

Datenerhebung in der Schulstatistik

@SchuLE.rlp – ein Pilotprojekt mit Java- und XML-Technologien

@SchuLE.rlp steht für **Automatischer Transfer von Schüler und Lehrerdaten** in Rheinland-Pfalz und startete als Pilotversuch zum Erfassen und Übermitteln von Daten für die amtliche Schulstatistik mit dem Beginn des Schuljahres 2002/2003. Ziel des Projektes @SchuLE.rlp ist es, Schuldaten in einem einheitlichen, plausiblen Datenformat zu erhalten.

Dies war in den vergangenen Jahren anders: Die Daten wurden von den rheinland-pfälzischen Schulen in Form von Erhebungsbogen in einem starren, vorgegebenen Satzaufbau entweder im ASCII-Datenformat oder als Excel-Datei dem Statistischen Landesamt übermittelt.

Der Schwerpunkt dieses Artikels liegt auf dem technischen Aspekt der neuen Anwendung.

Ziele des Projekts

Beim Initiieren des Projektes waren Projektziele hinsichtlich der Anforderungen an die zu erstellende Software zu definieren und Ziele aus DV-technischer Sicht festzulegen:

Die zu erstellende Software sollte folgende Anforderungen erfüllen:

- einheitliche Datenschnittstelle,
- Betriebssystemunabhängigkeit,
- selbsterklärende Installation sowie möglichst geringer Administrationsaufwand in den Schulen,
- Gewährleistung der Sicherheit der Daten und des Datenschutzes,
- geringer Erfassungsaufwand,
- keine Softwarekosten für die Schulen.

Aus DV-technischer Sicht waren als Ziele

- der Aufbau einer objektorientierten Entwicklungsumgebung,
- geringe Bindungen an Betriebssysteme und Hersteller und
- der Einstieg in die objektorientierte Software-Entwicklung mit dem vorhandenen Personal

für die Entwicklungsarbeit maßgeblich.

Anforderungen an die zu erstellende Software

1. Einheitliche Datenschnittstelle

Die Daten, welche die Schulen jedes Jahr an das Statistische Landesamt Rheinland-Pfalz übermitteln, sollen ab dem Schuljahr 2003/2004 landesweit über ein einheitliches Datenformat geliefert werden. Diese Daten werden aus unterschiedlichen technischen Quellen erzeugt: Sie können mittels des neuen Pro-

grammes @SchuLE.rlp erfasst oder mit Hilfe eines kommerziellen Schulverwaltungsprogrammes erstellt werden.

Erhebungsbögen, ASCII-Schnittstelle und Excel-Formate sollen nicht mehr unterstützt werden. Aus diesem Grund wurde frühzeitig Kontakt zu den Schulaufsichtsbehörden, dem Ministerium für Bildung, Frauen und Jugend (MBFJ) und der Aufsichts- und Dienstleistungsdirektion (ADD, Trier) sowie den Herstellern von Schulverwaltungsprogrammen aufgenommen, um eine einheitliche Datenschnittstelle zu definieren.

Technische Grundlage sollte hier die Extensible Markup Language (XML)¹⁾ sein, welche mittlerweile als Standard für Datenaustauschformate anzusehen ist. Dieses Format für die Übermittlung der Daten der amtlichen Schulstatistik durchzusetzen, erforderte einige Überzeugungsarbeit. Der Vorteil von XML liegt darin, dass es gut lesbar ist (keine Binärcodes), starke Unterstützung bei den führenden Softwareherstellern (Microsoft, Oracle, IBM, Sun und Apache) erfährt und vom World Wide Web Consortium (W3C) als unabhängiges Gremium standardisiert ist. Viele neue Standard- und Open Source Softwareprodukte, welche in Zukunft auch für das Statistische Landesamt von Interesse sein könnten, basieren ebenfalls auf XML. Ein weiterer, entscheidender Vorteil ist die Möglichkeit der Überprüfung des XML-Datenformates mit XML-Schema²⁾. Hiermit kann jede XML-Datei zweistufig wie folgt auf formale Gültigkeit geprüft werden:

1. Stufe: Prüfung auf syntaktische Korrektheit (= wellformed)
2. Stufe: Überprüfung auf Typen, Wertebereiche, Feldanordnungen entsprechend dem definierten XML-Schema (= valid)

Durch das Zusammenspiel von XML und XML-Schema wird eine sehr gute Datenqualität der zu liefernden Schuldaten erwartet. Ein Nachteil von XML-Daten ist der wesentlich größere Plattenspeicherbedarf gegenüber herkömmlichen Datenformaten. Durch moderne Komprimierungsprogramme lässt sich dieser Nachteil jedoch minimieren.

