

WebUIS 3.0

Empfehlungen für eine zukunftsfähige Neuausrichtung der webbasierten Informationssysteme des UIS Baden-Württemberg

*C. Döpmeier; C. Greceanu; T. Schlachter; C. Schmitt; R. Weidemann
Karlsruher Institut für Technologie
Institut für Angewandte Informatik
Hermann-von-Helmholtz-Platz 1
76344 Eggenstein-Leopoldshafen*

*F. Chaves; U. Bügel; J. Moßgraber; B. Schnebel; T. Usländer
Fraunhofer Institut für Optronik, Systemtechnik und Bildauswertung
Fraunhoferstr. 1
76131 Karlsruhe*

*W. Schillinger; M. Tauber; B. Nonnenmann; A. Koch; E. Schöpflin-Reichmann
LUBW Landesanstalt für Umwelt, Messungen und Naturschutz Baden-Württemberg
Griesbachstr. 1
76185 Karlsruhe*

*K. Zetzmann; R. Rossi
Ministerium für Umwelt, Klima und Energiewirtschaft Baden-Württemberg
Kernerplatz 9
70182 Stuttgart*

1. EINLEITUNG	11
2. AUSGANGSSITUATION	11
3. KONZEPTE UND EMPFEHLUNGEN FÜR EINE NEUAUSRICHTUNG	13
3.1 GRUNDLEGENDE ARCHITEKTUR	14
3.2 EMPFEHLUNGEN ZU ENTWICKLUNGSPLATTFORMEN UND LAUFZEITUMGEBUNGEN FÜR WEBANWENDUNGEN	16
4. ERSTE ERFAHRUNGEN IN PILOTPROJEKTEN	17
5. FAZIT UND AUSBLICK	18
6. LITERATUR.....	18

1. Einleitung

Das Ministerium für Umwelt, Klima und Energiewirtschaft Baden-Württemberg (UM) und die LUBW Landesanstalt für Umwelt, Messungen und Naturschutz Baden-Württemberg betreiben bereits seit 1996 Webangebote für das Umweltinformationssystem Baden-Württemberg (UIS BW), um Umweltinformationen auf wirtschaftliche Weise für die Öffentlichkeit zur Verfügung zu stellen.

Als Content-Management-System (CMS) wird bzw. wurde seit 2004 im Geschäftsbereich die Entwicklungsplattform WebGenesis vom Fraunhofer Institut für Optronik, Systemtechnik und Bildauswertung (IOSB) als technische Grundlage der Webpräsentationen der LUBW, des UM und vieler weiterer Fach- und Unterangebote eingesetzt.

Im Rahmen einer 2013 vom Karlsruher Institut für Technologie (KIT) und Fraunhofer IOSB unter Beteiligung der LUBW und des UM erstellten Studie *Web-UIS 3.0* zur Zukunftsfähigkeit der webbasierten Informationssysteme im Umweltinformationssystem Baden-Württemberg (UIS BW) /1/ wurde festgestellt, dass die damals bestehenden Systemversionen den Anforderungen heutiger Webanwendungen nicht mehr genügten und die Entwicklungsplattform in der damals vorhandenen aktuellsten Version nicht über alle Funktionalitäten verfügte, die man zur Modernisierung der existierenden Systeme benötigt.

Daher wurde als zentraler Teil der Web-UIS-Studie eine Konzeption erarbeitet, die als Grundlage für die Entwicklung des UIS BW hin zu einem modernen und zukunftsfähigen Internet-basierten Dienstleistungsangebot dient. Umweltinformationen werden künftig zunehmend nicht mehr primär für einen rein Browser-basierten Zugriff über Desktop-Computer bereitgestellt, sondern auch mobilen und anderen Anwendungen offene Schnittstellen zum Zugriff auf die Informationen bieten. Ein Kernaspekt ist dabei, dass alle Informationen bzw. Daten auf Dauer über offene serviceorientierte Schnittstellen modular und feingranular anderen Anwendungen zur gegenseitigen Nutzung bereitgestellt werden. Als weitere zentrale Ergebnisse beschreibt die Studie Konzepte, mit deren Hilfe Webanwendungen zukünftig wesentlich modularer und flexibler entwickelt und genutzt werden können.

