatenhaltung und zum Lucene-Index,

der eine schnelle, in OPUS 4 integrierte Volltextsuche realisiert, aus den Publikationstyp-Definitionen werden ebenfalls die Formulare erzeugt, über die die Publikation bzw. die Administration in OPUS 4 erfolgt. Die Gestaltung der Frontdoor und der Exporte über die OAI-Schnittstelle werden über XSLT-Transformationen gesteuert, die je nach Publikationstyp angepasst werden müssen, insbesondere wenn lokale Erweiterungen des Datenmodells auf die bibliothekarischen Standardformate abzubilden sind.

Der zweite Bereich des OPUS 4 Datenmodells betrifft die Kollektionen, mit denen in einheitlicher Weise alle Zusammenstellungen von Publikationseinträgen in OPUS 4 verwaltet werden. Solche Kollektionen sind in hierarchische Strukturen von Ober- und Untermengen gegliedert. Kollektionen sind z. B. die organisatorischen Einheiten (Fakultäten, Lehrstühle, Sonderforschungsbereiche, etc.) der Hochschule, denen Publikationen zugeordnet sind, Kollektionen können Klassifikationen abbilden, mit denen die Titel notiert sind, Kollektionen sind Schriftenreihen und Zeitschriften mit Jahrgängen, Heften und eingeordneten Artikel oder auch ganz andere Kollektionstypen, die der Betreiber des Repositoriums festlegen und mit jeweils eigenen Attributen ausstatten kann. Zu jeder Kollektion kann auch ein eigenes Erscheinungsbild hinterlegt werden, so dass z.B. eine Schriftenreihe unter einer eigenen Oberfläche präsentiert werden kann.

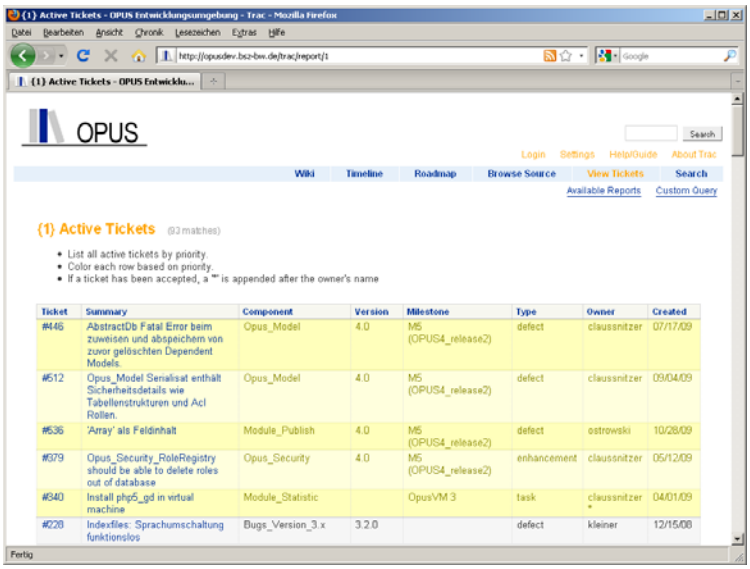
Die Gestaltung von OPUS 4 wird durch Templates gesteuert und kann mit Hilfe von CSS und der Anpassung von Bildern und Textressourcen sehr einfach und weitreichend abgewandelt werden. Alle verwendeten Textbausteine sind in allen vorgesehenen Sprachen in Sprachdateien ausgelagert, wo sie leicht an den konkreten Bedarf angepasst werden können.