2. Betriebssystemunabhängigkeit

Das Programm zur Erfassung der Daten in den Schulen muss betriebssystemunabhängig sein, da in den rund 1600 Schulen in Rheinland-Pfalz mit den unterschiedlichsten Hardware-Gegebenheiten gerechnet werden muss. Als plattformunabhängige Programmiersprache ist die objektorientierte Sprache Java mittlerweile stark verbreitet. Die Entscheidung zu Java erforderte im Statistischen Landesamt die Einrichtung einer neuen Softwareentwicklungsumgebung und eine Schulung der beteiligten Projektmitarbeiterinnen und -mitarbeiter.

1) Siehe hierzu: <http://www.w3.org/XML>

2) Siehe hierzu: <http://www.w3.org/XML/Schema.html>

3. Installation

Die Installation und die Bedienung der Anwendung muss möglichst einfach zu handhaben sein. Das Statistische Landesamt ist personell nicht in der Lage eine größere Anzahl von Mitarbeiterinnen und Mitarbeitern für die Betreuung eines einzelnen Verfahrens in den gut 1 600 Schulen zu binden. Ebenso muss eine Installation der Software in den Schulen von EDV-unerfahrenem Personal durchführbar sein. Hier bot sich das Produkt Java Web Start³⁾ an, das in der Java Runtime Environment (JRE) ab der Version 1.4.0 enthalten ist. Die JRE ist die technische Grundlage des gesamten Programmpaketes und muss auf den Rechnern in den Schulen installiert sein.

Um Java Web Start zu nutzen, wird die Anwendung @SchuLE.rlp serverseitig vom Statistischen Landesamt auf einem mit dem Internet verbundenen Webserver als Java Archiv (.jar-Datei) abgelegt. Die Anpassungen für Java Web Start auf diesem Rechner sind minimal. Der Klient, also der Bearbeiter in der Schule, ruft über ein Browserprogramm (z. B. den

Internet Explorer von Microsoft) und das Internet eine vom Statistischen Landesamt zur Verfügung gestellte HTML-Seite auf. Durch einen einfachen Mausklick auf eine Schaltfläche oder ein Bild auf dieser Seite wird automatisch Java Web Start gestartet und die Anwendung @SchuLE.rlp auf dem PC in der Schule in einem speziellen Arbeitsbereich, dem so genannten Cache von Java Web Start, installiert.

Bei jedem weiteren Aufruf des Programmes ist keine Internetverbindung mehr erforderlich. Besteht jedoch zum Zeitpunkt eines erneuten Programmstarts eine Internetverbindung, erfolgt ein automatischer Versionsabgleich der Software @SchuLE.rlp. Existiert auf dem Webserver eine neuere Version als in der Schule, wird eine automatische Aktualisierung der Software, vorgenommen, ansonsten wird das Programm aus dem Arbeitsbereich des PCs gestartet. Dieser Versionsabgleich benötigt wenig Zeit und basiert auf dem Java Network Launching Protocol (JNLP). Durch diesen Mechanismus wird ohne Installationsaufwand eine aktuelle Softwareversion der Anwendung in den Schulen gewährleistet. Ein teurer Versand von Disketten oder CDs und ein manuelles Installieren von neuen Versionen in den Schulen entfällt.

