

Ein Service-orientierter Ansatz für kontextbewusste Mash-Ups in mobilen Lernmanagementsystemen

Philipp Lehsten, Raphael Zender, Ulrike Lucke und Djamshid Tavangarian
Universität Rostock, Institut für Informatik, Lehrstuhl für Rechnerarchitektur
vorname.nachname@uni-rostock.de

Abstract: Durch die zunehmend kostengünstige Verfügbarkeit drahtloser Datennetze und die Verbreitung mobiler, internetfähiger Geräte sind Studenten in der Lage, jederzeit und überall auf eLearning-Systeme zuzugreifen. Zentrale Anwendungen wie Lernmanagementsysteme (LMS) sind allerdings nur sehr begrenzt fähig, sich an die verwendeten mobilen Geräte mit ihren Restriktionen anzupassen. Des Weiteren mangelt es LMS an Integrationsmöglichkeiten für externe Funktionen und Datenquellen, wie sie ein mobiles Gerät zur Verfügung stellen kann. In diesem Beitrag wird eine Systemarchitektur vorgestellt, die es unabhängig vom LMS ermöglicht, Informationen aus dem verwendeten mobilen System zu integrieren und mit diesen Kontextinformationen externe Funktionen und Datenquellen zu nutzen. Dies erlaubt neben einer Steigerung des Funktionsumfangs eines LMSs durch Nutzerdaten und Web-Dienste auch die Anbindung an institutionsspezifische Informationsquellen und -dienste wie Bibliothekssysteme oder die Gerätesteuerung in Vorlesungsräumen.

1 Einleitung

Smartphones, ein Sammelbegriff für leistungsstarke und internetfähige Geräte, sind in den letzten Jahren zunehmend populärer geworden [Gar10]. Auch die meisten der heute verkauften Mobiltelefone können auf das Internet zugreifen. Daher nutzen immer mehr Studenten das mobile Internet sowohl zu Hause als auch in ihren Hochschulen [Con10], da dort WLAN-Zugangspunkte weit verbreitet sind. Problematisch ist, dass viele Internetangebote der Hochschulen klassische Applikationen sind, bei denen Einschränkungen mobiler Geräte nicht berücksichtigt wurden.

Eines der wichtigsten Internetangebote an Hochschulen stellen die Lernmanagementsysteme (LMS) dar. Diese Systeme lassen sich technisch zwar durch die mobilen Geräte nutzen, jedoch bleibt neben der eingeschränkten Bedienbarkeit auch viel Potential mobiler Systeme ungenutzt [Lie09]. Des Weiteren erlauben Anpassungen an die Restriktionen dieser Geräteklasse und die Nutzung der speziellen Funktionalitäten die Realisierung von Anwendungen für verschiedene pervasive und ubiquitäre eLearning-Szenarien [ZLMM09]. Die Entwicklung von Mash-Ups, der Kombination von verschiedenen Mengen von Datenströmen hin zu einer neuen Nutzung [Kul07], führt im Umfeld von LMS und weiteren Campus-Informationssystemen zu einer Vielzahl neuer Möglichkeiten das Lernen zu bereichern. Ein Beispiel ist Sloodle, das eine Verbindung zwischen Moodle und der virtuellen Welt des Second Life herstellt [KLB09]. Die Einbettung von Mash-Ups in die Seiten eines LMS ist je nach System unterschiedlich komplex, doch gerade im mobilen Bereich wird