In diesem Beitrag werden die wesentlichen Ergebnisse der Web-UIS 3.0-Studie vorgestellt. Hierbei werden zunächst die Ergebnisse der Evaluation der bestehenden Systeme sowie der Anforderungsanalyse dargestellt. Danach werden die wesentlichen Aspekte des Zukunftskonzeptes sowie die Empfehlungen zur Programmier- und Laufzeitumgebung erläutert. Im Anschluss werden erste Ergebnisse aus Pilotprojekten zur Umsetzung und Evaluation der zentralen Aspekte des Konzeptes vorgestellt, bevor Fazit und Ausblick den Beitrag beschließen.

2. Ausgangssituation

Zur Analyse der bestehenden Landschaft der webbasierten Informationssysteme des UIS BW wurde 2013 eine Bestandsaufnahme der vorhandenen Systeme durchgeführt, d.h. die durch sie bereitgestellten Funktionalitäten und ihre technische Implementierung sowie verwendete Programmiersprachen und Grundsoftware (Frameworks) identifiziert. Dabei stellte sich heraus, dass vornehmlich Java-basierte Webanwendungen unter Verwendung von Teilkomponenten des Java Enterprise (JEE) Standards im (Produktions-)Betrieb genutzt wurden, die technisch jedoch auf älteren Versionen der eingesetzten Frameworks und Stan-

dards basierten und daher oftmals technologisch veraltet waren. Viele Systeme nutzten bzw. nutzen dabei noch als Programmier- und Laufzeitplattform die JEE-basierte Entwicklungsplattform WebGenesis.

Ein Vergleich der Systeme mit dem aktuellen „State of the Art“, der ebenfalls für die Web-UIS-Studie erhoben wurde und in dieser detailliert beschrieben ist, zeigte weiter, dass die Implementierung vieler Funktionalitäten in den bestehenden Systemen sowohl von den Bedienkonzepten als auch in Bezug auf ihre Implementierung ebenfalls veraltet sind.

Wesentliche Ergebnisse der Analyse sind u.a.:

- Veraltete, nicht-HTML5-konforme Weboberflächen, die nur für Webbrowser auf Desktoprechnern konzipiert sind und keine Unterstützung für mobile Endgeräte bieten
- Aus heutiger Sicht umständliche Bedienkonzepte, die die interaktiven Fähigkeiten moderner AJAX-basierter Weboberflächen nicht nutzen
- Webseiten lassen sich von Autoren nicht flexibel aus vorhandenen Inhaltselementen und funktionalen Bausteinen modular zusammensetzen. Insbesondere die Wiederverwendbarkeit von Inhalten ist dabei oft nicht gegeben. Dies limitiert die Flexibilität der Autoren, erzeugt Redundanzen oder Mehraufwand für Autoren.
- Generell vermissten Autoren eine Reihe von Standard-CMS-Funktionalitäten
- Fehlende Wiederverwendbarkeit von Funktionalitäten auf Programmiererebene: Jede Webanwendung ist ein Monolith, dessen Funktionalitäten sich nur schwer in anderen Anwendungen wiederverwenden lassen.
- Mangelnde Interoperabilität der Webanwendungen untereinander sowie fehlende standardisierte, serviceorientierte Schnittstellen zur flexiblen Verknüpfung von Systemen
- Bei der Analyse der Möglichkeiten zur Weiterentwicklung von Anwendungen zeigte sich, dass die verwendeten Systemversionen der Entwicklungsplattform bzgl. der eingesetzten JEE-Grundtechnologien veraltet waren und Grundfunktionalitäten, wie z.B. die Integration moderner AJAX-basierter Frameworks zur Weboberflächenprogrammierung, vermissen ließen.

Weiter zeigte eine Anforderungsanalyse unter Endnutzern, dass den existierenden Systemen und benutzten Programmierplattformen eine Reihe von Grundfunktionalitäten fehlten, um bestimmte Anforderungen der Nutzer, die bei anderen vergleichbaren Systemen längst zur Standardausstattung gehören, ohne größere Programmierarbeiten „out-of-the-box“ erfüllen zu können. Hier wurden von den Benutzern u.a. die folgenden Funktionalitäten genannt:

- Flexibles Benutzer- und Gruppenverwaltungssystem mit Anbindbarkeit an andere Systeme (z.B. für Single Sign On), komfortable Rollen-basierte Rechteverwaltung
- Trennung von Autoren- und Produktionssystemen mit automatischer Synchronisation von Inhalten (Staging), einer Versionierung von Inhalten und Möglichkeiten zur Definition von redaktionellen Workflows
- Komfortable modularisierte Verwaltung von beliebigen binären und Text-basierten Inhalten unabhängig von Webseiten, die diese Inhalte nutzen (Dokument- bzw. Assetmanagement); einfache, modularisierte und flexible Integration von solchen Inhalten in Webseiten unter Berücksichtigung der Wiederverwendbarkeit von Inhalten
- Komfortable Formular-gestützte Erstellung von Inhalten

- Standardmodule zur Darstellung und Präsentation von Inhalten z.B. in Form von aggregierten Blogseiten, Suchergebnissen, RSS- oder Media-Feeds, Slideshows oder anderer Formen von Mediengalerien, Teaser, News-Scroller, etc.
- Integration von Kalender- und Terminmanagement
- Integrierbarkeit von Inhalten aus sozialen Medien oder Netzwerken, wie Videos aus YouTube, Tweets von Twitter oder Anbindung an Facebook
- Bereitstellung von Elementen zur Suchmaschinenoptimierung (SEO), wie sprechende URLs oder maschinenlesbare Sitemaps
- Standardmodule zur Einbindung und Visualisierung von Daten innerhalb der Websysteme, wie z.B. Karten mit Fachlayern, Module zur Darstellung von Daten als Diagramme

In Bezug auf eine effiziente Nutzung einer Grund-IT-Infrastruktur zur Programmierung von modernen Webanwendungen zeigte die Analyse auch, dass hierzu die Unterstützung der folgenden Funktionalitäten als Grundbestandteile der verwendeten Infrastruktur bis zu dem Zeitpunkt der Studie nicht vorhanden war:

- Durchgängige Unterstützung von HTML5 und responsivem Design (z.B. durch Integration von Frameworks wie Bootstrap) sowohl bei der Darstellung von Webseiteneinhalten als auch bei der Programmierung von Design-Templates (Unterstützung für responsive Corporate Design-Vorlagen)
- Bereitstellung von Möglichkeiten sowie Kompatibilität der Grundinfrastruktur zur Verwendung von modernen UI-Bibliotheken für die Programmierung webbasierter Benutzerschnittstellen auf Basis von weitverbreiteten JavaScript-Frameworks
- Möglichkeiten, in sich geschlossene Funktionalitäten für Nutzer als wiederverwendbare Bausteine (z.B. Portlets oder Web-Widgets) bereitzustellen, die im laufenden Betrieb eines Systems installiert und von Autoren im Anschluss ohne Programmierkenntnisse genutzt werden können
- Durchgängige modularisierte Bereitstellung von Inhalten und Daten über serviceorientierte Programmierschnittstellen (REST-basierte Services) sowie die Nutzung solcher Services durch andere Websysteme und Anwendungen

Auf Basis der Erkenntnisse aus der Analyse wurde dann im Rahmen der Web-UIS 3.0-Studie ein Zukunftskonzept erarbeitet, wie eine zukünftige webbasierte Infrastruktur des UIS BW bzgl. ihrer verteilten IT-Architektur aufgebaut werden sollte, welche Grundinfrastrukturen hierfür nötig sind und welche Empfehlungen bei der Programmierung zukünftiger Webanwendungen beachtet werden sollen, um den aktuellen Anforderungen der Nutzer gerecht zu werden.

Die wesentlichen Aspekte dieses Zukunftskonzeptes werden im nächsten Kapitel genauer beschrieben.

3. Konzepte und Empfehlungen für eine Neuausrichtung

Das in der Web-UIS 3.0 Studie ausgearbeitete Konzept setzt auf eine konsequente Weiterentwicklung des UIS BW hin zu einer lose gekoppelten serviceorientierten, webbasierten Dienstleistungsinfrastruktur mit dazugehörigen Web- und mobilen Anwendungen, bei der nicht nur rein informationelle Angebote für die Öffentlichkeit über Services und Webanwen-

dungen bereitgestellt werden, sondern Daten und auch Fachanwendungslogik im Sinne einer serviceorientierten Softwareinfrastruktur als Dienstleistung universell verfügbar gemacht werden, damit diese dann in verschiedenen Anwendungen genutzt werden können. Sowohl unstrukturierte Informationsbausteine, wie HTML-Beschreibungstexte für Webseiten, als auch binäre Dokumente, Bilder, andere Medieninhalte oder sogar einzelne Datensätze (z.B. aktuelle Messdaten an einem bestimmten Standort) sollten auf Dauer feingranular über Services als Ressourcen bereitgestellt und in Anwendungen über Services abgerufen und flexibel verknüpft werden können.