3) Siehe hierzu: <http://java.sun.com/products/javawebstart>



4. Datensicherheit und Datenschutz

Die Applikation @SchuLE.rlp soll neben der Erfassungs- und Datenübermittlungsfunktionalität in den Schulen im Statistischen Landesamt als Auskunft- und Plausibilisierungstool genutzt werden. Hierfür ist es zur Programmierung der Anwendung wesentlich einfacher, wenn die Datenhaltung in den Schulen und im Statistischen Landesamt auf gleicher technischer Grundlage realisiert wird. Die von den Schulen eingehenden Daten werden im Statistischen Landesamt nach der Plausibilisierung und zur weiteren Verarbeitung in eine relationale Datenbank (Oracle) eingelagert. Eine Oracle-Datenbank in den Schulen kam nicht in Frage, da dies den Projektzielen

- keine Softwarekosten für die Schulen und
- selbsterklärende Installation sowie möglichst geringer Administrationsaufwand in den Schulen

widersprach. Die Entscheidung fiel letztendlich auf `hsqldb`⁴⁾ eine relationale Open Source Datenbank, welche sich bereits in einigen Projekten der Apache Group bewährt hatte. Aus den Javaprogrammen kann mit der genormten Java Database Connectivity-Schnittstelle (JDBC-API) auf diese Datenbank performant zugegriffen werden. Diese auf Java basierende Datenbank lässt sich in die Anwendung vollständig integrieren, so dass keine Administrationsaufgaben in den Schulen zur Wartung der Datenbank entstehen.

5. Geringer Erfassungsaufwand

In den Schulen werden Daten nicht nur für die amtliche Schulstatistik, sondern auch für die Schulaufsichtsbehörden (MBFJ, ADD) erhoben. Ein grundlegendes Konzept der neuen Anwendung beruht auf der Tatsache, dass für jeden Schüler und Lehrer anonymisierte Datensätze bestehen. Das bisherige Verfahren verarbeitet von Lehrern Einzeldaten, jedoch von Schülern Summen auf Klassenbasis. Diese Umstellung erfordert im ersten Jahr mehr Erfassungsaufwand, ermöglicht jedoch softwareseitig ein weitestgehend automatisiertes Generieren der benötigten Daten in den Folgejahren, wodurch der Erfassungsaufwand in den Schulen stark vermindert wird. Des Weiteren ist mittelfristig eine einheitliche Datenschnittstelle für das Ministerium für Bildung, Frauen und Jugend (MBFJ), die Aufsichts- und Dienstleistungsdirektion (ADD, Trier) und das Statistische Landesamt auf Basis von XML geplant. Dadurch wird von den Schulen nur noch eine einheitliche Datenbasis für alle Behörden benötigt. Das Verteilen der Daten an die unterschiedlichen Dienststellen geschieht dann auf Serverseite, unter Beachtung des Datenschutzes, in einem eigenständigen Prozess.

4) Siehe hierzu: <http://hsqldb.sourceforge.net>

5) Siehe hierzu: <http://xml.apache.org/xerces-j/index.html>

6) Siehe hierzu: <http://www.altova.com>

7) Siehe hierzu: <http://borland.com/jbuilder/index.html>

8) Siehe hierzu: <http://www.textpad.com>

9) Siehe hierzu: <http://java.sun.com/j2se>

10) Siehe hierzu: <http://xml.apache.org/xerces-j>

11) Siehe hierzu: <http://jdom.org>

6. Keine Softwarekosten für die Schulen

Die für den Ablauf der Anwendung in den Schulen benötigte Software ist frei erhältlich und es sind keine Lizenzgebühren zu bezahlen. Voraussetzung hierfür war die bewusste Entscheidung für frei erhältliche, standardisierte Technologien. Java wird von Sun Microsystems bereitgestellt, XML ist im weitesten Sinne nur eine Beschreibung eines Datenformates, dessen Nutzung jedem freigestellt ist. Die benötigten Java-Bibliotheken für eine XML-Verarbeitung sind Open Source Produkte der Apache Group und dort unter der Produktbezeichnung Xerces⁵⁾ erhältlich. Die Datenbanksoftware `hsqldb` ist ebenfalls eine Open Source Software (OSS) und unterstützt den Standard JDBC und somit auch die Standard Query Language (SQL) für relationale Datenbanken.