eine unangepasste Integration ohne Rücksicht auf die verfügbare Display-Größe und die dadurch bedingte Veränderung der Lage von Steuerelementen schnell unübersichtlich. Ein Ansatz diesem Problem zu begegnen, stellt die Anpassung an die Situation und den Kontext des Nutzers dar. Einem Nutzer in der Bibliothek kann der Weg zu vorlesungsrelevanter Literatur allein dadurch gewiesen werden, dass diese im LMS angegeben ist. Neben dem Ortskontext lassen sich auch personenbezogene Informationen mit einem LMS kombinieren. Der Dozent, der die Übungsaufgaben gepostet hat, lässt sich in Verknüpfung mit den frei zugänglichen Informationen kontaktieren. Auch der zeitliche Kontext lässt sich in eine Vielzahl verschiedener Mash-Ups integrieren. Dieser erlaubt die Bestimmung des nächsten Seminars und somit die Einbindung des jeweilig am besten geeigneten Mash-Ups. Dies, in Verbindung mit den Präferenzen des Nutzers sowie den aktuellen technischen Möglichkeiten seines Gerätes, erlaubt eine hohe Personalisierung des Lernens [ZDLT09]. Dieser Artikel gibt einen Überblick über einen Service-orientierten Ansatz, welcher Informationen aus einem LMS und Kontextinformationen aus mobilen Geräten verknüpft und so ein kontextbewusstes LMS erzeugt. Dazu werden verwandte Arbeiten betrachtet, die Architektur unseres Systems erläutert sowie auf die vier Kernkonzepte des Systems eingegangen. Abschließend erfolgen eine Zusammenfassung sowie ein Ausblick auf die nächsten Schritte und Projekte im Rahmen dieses Systems.

2 Verwandte Arbeiten

Im Folgenden wird auf Arbeiten im Umfeld von LMS, Kontext und mobilen internetfähigen Geräten eingegangen. Mit der Mobile Learning Engine (MLE) [MN04] ist es möglich ein bereits existierendes LMS zu erweitern, um Inhalte für ein mobiles Gerät aufzubereiten. Über ein Plugin ist es möglich, Inhalte aus dem LMS Moodle zu exportieren. Der Unterschied zu unserer Arbeit liegt darin, dass wir nicht direkt mit den Lernmaterialien des LMS arbeiten, sondern ein höheres Abstraktionsniveau ansetzen und uns auf die Kommunikation konzentrieren. Das Anzeigen der Inhalte des LMS überlassen wir dem Browser des Gerätes, da dieser die notwendigen Plugins zur Wiedergabe von Mediaformaten systemspezifisch ermitteln und ansprechend kann. Unser System stellt somit eine kontextorientierte Middleware zwischen einem mobilen Gerät und einem existierenden LMS dar.

Eine weitere Arbeit, die näher an unseren Intentionen liegt, konzentriert sich auf eine abstrakte Ebene zur Verknüpfung von Kontext und externen Diensten [MGC⁺09]. Der Kontext kann von Applikationen genutzt werden, jedoch obliegt ihnen dabei auch das Schlussfolgern aus den Daten. Bei unserer Arbeit wollen wir sowohl den Rohkontext, zum Beispiel GPS-Koordinaten einer Person, als auch den Kontext höherer Abstraktionen, wie das Gebäude in dem sich eine Person befindet, liefern können. Dies soll als Anfrage an Services erfolgen, die direkt und in Kombination mit anderen konsumiert werden können. Den Ansatz, Inhalte exklusiv für mobile Geräte bereitzustellen, verfolgen wir nicht, da die Entwicklung der Browser für mobile Geräte hin zu einer Annäherung an die Fähigkeiten von klassischen Browsern geht. Somit konzentrieren wir uns auf das Erkennen und Anpassen der Oberfläche und verlassen uns ansonsten auf die Weiterentwicklung der Browser und Darstellungsprogramme auf mobilen Geräten.