Dies reduziert nicht nur die Redundanz solcher Informationsbausteine bzw. von Fachlogik in Anwendungen und damit deren Entwicklungskosten, sondern ermöglicht auch die einfache Wiederverwendung in ganz unterschiedlichen Frontendanwendungen, wie mobilen Apps oder Webanwendungen in unterschiedlichen Websystemen, wie CMS, Portalen oder Fachanwendungen. Die in bestimmten Situationen sinnvolle Nutzung externer Dienste, z.B. von Cloud-basierten Dienstangeboten wie der Google Maps Engine [2], reduziert dabei ebenfalls wesentlich Entwicklungskosten, spart Entwicklungszeit und bringt andere Vorteile, wie eine permanente Verfügbarkeit von Internet-Datendiensten, ohne dass die eigene Rechnerinfrastruktur ständig zur Verfügung stehen muss.

Der Vorschlag in der genannten Studie setzt daher auf eine durchgehende serviceorientierte Architektur für das Web-UIS 3.0.

3.1 Grundlegende Architektur

Diese Architektur sieht vor, dass an Stelle einer Vielzahl von nicht miteinander verknüpften Webanwendungen eine serviceorientierte Dienstleistungsinfrastruktur von lose koppelbaren Grunddiensten und internen webbasierten Fachanwendungen mit Dienst-orientierten Schnittstellen aufgebaut wird, auf die von Frontend-Webanwendungen, wie CMS und Portalen, die integrierte Zugänge zu Informationen und fachlichen Anwendungen für bestimmte Zielgruppen bereitstellen, zugegriffen wird (siehe Abbildung 1).

Hierbei ist die gesamte entstehende Softwareinfrastruktur durchaus als logische Einheit zu sehen, für die auf Dauer ein Gesamtkonzept aufeinander abgestimmter Dienste und diese nutzender Anwendungen entworfen werden muss. Alle Informationen und Daten sollten dabei möglichst modular und feingranular von den Services als eigenständig nutzbare Dienstleistungen angeboten werden (SOA-Prinzip), jedoch durch Harmonisierung und Abgleich der Informationsmodelle über die einzelnen Dienste hinweg von konsumierenden Anwendungen sinnvoll verknüpft werden können. Dies lässt sich z.B. durch Linked-Data-Ansätze, wie sie im semantischen Web angewendet werden, auch technisch instrumentieren.

Die (UIS-)internen Services lassen sich an Stellen, wo dies sinnvoll ist, durch externe über das Internet bereitgestellte Dienstleistungsangebote oder durch kommerzielle hinzukaufbare Produkte, die z.B. im Intranet bereitgestellt werden können, ergänzen (wie z.B. der Google Search Appliance als Suchmaschine). Dies ist beispielsweise dann sinnvoll, wenn die Eigenentwicklung eines solchen Dienstes im Intranet zu aufwändig wäre oder wenn der Betrieb einer Software innerhalb der eigenen Infrastruktur mit gewünschter Quality of Service nicht möglich ist.