DV-technische Ziele

1. Aufbau einer objektorientierten Entwicklungsumgebung

Für die Entwicklung der neuen Anwendung bzw. den Einsatz der neuen Technologien musste im Statistischen Landesamt eine neue Entwicklungsumgebung eingerichtet und die Projektmitarbeiter mussten entsprechend geschult werden. Da es sich bei @SchuLE.rlp um ein Pilotprojekt handelt, sollen durch das Projekt auch Rückschlüsse auf die Tauglichkeit der eingesetzten Entwicklungstools gewonnen werden.

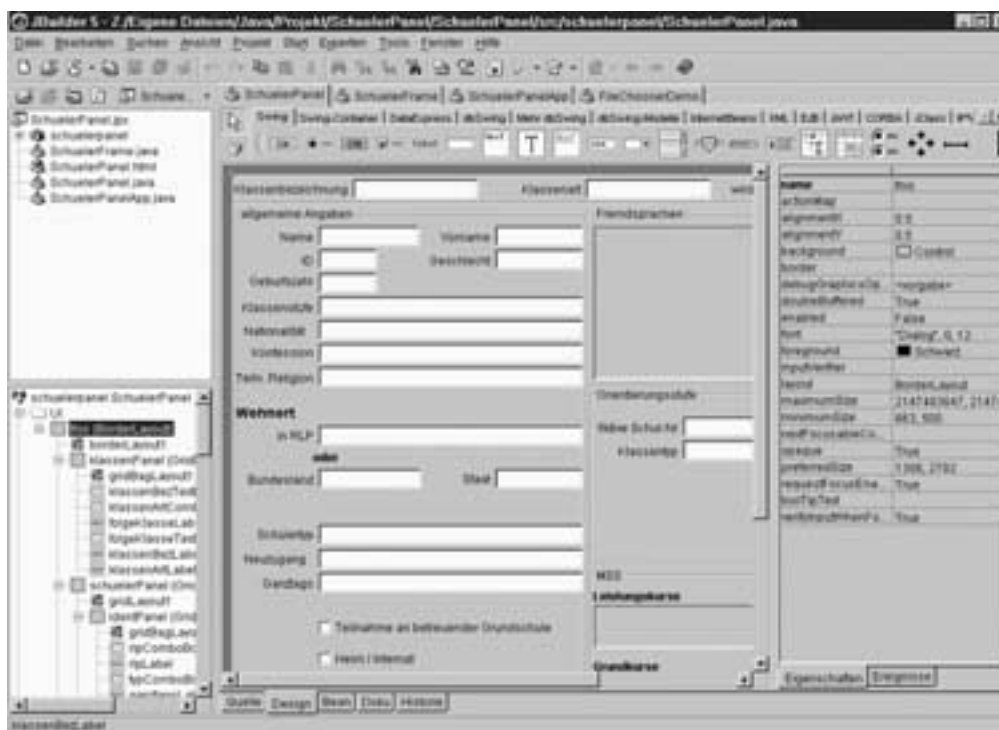
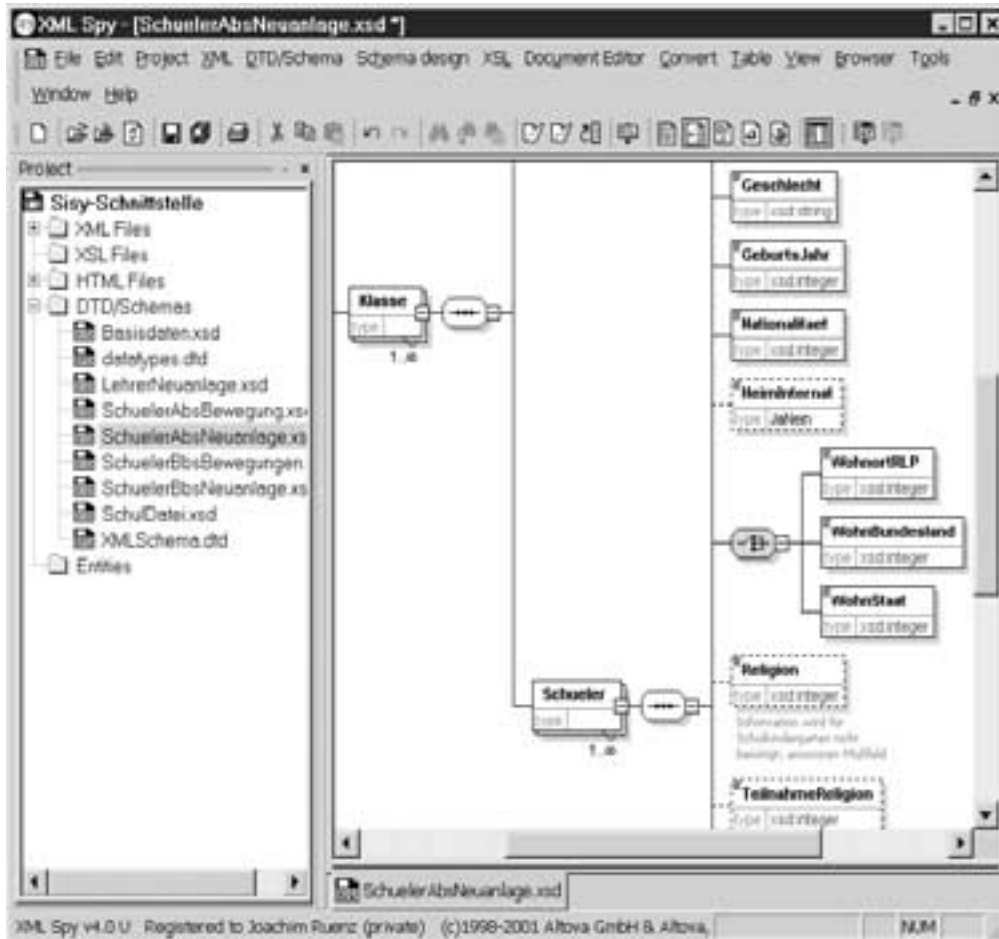
Für das Erstellen von XML-Schemata hat sich das Tool XML Spy Version 4 von Altova⁶⁾ bewährt. Mit diesem Tool kann auf sehr einfache Weise ein XML-Schema mit einem Grafikeditor erstellt werden. Die Dokumentation der erstellten Schemata kann in HTML erzeugt werden. In diesem Format wurde sie auch den Herstellern von Schulverwaltungsprogrammen zur Verfügung gestellt.