OPUS 4 wurde objektorientiert programmiert und folgt gängigen Entwicklungsmustern für objektorientierte Web-Anwendungen. Die Softwarearchitektur definiert Schichten der Anwendung und verfolgt ein „Plugin-Konzept“, das eine Konfektion der Anwendung und – bei Bedarf – die Entwicklung weiterer Komponenten erleichtern soll. Die Basis bilden dabei die Open Source Komponenten Apache, MySQL und PHP. Darüber wird das gleichfalls freie Zend-Framework eingesetzt, das elementare Module, wie Formulare, Datenbankanbindung und Konfiguration bereitstellt. Diese werden vom Kern-Framework verwendet, welches wesentliche Komponenten für eine Repositorien-Software beinhaltet, wie z.B. Document, Search, Identifier etc. Schließlich realisiert eine Applikationsschicht Anwendungsmodule wie z.B. Publizieren, Administration, Import, OAI und Frontdoor. Alle eingesetzten – nicht selbst entwickelten Komponenten – sind Open Source, lizenzkompatibel und frei erhältlich.



Die Architektur von OPUS folgt den gängigen Mustern objektorientierter Softwareentwicklung.

Für die Entwicklung, die dezentral bei den Partnern der Entwicklungsgemeinschaft stattfand, wurde ein organisatorischer Rahmen geschaffen. Dieser legte die erforderlichen Werkzeuge bereit, z.B. eine auf Ubuntu-Basis angelegte Virtuelle Maschine mit Eclipse-IDE, so dass jedem Entwickler eine einheitliche Umgebung zur Verfügung stand. Ein Versionskontrollsystem nahm den kooperativ erstellten Code auf und eine Entwicklungsumgebung auf Trac-Basis stellte Bug Tracking, Wiki sowie Diskussionsforen zur Verfügung. Diese Plattform wird im BSZ betrieben und dauerhaft gepflegt. Zur Kommunikation zwischen den Entwicklern wurden Mailinglisten eingerichtet und wöchentlich Video- bzw. Telefonkonferenzen durchgeführt. An fünf Entwicklungsworkshops in Dresden und Stuttgart trafen sich alle Entwickler, um die Entwicklung eine Woche lang intensiv gemeinsam voranzutreiben. Schließlich vereinbarte man auch Softwareentwicklungsmethoden, z.B. Coding Standards, UnitTests und Sprints.



Die Entwicklungsumgebung Trac, die für OPUS eingesetzt wird, enthält SVN-Anbindung, BugTracking und Wiki.

Die OPUS 4 Entwicklung umfasst allerdings nicht nur Software-Entwicklung: XMetaDissPlus, das für viele der Standardschnittstellen von OPUS 4 die Daten transportiert, wurde am Rande der OPUS 4 Entwicklung überarbeitet und in die Verantwortung der DNB übergeben. Das Format war ursprünglich eine Ergänzung von XMetaDiss, mit dem die Ablieferung ausschließlich von Dissertationen und Habilitationen an die DNB vorgenommen wird. Das BSZ, das für die Übertragung der Metadaten aus OPUS in die Verbunddatenbank ein Format benötigte, ergänzte XMetaDiss um weitere Publikationstypen zum Format XMetaDissPlus. Die DNB benötigt nun ebenfalls für die elektronische Pflichtabgabe einen erweiterten Publikationstypsatz und greift daher auf diese Spezifikation zurück, wobei allerdings gemeinsam mit Experten des BSZ und DINI die Typbezeichnungen unter Beachtung der internationalen Entwicklung (Dublin Core, Driver) überarbeitet wurden. Das neue Format wird nun auch zur Set-Bildung im DINI-Zertifikat sowie von Suchmaschinen wie z.B. BASE genutzt.

Ein weiteres Beispiel für konzeptionelle Arbeiten im Kontext OPUS 4 stellt die Erarbeitung der Geschäftsgänge zur Meldung von Online-Publikationen und deren Nutzung an die VG Wort dar. Auch diese Spezifikationen werden von der OPUS-Entwicklungsgemeinschaft und DINI vorangetrieben. Auf Seiten des METIS-Projekts, in dem die VG Wort diese Verfahren ausgestalten will, konnten die Spezifikationen und der Aufbau der Infrastruktur noch nicht abgeschlossen werden.