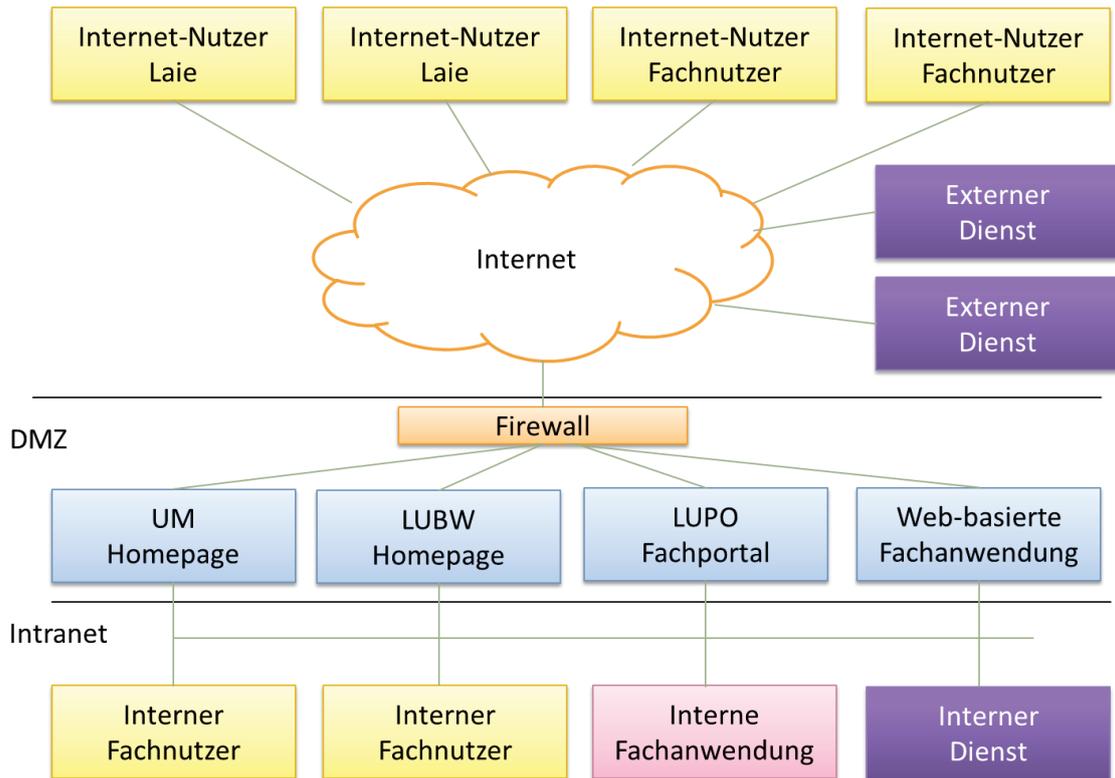


Abbildung 1: Zusammenspiel einzelner Systeme im zukünftigen Web-UIS

Aufgabe von Fachanwendungen und Frontend-Webanwendungen als Nutzerschnittstelle ist es, Informationen und Daten im jeweiligen Nutzungskontext sinnvoll zu verknüpfen und dem Nutzer ergonomisch für die jeweiligen Nutzungszwecke integriert anzubieten. Dabei wird eine Fachanwendung den Nutzern eine detailliertere und funktional reichere Nutzungsoberfläche anbieten, mit der die Daten oftmals nicht nur recherchiert, sondern auch editiert werden können, während sich Web-Frontendanwendungen für den allgemeinen Nutzer hier eher auf übersichtsartige Darstellungen der Informationen konzentrieren und den Nutzer bei Bedarf an Fachinformationssysteme weiterleiten, die detailliertere Informationen bieten. Aber auch auf Ebene der Web- und Fachanwendungen sollte die angebotene Funktionalität modular und wiederverwendbar implementiert sein, so dass sich Anwendungsmodulare in verschiedenen Anwendungen gemeinsam nutzen lassen. Weiter sollten diese Module die Nutzung durch verschiedene Clients, Desktop-Browser, mobile Geräte oder in mobilen Anwendungen erlauben.

Eine solche modularisierte Struktur von Webanwendungen lässt sich z.B. durch Technologien wie Web-Widgets und Portlets erreichen, wie sie in Portalservern und Content Management Systemen als Laufzeit- und Entwicklungsplattformen für Webanwendungen häufig verwendet werden. Solche Entwicklungsplattformen für Webanwendungen mit Portalcharakter können dabei alle wesentlichen Daten und Funktionalitäten, die Nutzer benötigen, nicht nur über Widgets oder Portlets für den Nutzer bereitstellen, sondern für den Zugriff auf die Daten auch benötigte Hilfsdienstleistungen (z.B. Proxy und Autorisierungshilfen) bereitstellen oder eigene Funktionalitäten wiederum selbst als Services anbieten. Durch die angebotenen Services können die Portale dann wiederum von externen Anwendungen und mobilen Applikationen genutzt werden, die den Anwendern ihrerseits Daten und Funktionalitäten (ggf. mit einem Mehrwert) zur Verfügung stellen. Auf diese Weise können von Anwendern nicht nur Webbrowser, sondern auch native Anwendungen zum Zugang zu Daten verwendet werden.