Ursprünglich sollte für die Programmierung der Java-Programme ein modernes RAD-Tool (Rapid Application Development), JBuilder 5 Professional von Borland eingesetzt werden.⁷⁾ Die Handhabung dieses Tools war nicht sehr einfach und benutzerfreundlich. Genutzt wurde der JBuilder letztlich nur zum Erzeugen der graphischen Oberflächen, wo er unbestritten sehr hilfreich war, ansonsten einigte sich die Projektgruppe darauf, mit dem Editor Textpad 4.5⁸⁾ die Java-Programme zu erstellen. Hieraus ergab sich, dass die Professional-Version des JBuilders für dieses Projekt überdimensioniert war. Die wesentlich kostengünstigere Standard-Version des JBuilders wäre ausreichend gewesen.

Zum Übersetzen, Ausführen und Testen der Java-Programme wurde die Java 2 Standard Edition (J2SE) von Sun Microsystems⁹⁾ in der Version 1.4.0_01 benötigt.

Für das komfortable Verarbeiten von XML-Dateien in Java-Programmen wurde aus dem Internet die Open Source Software Xerces der Apache Group¹⁰⁾ und JDOM¹¹⁾ heruntergeladen.

Eine mit Java Mitteln selbst erstellte Versionsverwaltung hat sich im Einsatz bewährt, soll aber im nächsten Projekt durch die Projektverwaltungssoftware



Concurrent Versions System (CVS)¹²⁾, ebenfalls Open Source Software, abgelöst werden. Diese Projektverwaltungssoftware soll auch in der Verbundprogrammierung der Statistischen Ämter eingesetzt werden.

Ein auf Basis des UNIX-Kommandos make eigenentwickeltes Build-Tool zum Zusammenstellen aller Softwarekomponenten soll ebenfalls durch ein Open Source Produkt, Ant¹³⁾, ersetzt werden. Ant ist ein plattformübergreifendes Tool und wird über XML-Dateien gesteuert.