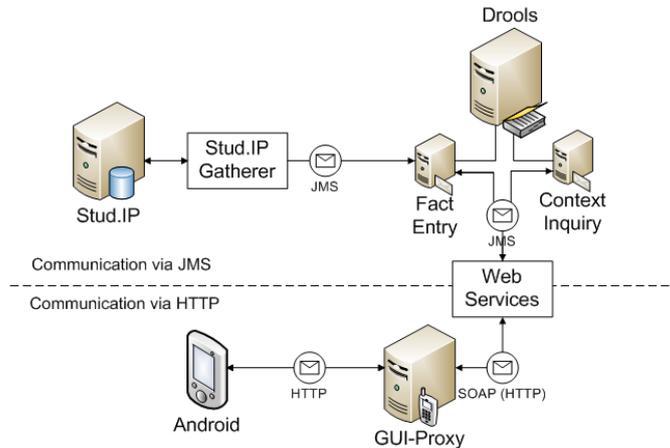


Abbildung 1: Architektur zum kontextbasierten LMS-Zugriff

3 Service-orientiertes Kontextbewusstsein in LMS

Die Anwendung von Kontext auf mobiles eLearning zeigt, dass viele relevante Informationen direkt auf dem mobilen Gerät erfasst werden können. Dies bezieht sich vor allem auf die Identität des Nutzers sowie seine Position, während hingegen Informationen über sein Studium im LMS gespeichert sind. Mit diesen beiden Kontextquellen lassen sich bereits mehrere Aspekte der Nutzersituation beschreiben. Das Sammeln und Interpretieren dieser Informationen muss flexibel und erweiterbar sein, um eine Integration neuer Kontextquellen aus intelligenten Veranstaltungsräumen und der Heimautomation zukünftig ebenfalls zu ermöglichen.

3.1 Gesamtarchitektur des mobilen LMS Prototyps

In der Realisierung unseres Systems nutzen wir das LMS Stud.IP als Ausgangspunkt. Wie Moodle ist es Open Source, jedoch liegt der Fokus neben den Lernmaterialien auf Veranstaltungen und Einrichtungen, wodurch mehr relevanter Kontext enthalten ist. Diesen speichern wir in einem Kontextserver, den wir mit Drools, einem freien regelbasierten System auf Javabasis, realisiert haben. Dieses bringt Schnittstellen für den Java Message Service (JMS) mit, welchen wir als nachrichtenorientierte Middleware nutzen. Für das mobile Gerät nutzen wir das Android Betriebssystem, da es ebenfalls Open Source ist und sich die Entwicklung effizient gestaltet. Die Verbindung zwischen dem mobilen System und Stud.IP erfolgt über einen HTTP-Proxy, der die Anfrage des mobilen Systems empfängt und nach Auswertung weiterer Kontextinformationen über Web Services eine Weiterleitung zu einer gerätespezifisch geeigneten Version des Stud.IP realisiert. Ein Überblick über die Architektur ist in Abbildung 1 gegeben.

3.2 Wissensbasiertes Backend für das LMS Stud.IP

Stud.IP speichert alle Inhalte in einer MySQL-Datenbank. Die dort gesammelten Informationen über Stud.IP-Konstrukte wie Nutzer, Kurse, Veranstaltungen und Räume werden in serialisierbare Objekte überführt. Diese Objekte repräsentieren Entitäten mit statischen Informationen wie der Raumbezeichnung oder dem Titel der Vorlesung. Externe Inhalte, wie die Position mobiler Geräte, werden ebenfalls in diesen Objekten abgelegt. Den ersten Teil des Systems stellt somit das Datenmodell für den Kontext dar, da hier die verschiedenen Kontextarten miteinander verknüpft werden. Ein faktenorientierter Ansatz mit einem wissensbasierten System (Knowledge-based System, KBS) deckt sowohl Modell als auch Speicherung ab. Dies erlaubt eine relationale Repräsentation der Daten sowie eine mehrstufige Grammatik für die verschiedenen Ebenen des Kontextes [BBH⁺09]. Das verwendete Kontextmodell trennt zwischen statischen und dynamischen Informationen, die jedoch auf einander verweisen und somit beide im KBS gespeichert werden können. Nach dem Sammeln der Informationen werden die Regeln des KBS angewendet und damit Verbindungen zwischen den Objekten etabliert. Der Zugang zum System muss skalierbar sein, um in Lastzeiten des Hochschultages arbeitsfähig zu bleiben. Zu diesem Zweck nutzen wir eine nachrichtenorientierte Middleware, die asynchrone Kommunikation ermöglicht. Das KBS erhält dabei Nachrichten, die nur die Objekte enthalten, die neu sind oder über neue Daten verfügen. Diese Objekte stellen Abstraktionen der realen Welt dar und werden über Regeln verknüpft. Während die Anwendung von Regeln durch das System selbst angestoßen wird, werden die Anfragen von außen gestellt. Dabei werden Fragen, wie die nach der nächsten Veranstaltung eines Nutzers, beantwortet, um diesem einen Link zu der damit verbundenen Seite im LMS zu liefern. In unserem System wird diese Funktionalität durch Web Services gekapselt, die wiederum miteinander gekoppelt werden können um komplexere Anfragen zu realisieren.

