ERItemManager — ein dynamisch rekonfigurierbares Webapplikationsframework

Sascha Preibisch, Tao Zheng und Wolfgang Renz *
Labor für Multimediale Systeme (MMLab),
Fakultät Technik und Informatik,
Hochschule für Angewandte Wissenschaften Hamburg,
Berliner Tor 7, 20099 Hamburg, Germany

Abstract:
XML–konfigurierbare Webapplikationsframeworks sind Stand der Technik. Deskriptive XML–Programmierung mit eingebetteten Aktivitäten auf Basis von Mittelwareplattformen wurde in verschiedenen Kontexten als Alternative zur schwerfälligeren J2EE–Webapplikationsentwicklung vorgeschlagen und realisiert. Mit dem ERItemManager wird ein vom Ercaton-Paradigma[Ercato] inspirierter dynamisch rekonfigurierbarer ItemManager konzipiert und realisiert. Um den Entwickler dabei zu unterstützen, sich auf das fachliche Problem konzentrieren zu können, bietet das Framework die Möglichkeit, mit Hilfe einer validen XML-Datei sämtliche Teile der Anwendung, die nicht von der fachlichen Logik abhängig sind, generieren zu lassen. Darunter fällt der gesamte Bereich der Persistenz, der Schnittstelle zwischen Client und Server und die Erzeugung von Benutzerschnittstellen. Alle diese Elemente sind sofort, ohne jegliche Änderung des Frameworks, nutzbar. Einzig die fachliche Funktionalität oder eine spezialisierte Benutzerschnittstelle müssen vom Entwickler neu entwickelt werden. Dies geschieht gegen spezifizierte Schnittstellen. Exemplarisch wurde damit ein Prototyp einer Webapplikation zu Projektmanagement, Bugtracking und Zeiterfassung realisiert, um das geschaffene Framework bewerten zu können.

1 Zielsetzung

Zu Beginn der Diplomarbeit des ersten Autors wurden Alternativen untersucht, wie für die firmeninterne Nutzung eine Software zu entwickeln ist, mit der Arbeitszeiten erfasst und ausgewertet, Benutzer verwaltet, Berichte erzeugt werden können und die Bugreporting ermöglicht. Im Laufe der Anforderungsanalyse stellte sich heraus, dass es sinnvoll war, ein Framework zu entwickeln, mit dem man auf immer neue Anforderungen eingehen kann, da keine passende Software bekannt war, die die gesetzten Anforderungen direkt erfüllte als im Sept. 2004 die Entwicklung des ERItemManager gestartet wurde. Niedrigen Beschaffungskosten waren Voraussetzung, da die zu erstellende Software nicht direkt zur Umsatzsteigerung beiträgt. Zudem sollte die benötigte Infrastruktur möglichst einfach sein. Bei jeder firmeninternen Eigenentwicklung hatte man sich mit Persistenz,

*sascha.preibisch@ipt.ch, zhengtaofw@hotmail.com, wr@informatik.haw-hamburg.de

Zugriffschicht, Benutzerschnittstelle und Ablaufkontrolle beschäftigen müssen, ein sehr grosser Aufwand, der dazu geführt hatte, auf die Entwicklung eines solchen Werkzeugs zu verzichten. Dieses Problem sollte durch den ERItemManager gelöst werden, dessen aktuelle, einsetzbare Version im Rahmen der sequentiellen Abschlussarbeiten der beiden erstgenannten Autoren entwickelt wurde.

2 Das Framework

Inspiriert vom Ercaton-Paradigma[Ercato] stellten wir die Anforderung, dass das ERItemManager Framework es ermöglicht, ein Projekt einzig in Form einer XML Datei zu beschreiben, die ggf. Referenzen auf in Java implementierte Aktivitäten enthalten kann. Diese Projektdatei sollte die Struktur von Geschäftsobjekten (Businesslogik) sowie die Konfiguration der Persistenzschicht und einer passenden Zugriffsschicht enthalten. Ausserdem sollte auf dieser Basis ein Standard Fat Client generiert werden. Nach der Fertigstellung der Projektdatei sollte man also folgende Bestandteile erhalten:

1. die Persistenzschicht in Form von Tabellen einer relationalen Datenbank inkl. sämtlicher Schlüssel und Indizes,
2. eine in Java implementierte Zugriffsschicht; für die Kommunikation mit der Persistenzschicht werden webservices genutzt,
3. Benutzerschnittstelle implementiert mit dem Eclipse RCP Framework.

Bestandteile der zur Verfügung gestellte Infrastruktur sind: Java 5, Apache Tomcat 5.5, Apache Axis 1.2, PostgreSQL 8.0 (oder ein anderes RDBMS) Der Workflow für die Erzeugung eines Projektes sieht wie folgt aus:

1. Anlegen zweier Datenbanken, eine für Projekttabellen, eine für Systemtabellen, die Projekt unabhängige Daten enthält,
2. Erstellen der Projektdatei mit Hilfe des enthaltenen Projekt Editors,
3. das Projekt mit Hilfe des vorhandenen Administrationsdialogs instanziieren

Die technische Analyse, Design und die erste Implementierung erforderten ca. 15 Wochen. Die ursprüngliche Implementierung benutze DTD, wurde in der zweiten Abschlussarbeit aus technischen Gründen durch Schema ersetzt. Wesentliche Erweiterung der zweiten Arbeit war, dass typische Fälle späterer Änderungswünsche für existierende Projekte erfasst wurden, und die Projektänderung ohne Datenverlust der zu erhaltenden persistenten Daten auf dieser Basis konzipiert und implementiert wurde. Erst dadurch wurde das Framework für den praktischen Einsatz akzeptabel. Außerdem wurde Internationalisierung mit Unterstützung der Sprachen deutsch, englisch und chinesisch realisiert.