Die Software @SchuLE.rlp kann mit jeder relationalen Datenbank, die einen JDBC-Treiber zur Verfügung stellt zusammenarbeiten. In der Anwendung wird nur Standard SQL verwendet. Lediglich an einer Stelle im gesamten Programmpaket traten kleine Unterschiede zwischen den benutzten Datenbanken, Oracle und hsqldb, auf.

2. Geringe Bindungen an Betriebssysteme und Hersteller

Durch die Berücksichtigung und Einhaltung von Standardtechnologien ist die neue Entwicklungsumgebung weitgehend unabhängig von Herstellern und Betriebssystemen. Dadurch ist es jederzeit möglich, einzelne Teile dieser Umgebung mit relativ wenig Arbeitsaufwand auszutauschen oder zu erweitern.

3. Einstieg in die objektorientierte Softwareentwicklung durch das verfügbare Personal

Start des Projektes war im August 2001. Der Pilotversuch an ausgesuchten Schulen fing mit Beginn des Schuljahres 2002/2003 am 19. August 2002 an. Die Pilotschulen wurden regelmäßig informiert und in die Entwicklung mit eingebunden. Ebenso wurde Kontakt zu den Herstellern von Schulverwaltungsprogrammen gehalten. Das Kernprojektteam sollte aus 4 Softwareentwicklern bestehen. Neben der Fach- und

DV-Analyse wurde im September 2001 mit der Ausbildung begonnen. In wöchentlich stattfindenden, jeweils ca. zweistündigen Vorträgen wurden die Themen Objektorientierung, Java und XML angerissen. Im November 2001 erfolgte eine vierwöchige Blockausbildung mit den Ausbildungseinheiten:

- Java Grundkonzepte
- Java Workshop (Java, Swing und JDBC)
- Java und XML
- Java und XML Workshop, erweiterte Konzepte

Neben Java und XML wurden in dieser Blockausbildung Schwerpunkte auf objektorientierte Konzepte gelegt.

Projektverlauf

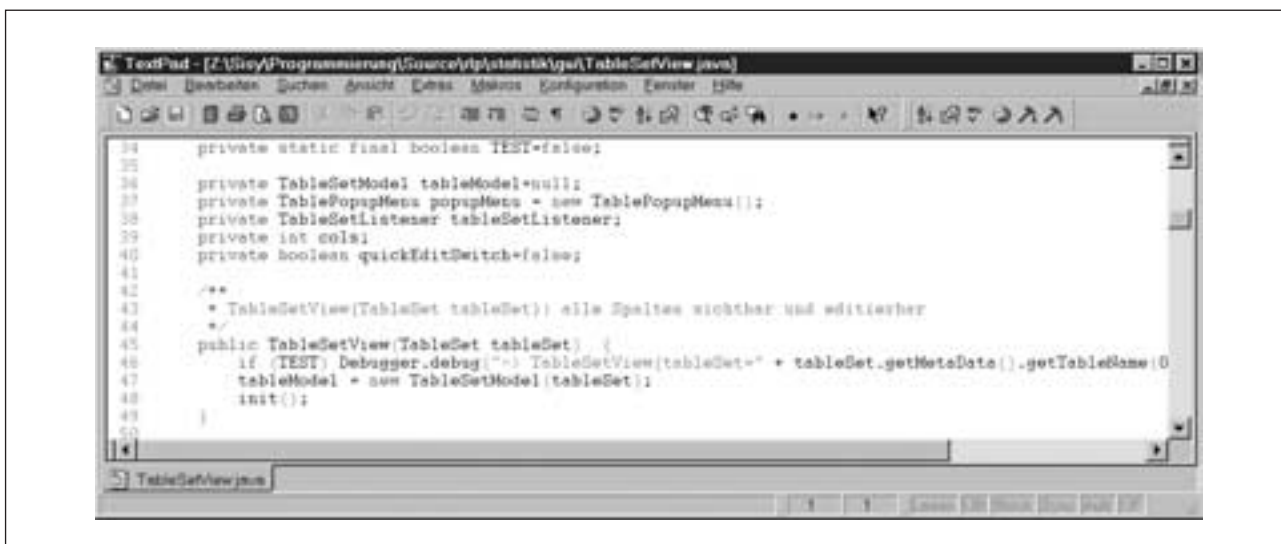
Ein erstes organisatorisches Problem war, durch Umschichtung von Arbeitsabläufen, die Projektmitglieder von bisherigen Aufgaben zu befreien. Dies führte dazu, dass ein Projektmitglied erst im Februar 2002 die Arbeit für das Projekt aufnehmen konnte und ein weiterer vorgesehener Mitarbeiter nicht dem Projekt zur Verfügung stand. Dieser Umstand wurde durch einen vom Statistischen Landesamt neu eingestellten Mitarbeiter, der durch seine Ausbildung direkt ins Projektteam aufgenommen werden konnte, gemildert.