3.2 Empfehlungen zu Entwicklungsplattformen und Laufzeitumgebungen für Webanwendungen

Um die Wiederverwendbarkeit von Funktionalitäten über verschiedene Webanwendungen hinweg zu erreichen, sollten die Fachanwendungen und Frontend-Webanwendungen gemäß den Empfehlungen der o.g. Studie nach Möglichkeit auf einigen wenigen technischen Grundinfrastrukturen basieren. Für komplexere Anwendungen bieten dabei Java-basierte Webanwendungsinfrastrukturen, wie Portalserver, die auf dem Java Enterprise (JEE) Standard basieren, optimale Bedingungen, da der JEE-Standard einerseits sehr viele für die Programmierung von modernen Webanwendungen benötigte Grundfunktionalitäten bereits mit sich bringt und zusätzlich eine Vielzahl qualitativ hochwertiger Open-Source-Frameworks diese Funktionalitäten ergänzen, so dass sich sämtliche Anwendungsbereiche im UIS BW damit gut abdecken lassen (siehe Abbildung 2).

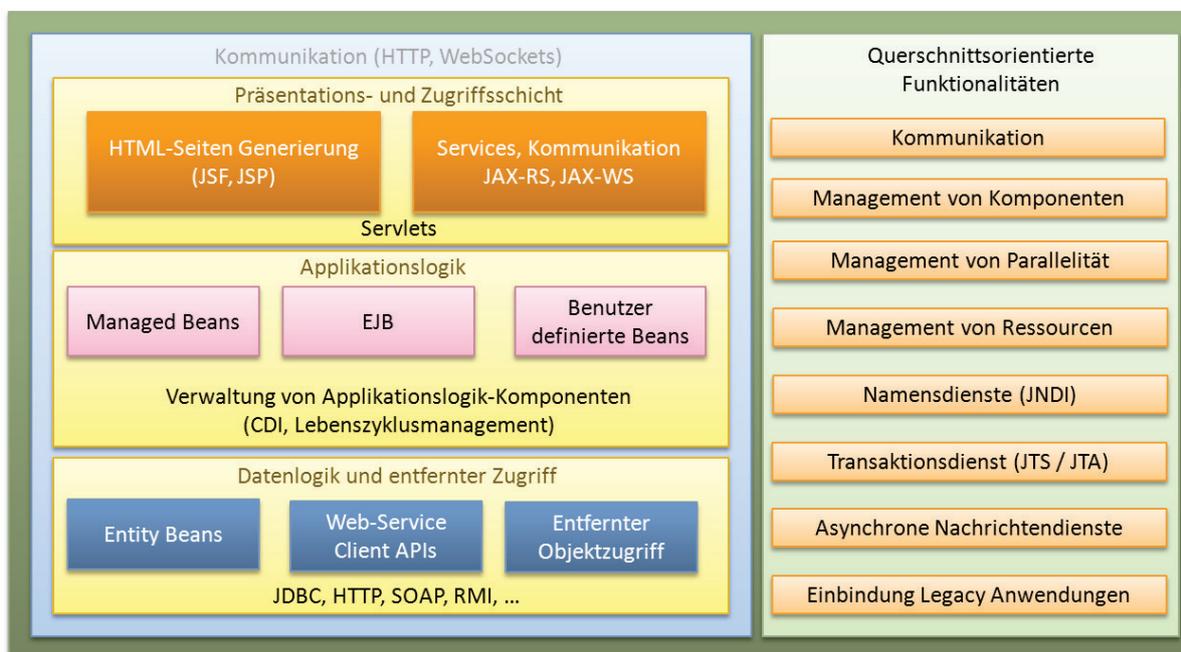


Abbildung 2: Die Grundfunktionalitäten einer JEE-Entwicklungsumgebung in der Übersicht

Da Java-basierte Portalserver bereits Konzepte für die Modularisierung der Funktionalitäten als eigenständige Komponenten (funktionale Erweiterungen, Portlets, Web-Widgets), die sich zur Laufzeit im System installieren bzw. de-installieren lassen, mitbringen, lassen sich neue Funktionalitäten auf deren Basis modular und wiederverwendbar realisieren und für Autoren und Benutzer flexibel und ergonomisch in Webseiten anwenden. Dabei bieten solche Portalserver schon eine ganze Reihe der komplexeren Funktionalitäten an, die von Systembetreibern für größere Anwendungen benötigt werden, z.B. Möglichkeiten für Single Sign On, Versionierung von Inhalten, Staging, etc. Auch viele gängige CMS-Funktionalitäten (siehe Anforderungen) werden von diesen Systemen durch vorhandene Grundmodule oder Erweiterungen bereits bedient.