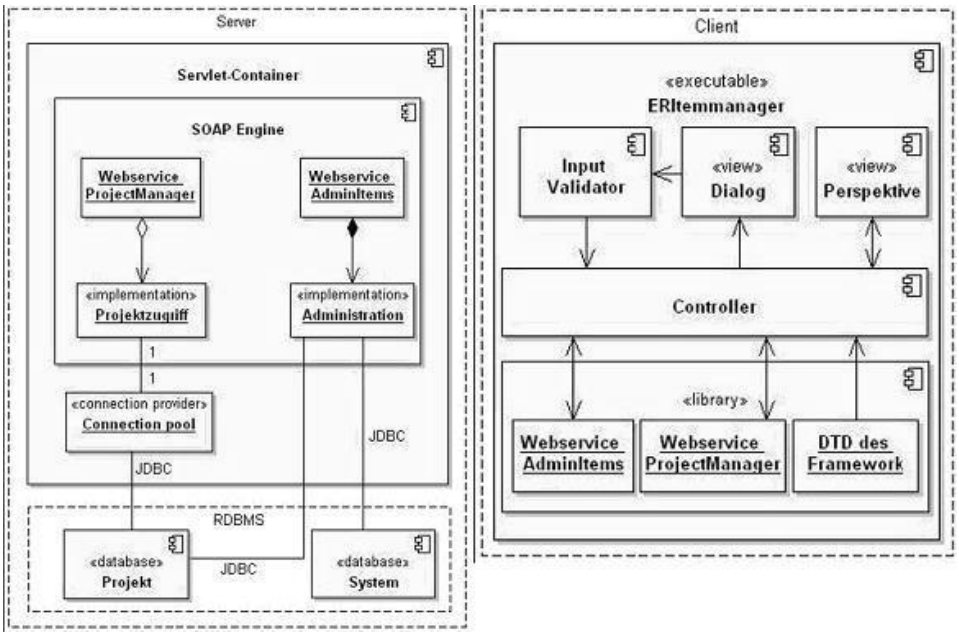

Abbildung 1: Architektur des ERItemManager–Frameworks

3 Abgrenzung

ERItemManager verzichtet bewusst auf die Nutzung der im folgenden beschriebenen Frameworks, um die Komplexität und Einarbeitungszeit gering zu halten:

- *Hibernate (www.hibernate.org):*
  Hibernate sorgt als objektrelationales Mappingtool für eine Abbildung von Objekten einer objektorientierten Programmiersprache auf Tabellen einer relationalen Datenbank. Damit können komplexe Abbildungen und Änderungen an Datenstrukturen durchgeführt und Transaktionsmanagement realisiert werden.

- *Spring (www.interface21.com):*
  Spring hat eine sehr ähnliche Intention wie das ERItemManager–Framework, nämlich eine einfachere Lösungen für immer wiederkehrende Probleme bereitzustellen als z.B. Suns EJB Technologie. Das Spring Framework besteht heute aus verschiedenen unabhängig voneinander nutzbaren Modulen, wie Spring MVC, Spring WEB, Spring AOP und andere. Jedes einzelne Projekt verfolgt das Ziel, die Entwickler von wiederkehrenden und nicht fachlichen Aufgaben weitestgehend zu entbinden. Spring kann hibernate für die Persistierung einsetzen. Durch weitgehende Abstraktion sind beim Entwickler nur geringe hibernate-Kenntnisse erforderlich. Allein durch Konfiguration kann von einer jdbc-Datenquelle auf eine andere gewechselt werden kann.

- JSP/ Servlet (java.sun.com) : Bei JSP (JavaServer Pages) und Servlets handelt es sich um Technologien, um dynamisch webbasierte Benutzeroberflächen generieren zu können. Das ERItemManager Framework wurde zwar mit einem Fat Client ausgestattet, aber die offene Struktur ermöglicht es Entwicklern auch, Benutzeroberflächen einzusetzen, die in einem Browser dargestellt werden können. JSP/ Servlets wurden nicht für die Standardimplementierung einer Benutzeroberfläche eingesetzt, weil man nicht auf die Qualitäten eines Fat Clients verzichten wollte.

4 Zusammenfassung

Durch das ERItemManager Framework können Projektteams in sehr kurzer Zeit mit sehr geringem Kostenaufwand eine Software erstellen, die ihren Bedürfnissen angepasst ist. Mit den aus der Projektdatei generierten Elementen ist man in der Lage, Daten verschiedenster Typen zu persistieren, über Netzwerkgrenzen hinweg an einem Projekt zu arbeiten. Zukünftig soll es möglich sein, benutzerdefinierte Funktionen auf einfache Art in den Arbeitsprozess einzubinden. Damit ist die Personalisierung der generierten Elemente möglich. Für den sofortigen Einsatz ist nach der Erstellung der Infrastruktur alles vorbereitet. Eine erste Version des Frameworks wurde während der Diplomarbeit des ersten Autors für die Firma Comosoft entwickelt. In der Abschlussarbeit des zweiten Autors wurde das Framework wie erläutert erweitert. Die Arbeiten haben einschließlich Einarbeitungszeit ca. 6 Monate erfordert.

5 Danksagung

Wie danken Firma Comosoft für die Unterstützung und insbesondere Matthias Schaadt für wertvolle Diskussionen und firmeninterne Betreuung.

Literatur

[] Standardwerke des Software-Engineerings sowie Bücher und Webseiten zu den eingesetzten Techniken

[Ercato] O. Imbusch, F. Langhammer und G. von Walter. Ercatons: Thing Oriented Programming. *LNCS 3263*, 216–237, 2004