Eine deutliche Entlastung der Entwickler wurde durch die starke Projekteinbindung von Mitarbeitern des Fachbereiches erzielt. Neben Beratung und Lösung von fachlichen Problemen, erledigten diese Mitarbeiter viele organisatorische Aufgaben, wie beispielsweise das Organisieren von Treffen mit den Pilotschulen und Herstellern von Schulverwaltungsprogrammen, Erstellen von Protokollen, Absprachen mit den Schulaufsichtsbehörden, usw.

Ein Wunsch des Projektteams war es, dass die Arbeitsplätze während der Projektphase räumlich nahe, am Besten in einem Zimmer, eingerichtet wurden. Durch die anfängliche Reduzierung der Gruppe auf drei Mitarbeiter, konnten diese in einem Arbeitszimmer untergebracht werden. Damit war der Informationsfluss innerhalb des Teams optimal gewährleistet. Verständ-

12) Siehe hierzu: <http://jakarta.apache.org/ant>

13) Siehe hierzu: <http://www.cvshome.org>



nisfragen und programmiertechnische Schwierigkeiten konnten so zeitnah innerhalb der Gruppe ausdiskutiert werden.

Durch die Zusammensetzung der Projektgruppe mit IT-Fachpersonal aus heterogenen DV-Teilgebieten, Host und strukturierte Programmierung auf der einen, sowie Java und Objektorientierung auf der anderen Seite, ergaben sich anfangs Verständigungsschwierigkeiten. Die Einarbeitung in die neuen Technologien erfolgte aber erstaunlich schnell. Eine 3-schichtige Architektur mit einer Trennung von Oberflächenverarbeitung, Datenhaltung und fachlichen Anforderungen wurde weitestgehend eingehalten. Viel Wert wurde auf eine so weit wie mögliche (bzw. sinnvolle) Verallgemeinerung der fachlichen Problemstellungen gelegt, so dass allgemein gültige Klassen erzeugt wurden, welche in Folgeprojekten wieder verwendet werden können. Durch diese Vorgehensweise konnten sichtbare Ergebnisse für den Fachbereich nicht sehr schnell vorgelegt werden. Diese Anfangsinvestition zahlte sich jedoch im weiteren Projektverlauf durch eine gesteigerte Produktivität aus. Allerdings öffneten sich hierdurch des öfteren Fallen für Neueinsteiger der objektorientierten Programmierung, wie:

- eine starke Kopplung der einzelnen Objekte, da teilweise eine klare Trennung der Zuständigkeiten der einzelnen Klassen fehlte
- strukturierte, statt objektorientierte Programmierung mit Java

Diese Probleme wurden erkannt und gemeinsam behoben. Fertig gestellte Funktionalitäten wurden dem Fachbereich zum Test im Intranet des Statistischen Landesamtes über Java Web Start bereit gestellt, um so Erfahrungen mit Java Web Start zu sammeln. Schwierigkeiten ergaben sich bei Java Web Start im Zusammenspiel mit der Software Internet Information Server (IIS) von Microsoft, die als Webserver eingesetzt wurde, und für den Zugriff auf Ressource-Dateien (z. B. Bilder für den Bildschirmaufbau).