Für weniger komplexe Webanwendungen, wie reine Homepagesysteme, die im Wesentlichen öffentlichkeitsorientierte textuelle und mediale Inhalte bereitstellen, lassen sich auch gängige Content-Management-Systeme, z.B. auf Basis von PHP, nutzen. Diese implementieren in der Regel ebenfalls ein Erweiterungskonzept, mit dem sich das System durch zusätzliche Komponenten erweitern lässt. Dabei gibt es für Systeme wie TYPO3, Drupal™

oder Joomla!® bereits eine Fülle von Erweiterungen, mit denen sich viele Grundfunktionalitäten „out of the box“ bedienen lassen. Für die Programmierung komplexerer Fachlogik sind diese PHP-Systeme aber eher weniger geeignet.

Unter Nutzung rein auf Webtechnologien basierender Programmierkonzepte (typischerweise JavaScript-basiert) lassen sich auch sehr modulare, wiederverwendbare funktionale Komponenten für webbasierte Anwendungen schreiben, die sowohl in Java-basierten Portalservern als auch in PHP-basierten Webanwendungen auf einfache Art und Weise genutzt werden können, sogenannte Web-Widgets. Daher empfiehlt die Studie die Verwendung von Web-Widgets in Anwendungssituationen, in denen eine möglichst große Wiederverwendbarkeit von webbasierten UI-Funktionalitäten gegeben sein soll. Diesen Ansatz greifen u.a. die Pilotprojekte zur Machbarkeitsanalyse der vorgestellten Konzepte auf, die im Folgenden kurz vorgestellt werden.

4. Erste Erfahrungen in Pilotprojekten

Um die Aussagen der Web-UIS 3.0-Studie zu überprüfen, wurden für das UIS BW einige Pilotprojekte aufgesetzt, in denen die wichtigsten Teilaspekte des Konzeptes prototypisch umgesetzt und an realen Anwendungen verifiziert werden sollen.

Eines der zentralen Projekte ist hierbei das länderübergreifende Projekt LUPO. In diesem soll prototypisch das Landesumweltportal Baden-Württemberg auf eine zeitgemäße Java-basierte Portalentwicklungsplattform (Liferay Portal) portiert und hierbei durchgängig modular und serviceorientiert gemäß der Web-UIS 3.0-Studie reimplementiert werden. Bereits an den ersten Versionen des Prototyps zeigte sich, dass die dabei verwendete Liferay-Plattform sowie die Umsetzung der Modularisierung der Anwendungsfunktionalitäten unter Verwendung von Services und Widgets den Programmierern völlig neue Möglichkeiten eröffnete, komplexe Informationen innerhalb des Portals dynamisch zu Webseiten zu aggregieren. Dabei lassen sich mit den entwickelten Konzepten für den Endnutzer Funktionalitäten implementieren, die die Benutzerfreundlichkeit des Umweltportals erheblich steigern. Nach einer ersten Demonstration des Prototyps vor Vertretern der an der LUPO-Kooperation beteiligten Bundesländer haben mittlerweile alle am LUPO-Entwicklungsverbund teilnehmenden Bundesländer den Wunsch geäußert, dass ihre Systeme mittelfristig auf die neue Plattform umgestellt werden sollen, da die Vorteile des neuen Ansatzes offensichtlich sind. Eine genauere Beschreibung der Liferay-Portalsoftware und der Pilotprojekte, die diese Software für ihren Einsatz im Land Baden-Württemberg testen, findet sich in einem weiteren Beitrag dieses Berichtes /3/.

Eine detailliertere Beschreibung der in dem Portal-Prototyp umgesetzten Modularisierungskonzepte auf Basis von Web-Widgets findet sich ebenfalls in einem Beitrag dieses Berichts /4/. Dieser detailliert nicht nur das im Liferay-Prototyp umgesetzte Widget-Konzept, sondern zeigt auch auf, wie solche Widgets in ähnlicher Form bereits im Umweltportal Nordrhein-Westfalen umgesetzt und erprobt wurden, das zurzeit noch mit WebGenesis betrieben wird. Dieser Fakt verdeutlicht, dass sich Widgets in beliebigen Webanwendungen nutzen lassen und nicht unbedingt eine (bestimmte) Portalserverinfrastruktur benötigen.