3.3 Service-orientierter Kontextzugriff

Um auf die gespeicherten Informationen zugreifen zu können ist es notwendig zu wissen, welche Objekte gespeichert und welche Anfragen möglich sind. Dieses Wissen wird auf einem plattformunabhängigen Weg kommuniziert, da die spätere Verwendung der Kontextinformationen nicht auf das LMS beschränkt bleiben, sondern der Weg auch für PHP-Seiten oder Browser-Plugins offen bleiben soll. Um dies zu erreichen, ist eine Service-orientierte Architektur mit Web Services der komfortabelste Weg. Die Objekte werden serialisiert und liegen durch XML in einem sowohl für Menschen als auch für Maschinen lesbaren Format vor. Die Kommunikation erfolgt über SOAP-Nachrichten, die über HTTP versendet werden. Die Nutzung dieser Konstruktion wird von jeder Programmiersprache unterstützt, die über eine Netzwerkfunktionalität verfügt. Anfragen werden von Web Services bearbeitet, die auch das KBS im Hintergrund vor zu viel Last schützen, da sie nicht direkt auf Informationen aus dem System zugreifen, sondern über die nachrichtenorientierte Middleware mit ihm kommunizieren. Die Web Services werden dynamisch über einen Broker in das System eingebunden und ermöglichen so neben der Kapselung auch

die Erweiterung des Systems um weitere Kontextquellen. Diese Nutzung einer SOA in Kombination mit einer nachrichtenorientierten Middleware ermöglicht neben der Plattformunabhängigkeit auch die Implementierung von Subsystemen, die in entsprechenden Situationen Nachrichten cachen oder weitere Instanzen existierender Web Services starten.

3.4 Kontextnutzung bei Zugriff von mobilen Geräten

Um bei der Vielfalt an mobilen Systemen ein möglichst breites Spektrum erreichen zu können, muss das System zur Anzeige möglichst einfach und im Idealfall mit jedem internetfähigen Gerät lauffähig sein. Daher verwenden wir zum Zugriff auf das LMS einen Proxy. Dieser versucht, das verwendete Gerät sowie dessen Plattform zu identifizieren und ihm so eine optimierte Version des LMS liefern zu können. Dieser Vorgang ist unabhängig von der Erfassung weiterer Kontextinformationen und basiert ausschließlich auf dem HTTP-Aufruf. Um ein Kontextbewusstsein zu erzeugen, wurde auf dem Gerät eine Software installiert, die beim Aufruf des LMS den HTTP-Aufruf mit geräte- und nutzerspezifischen Informationen anreichert. Diese Software nutzt nur grundlegende Funktionalitäten und kann durch den geringen Funktionsumfang sehr klein gehalten werden. Des Weiteren ist sie unabhängig von der Darstellung des LMS, die im Standardbrowser des Gerätes abläuft. Während ein Nutzer ohne spezielle Software auf seinem Gerät lediglich eine für sein Gerät optimierte Version erhält, ist es bei Auswertung mitgelieferter Kontextinformationen möglich, durch Mash-Ups die dargestellten Inhalte zu personalisieren. So wird zum Beispiel durch beigefügte Positionsinformationen aus dem GPS-Sensor ermittelt, ob sich der Nutzer auf dem Campus befindet. Zur Positionsbestimmung innerhalb von Gebäuden verwenden wir visuelle QR-Tags, die vom Nutzer proaktiv eingelesen werden. Aus diesen Informationen und der Identifikation des Nutzers ist es möglich, die potentiellen Veranstaltungen des Nutzers zu bestimmen. Die entsprechenden Seiten des LMS werden verlinkt, wobei Raumänderungen hervorgehoben werden können.