Diese konnten jedoch schnell durch Parametereinstellungen bzw. Umprogrammierung behoben werden. Ein weiteres schwer wiegendes Problem, welches bis heute nicht behoben werden konnte und erst im Pilotversuch auftrat, war ein Fehlverhalten von Java Web Start mit Internet Information Server und der Firewallsoftware (Checkpoint Firewall 1/Application Firewall) beim Provider des Statistischen Landesamtes, dem Daten- und Informationszentrum Rheinland-Pfalz (DIZ). Eine Umgehung des Problems war kurzfristig nur durch die Nutzung eines anderen Rechners zu erreichen. Das Landesmedienzentrum (LMZ) stellte kurzfristig einen Rechner mit der Betriebssystemsoftware LINUX und einem Apache Webserver zur Verfügung.

Um die Qualität der zu liefernden XML-Daten zu erhöhen, wurde den Herstellern von Schulverwaltungsprogrammen ein von der Projektgruppe erstelltes Validierungsprogramm zur Verfügung gestellt. Mit Hilfe

dieses Programmes konnten die Softwarehäuser eine syntaktische Prüfung ihrer Schnittstellendaten auf Basis der vom Statistischen Landesamt vorgegebenen XML-Schema Beschreibungen vornehmen.

Die Daten aus den Schulen sollten gemäß des ursprünglichen Konzeptes über eine Internetverbindung mit Hypertext Transport Protocol (http) auf den Bildungsserver des Landes Rheinland-Pfalz übertragen werden. Voraussetzung hierfür wäre eine Erweiterung der Webserver-Software für eine Java-Servlet Verarbeitung auf dem Rechner des Landesmedienzentrums. Wegen Zeitdrucks musste diese Konzeption auf File Transfer Protocol (ftp) umgestellt werden. Dies erwies sich im Pilotversuch als ungünstig, da dieses Protokoll teilweise durch die in den Schulen eingesetzte Firewallsoftware nicht unterstützt wird. Für den Flächeneinsatz der Applikation im kommenden Schuljahr wird auf das ursprüngliche Konzept zurückgegriffen.

Fazit

Bei Projektstart waren im Statistischen Landesamt Rheinland-Pfalz nur geringe praktische Kenntnisse über Java und XML vorhanden. Die Projektmitglieder kannten relationale Datenbanken, SQL und unter anderem die Programmiersprache C. Die Kenntnis dieser Programmiersprache erleichterte den Einstieg in Java, da C und Java von der Syntax her ähnlich sind. Die bekannten Konzepte der Datenkapselung kommen Teilen der objektorientierten Programmierung nahe. Trotzdem war die Umstellung von der strukturierten zur objektorientierten Programmierung nicht ganz einfach. Unbedingt erforderlich war es, dass die beteiligten Mitarbeiter von ihren bisherigen Aufgaben befreit wurden. Im Projekt wurde bewusst auf weitergehende Technologien, wie zum Beispiel Enterprise Java Beans (EJB) und die Java 2 Enterprise Edition (J2EE) verzichtet, um eine „Überfrachtung“ mit neuen Technologien zu verhindern. Aus diesem Grund wurde auch auf eine Analyse mit der Unified Modelling Language (UML) verzichtet. Im Projektverlauf zeigte es sich allerdings, dass dies zum Verständnis der Zusammenhänge der einzelnen Klassen und Objekte sehr hilfreich gewesen wäre. Deshalb soll im nächsten Projekt ein entsprechendes UML-Tool eingesetzt werden. Die Entwicklungsumgebung hat sich mit den oben aufgeführten Einschränkungen bewährt und zeigte, dass auch Open Source Produkte eine Alternative zu proprietären Entwicklungstools bieten. Alle Mitarbeiter waren sehr motiviert und haben sich im Projektverlauf eine gute Wissensbasis für weitergehende Projekte in diesen Technologien aufgebaut. Lobenswert war die Zusammenarbeit mit einigen Pilotschulen, die, trotz Mehraufwand für die Ermittlung der Daten, sehr engagiert mit dem Statistischen Landesamt zusammenarbeiteten und Anregungen sowie Verbesserungsvorschläge für den Flächeneinsatz von @SchuLE.rlp ab dem kommenden Schuljahr einbrachten.

Stefan Stahlhofen