XMetaDissPlus und METIS sind Beispiele dafür, dass oft OPUS der Anlass für die Erarbeitung neuer Standards bildet, die dann mit breiter Beteiligung und aufwendiger Abstimmung natürlich so angelegt werden, dass sie auch für andere Repositoriensoftware und Bibliothekssysteme funktionieren.

Rechnet man vom ursprünglich beabsichtigten Projektbeginn von OPUS 4 an, dann haben sich die Release-Termine für die Anwendung mittlerweile mehrfach und insgesamt über ein Jahr verschoben.

Dies gibt Anlass, sich doch Gedanken zu machen, ob in dem Projekt – über die regelmäßige und übliche Verspätung von Software-Projekten hinaus – nicht verschiedene Dinge unglücklich gelaufen sind.

Ein Faktor dabei war sicher, dass das Vorhaben zunächst auf eine Renovierung von OPUS 3.3 abgestellt war. Recht schnell stellte sich jedoch heraus, dass für die Nachhaltigkeit und Zukunftsfähigkeit der Entwicklung eine vollkommene Neukonzeption und Programmierung erforderlich ist. Mittlerweile ist kein Stein auf dem andern geblieben und die Software völlig neu implementiert. Es ist aber klar, dass mit Mitteln, die für einen punktuellen Ausbau kalkuliert sind, ein vollständiger Neubau sich mindestens verzögern muss.

Ein weiterer Moment bildet sicherlich die oben geschilderte Situation, dass die Software für wenig standardisierte und sich in dynamischer Veränderung befindliche Anforderungen ausgerichtet werden musste. Der Ausweg konnte lediglich in einer hohen Generizität und Flexibilität liegen, die weitere Komplexität und Aufwand verursachte.

Organisatorisch war das Projekt schon zu Beginn durch die Schwierigkeit belastet, geeignet qualifiziertes technisches Personal im Rahmen der kurzen Projektdauer und des Tarifvertrags der Länder zu finden. Dadurch musste der Projektstart um mehrere Monate verschoben werden. Trotz dem Einsatz der Entwicklungsplattform und moderner Kommunikationsmittel hat sich auch die bundesweit verteilte Ansiedlung von Entwicklern als zusätzliche Herausforderung gezeigt.

Technisch wurde OPUS 4 mit modernsten Konzepten realisiert, die zumindest im verwendeten PHP-Umfeld derzeit noch ungewöhnlich sind und die daher momentan nur ausgewiesene Spezialisten beherrschen. Dies erschwerte die Rekrutierung weiterer Entwicklungskapazität über das Projekt und es steht zu hoffen, dass sich Objektorientierung, Entwurfsmuster und Frameworks auch im PHP-Umfeld durchsetzen, so dass diese Entscheidung keine Hürde für der Nutzung und Weiterentwicklung der Software darstellt.

Nach der Veröffentlichung des Entwicklungs-Release soll in den nächsten Monaten die Anwendung fertiggestellt werden. Dazu sind eingehende System- und Usabilitytests erforderlich sowie die Erstellung von Dokumentation, Hilfetexten und Übersetzungen. Erste Pilotinstallationen sollen im Frühsommer in Berlin, Hamburg und Konstanz entstehen und werden möglicherweise noch diverse Kinderkrankheiten zu Tage fördern. Es ist daher schwer abzuschätzen, wann ein Release möglich sein wird.

Als OPUS-Geschäftsstelle wird das BSZ diese Arbeiten weiter koordinieren und parallel die Community der OPUS-Anwender pflegen. Im Sommer 2010 wird dazu das reguläre OPUS-Anwendertreffen ausgerichtet, das alle eineinhalb Jahre stattfindet. Als gemeinsame Plattform wird opus-repository.org weiter ausgebaut. Sobald OPUS 4 released ist, sollen Workshops zu Installation, Konfiguration und Administration der neuen Software organisiert werden. Eine wesentliche Aufgabe der Geschäftsstelle wird auch sein, die Entwicklergemeinschaft, die sich für OPUS 4 gebildet hat, zu stabilisieren und damit auch weiterhin die Fortentwicklung zu gewährleisten.