4 Zusammenfassung und Ausblick

Dieser Artikel erläutert ein Konzept und einen Prototypen zur Erfassung und Auswertung von Kontextinformationen sowie zum vereinfachten und universellen Zugriff auf ein LMS durch Services, in denen Informationen aus verschiedenen Quellen aggregiert werden. Das vorgestellte System kann Schlussfolgerungen aus erfassten Informationen ziehen und neben den Rohdaten auch Daten höherer Abstraktion liefern. Die Kommunikationsinfrastruktur mit ihrer Kopplung aus Services und einer nachrichtenorientierten Middleware ist erweiter- und skalierbar. Mit der Verwendung von Open Source Software in jeder Ebene ist der Einsatz an Bildungseinrichtungen sowie eine Anpassung an institutionelle Gegebenheiten möglich. Es stellt somit einen Schritt hin zu einer weitreichenden Integration mobiler Geräte in den universitären Alltag dar. Aktuell steht eine Erweiterung des Systems zur Anzeige von Veranstaltungen, Rauminformationen und weiteren nutzerspezifisch relevanten Orten über OpenStreetMaps kurz vor der Fertigstellung. Im nächsten Schritt soll

die Lokalisierung innerhalb von Gebäuden ohne Nutzerinteraktion realisiert und integriert werden. Darüber hinaus sollen Services dynamisch Video- und Audio-Streams anbieten, die den Gerätespezifikationen der Nutzer angepasst sind. Ferner ist auch eine verbesserte mobile Kontrolle von Veranstaltungstechnik wie Beamern und Sonnenblenden geplant.

Danksagung

Diese Arbeit wird durch die Deutsche Forschungsgemeinschaft unterstützt (DFG, GRK1424).

Literatur

- [BBH⁺09] Claudio Bettini, Oliver Brdiczka, Karen Henriksen, Jadwiga Indulska, Daniela Nicklas, Anand Ranganathan, and Daniele Riboni. A survey of context modelling and reasoning techniques. *Pervasive and Mobile Computing*, In Press, Corrected Proof, 2009.
- [Con10] New Media Consortium. The 2010 Horizon Report. 2010.
- [Gar10] Gartner, Inc. Gartner Says Worldwide Mobile Phone Sales to End Users Grew 8 Per Cent in Fourth Quarter 2009; Market Remained Flat in 2009, 2010.
- [KLB09] J. W. Kemp, D. Livingstone, and P. R. Bloomfield. SLOODLE: Connecting VLE tools with emergent teaching practice in Second Life. *British Journal of Educational Technology*, 40(3):551–555, 2009.
- [Kul07] N. Kulathuramaiyer. Mashups: Emerging Application Development Paradigm for a Digital Journal. *Journal of Universal Computer Science*, 13(4):531–542, 2007.
- [Lie09] G. Liestøl. Situated Simulations: A Prototyped Augmented Reality Genre for Learning on the iPhone. *International Journal of Interactive Mobile Technologies*, 3, 2009.
- [MGC⁺09] S. Martin, R. Gil, E.S. Cristobal, M. Milev, N. Mileva, G. Daz, M. Castro, and J. Peire. Middleware for the development of context-aware applications inside m-Learning: Connecting e-learning to the mobile world. In *2009 Fourth International Multi-Conference on Computing in the Global Information Technology*, pages 217–222, Cannes/La Bocca, France, August 2009. IEEE Press.
- [MN04] M. Meisenberger and A. Nischelwitzer. The mobile learning engine (MLE) - a mobile, computer-aided, multimedia based learning application. In *Mapec, Multimedia Application in Education Conference*, 2004.
- [ZDLT09] R. Zender, E. Dressler, U. Lucke, and D. Tavangarian. Pervasive media and messaging services for immersive learning experiences. *Pervasive Computing and Communications, IEEE International Conference on*, pages 1–6, 2009.
- [ZLMM09] R. Zender, U. Lucke, D. Maciuszek, and A. Martens. Interconnection of Game Worlds and Physical Environments in Educational Settings. In *8th Workshop on Network and Systems Support for Games (NetGames)*, Paris, 2009.