In einem weiteren Pilotprojekt wird ein serviceorientiertes Backend-System für die mobile App „Meine Umwelt“ entwickelt, das ebenfalls mit Liferay als Entwicklungsplattform realisiert wird /5/. Ziel ist es hierbei, modularisierte Backend-Dienste für die Administration der Datenquellen und -senken der App zu entwickeln, auf die Fachleute der Umweltbehörden, aber

auch Endnutzer der App über das Web zugreifen können. Dabei wird darauf geachtet, dass sich in Bezug auf die Wiederverwendung und Modularisierung der Funktionalitäten eine möglichst große Synergie zwischen den Projekten LUPO und „Meine Umwelt“ ergibt. Beide Systeme sollen dieselben Komponenten und Datendienste nutzen können und (wo möglich) gemeinsame Web-Widgets verwenden.

Im Rahmen der Evaluation der Nutzung von Cloud-Diensten werden schließlich sowohl im LUPO-Projekt als auch für die „Meine Umwelt“-App auf der Google Cloud bereitgestellte Datendienste genutzt /2/. So wurden z.B. zur Darstellung von Karten in den Portalen ein Kartenservice und ein zugehöriges Web-Widget entwickelt, mit dessen Hilfe beliebig komplexe Karten bestehend aus Layern, die z.B. über die Google Maps Engine oder als Fusiontables angeboten werden, in Portalen angezeigt werden können. Diese Lösung hat ebenfalls bereits gezeigt, dass die Nutzung von externen Cloud-Diensten nicht nur eine sehr gute Performanz und hohe Verfügbarkeit bietet, sondern in Kombination mit dem Widget-Konzept eine schnelle Implementierung modularer UI-Komponenten zur Integration in Webanwendungen bietet, über die darin Daten flexibel und hochmodern dargestellt werden können.

5. Fazit und Ausblick

Die Pilotprojekte zur Evaluation der Konzepte der Web-UIS 3.0-Studie haben bereits gezeigt, dass die zentralen Konzepte der Studie mehr als tragfähig sind, und eine wesentlich schnellere, innovativere und zukunftsfähige Entwicklung von UIS-Webanwendungen gemäß dem gewollten modularen Ansatz ermöglichen. Auch die für die Pilotimplementierungen verwendete Liferay-Plattform hat dabei in den Pilotprojekten bereits ihre Einsatztauglichkeit bewiesen und gezeigt, dass man auf ihrer Basis auch hochkomplexe Anwendungssituationen schnell und einfach implementieren kann. Daher scheint zu diesem Zeitpunkt eine vollständige Migration der auf dem LUPO-Baukasten basierenden Landesumweltportale auf die Liferay-Portalplattform nur noch eine Frage der Zeit zu sein. Diese Migration wird dabei in Zukunft das Potential dieser Portalanwendungen weiter erschließen und bietet auch eine gute Vorlage, um weitere UIS-Webanwendungen, wie z.B. die Homepage der LUBW, dann ebenfalls auf die neue Plattform umzustellen.

6. Literatur

- /1/ Döpmeier, C. et al. (2013): Konzeption Web-UIS 3.0. Interner Bericht, http://www.fachdokumente.lubw.baden-wuerttemberg.de/content/107754/Konzept-Webuis-3_0_v10_20130521.pdf.
- /2/ Schlachter T. et al. (2014): Cloud-Dienste – Erste Ergebnisse der Evaluierung von Cloud-Diensten für das UIS Baden-Württemberg. In diesem Bericht.
- /3/ Döpmeier, C. et al. (2014): Portalplattform Liferay – Erprobung neuer Portaltechnologien für E-Government-Portale der Landesverwaltung Baden-Württemberg. In diesem Bericht.
- /4/ Schlachter T. et al. (2014): LUPO – Weiterentwicklung der Landesumweltportale. In diesem Bericht.
- /5/ Schlachter, T. et al. (2014): LUPO mobil – Umweltdaten mobil: Konzepte und technologische Einblicke in die Meine Umwelt-App. In diesem Bericht.