



Langzeitarchivierung von strukturierten Daten

Das Schweizerische Bundesarchiv (BAR) ist das „Gedächtnis des Bundesstaates“. Laut dem Bundesgesetz über die Archivierung (BGA, SR 152.1) hat es die dauernd wertvollen Unterlagen des Bundes zu sichern, dauerhaft aufzubewahren, zu erschliessen, benutzbar zu halten und zu vermitteln. Gemäss BGA gilt der Archivierungsauftrag des Bundesarchivs unabhängig vom Informationsträger, an den die Unterlagen gebunden sind. Das Bundesarchiv archiviert seit 1982 auch digitale Daten und verwaltet heute rund sechs Terabytes an digitalen Archivdaten. Im Jahr 2003 wird diese Menge um weitere neun Terabytes anwachsen; mittelfristig wird ein Zuwachs von 20 Terabytes pro Jahr erwartet. Die Fachstelle ARELDA (Archivierung elektronischer digitaler Daten und Akten) führt solche Archivierungen durch, unterstützt und berät die Sicherungsteams des Bundesarchivs sowie die Dienststellen der Bundesverwaltung. Zudem konzipiert und realisiert die Fachstelle im Rahmen des E-Government-Projektes ARELDA langfristige Lösungen für die Langzeitarchivierung digitaler Unterlagen. ARELDA ist eines der fünf Schlüsselprojekte in der E-Government-Strategie des Bundes.

Die Lösungsansätze des Bundesarchivs orientieren sich am Grundsatz der „applikationsinvarianten Archivierung“, bei dem die Unterlagen aus den sie erzeugenden, spezifischen Original-Umgebungen (Software, Hardware, Speicher-, Daten- und Dateiformate) gelöst werden (unter Inkaufnahme von gewissen Verlusten an Information und Authentizität) und für die Archivierung in offene, standardisierte, generische und vollständig dokumentierte Archiv-Umgebungen überführt werden. Dort werden sie in möglichst langen Migrationszyklen (mindestens jeweils 15 Jahre) dauerhaft unterhalten. Es erfolgt prinzipiell keine Archivierung von Funktionalitäten (Software, Hardware).

Trotzdem bleibt die digitale Archivierung wegen des periodischen Konversions- und Migrationsbedarfs arbeits- und kostenintensiv. Dabei bilden die Länge der Migrationszyklen und die „Archivische Datenqualität“ (z. B. die genaue Einhaltung der Archivformatspezifikationen und die Güte der technischen Metadaten) die wesentlichen Erfolgsfaktoren, die bestimmen, ob das digitale Archiv dauerhaft finanziert und unterhalten werden kann. Die dauernde Übernahme von grossen und heterogenen Mengen digitaler Unterlagen stellt deshalb eine grosse Herausforderung dar. Einen besonders wichtigen Stellenwert hat im Bundesarchiv die Archivierung der Datenbanken, weil ihre „applikationsinvariante Archivierung“ besonders komplex und arbeitsintensiv ist, da häufig Daten aus einer Vielzahl unterschiedlichster, herstellereinspezifischer Datenbankprodukte zahlreicher Bundesstellen übernommen werden müssen. Das Projekt SIARD (Software Invariant Archiving of Relational Databases) hatte zum Ziel, in diesem Bereich Kosten sparende und qualitätssteigernde Lösungen zu erarbeiten.



Kunde

Schweiz. Bundesarchiv

Branche

Öffentliche Verwaltungen

Projekt

SIARD (Software Invariant Archiving of Relational DBs)

Themen

Application Development

Welche Herausforderung galt es zu meistern?

Die Langzeitarchivierung von Datenbanken im Bundesarchiv hat vier Hauptziele: Die aus unterschiedlichsten Quellsystemen archivierten Daten sollen (1) möglichst authentisch (d.h. in Form und Inhalt möglichst nahe am Original) erhalten werden, (2) möglichst lange (d. h. 10 bis 20 Jahre) ohne permanenten Betreuungs- und Migrationsaufwand „lagerbar“ sein, (3) dauerhaft inhaltlich sowie in ihrem Entstehungs- und Nutzungszusammenhang verständlich bleiben und (4) jederzeit möglichst komfortabel durch die Kunden des Bundesarchivs benutzt werden können. Es ist offensichtlich, dass sich diese Ziele untereinander erheblich konkurrenzieren.

Vor Beginn dieses Projektes wurden Daten in „Flat Files“ archiviert, also in semantisch und syntaktisch weit gehend unstrukturierten reinen Textdateien mit unverbundenen Einzeltabellen. Logische Verknüpfungen zwischen einzelnen Tabellen sowie Views gingen bei dieser Methode verloren. Zusätzliche Informationen sowie Erläuterungen zu Schlüsselwörtern, Codetabellen, Recordlängen, Check Constraints etc. mussten mit grossem Aufwand und fehleranfällig in begleitenden Papierdokumenten beschreibend festgehalten werden und waren teilweise widersprüchlich und unvollständig. Ein wesentliches Problem lag auch darin, dass für die „Flat Files“ keine einheitliche Normierung bezüglich Trennzeichen, numerische Zahlenformate etc. existierten. Unter diesen Bedingungen archivierte Datenbanken sind nicht nur wenig authentisch, sie lassen auch nur eine sehr schwerfällige und eingeschränkte Benutzung für Kunden des Bundesarchivs zu.

Dieses Projekt hatte im Wesentlichen drei Ziele: eine **verbesserte Datenintegrität** sicherzustellen, d.h., die für das Verständnis der Daten notwendigen Metadaten (Keywordlisten, Check Constraints, Codetabellen, Recordlängen, Masseinheiten etc.) sollten in einer Weise dokumentiert werden, die fehlerhafte, fehlende und widersprüchliche Angaben verhindert. Wenn möglich, sollten diese technischen Metadaten nicht einen „artfremden“ Zusatz zu den Daten bilden, sondern innerhalb des Archivformats bereits einen integralen Bestandteil der Kodierung der Datenstruktur darstellen. Ein weiteres Ziel war, die **Ablieferung der Daten aus dem produktiven System an das Bundesarchiv** für alle beteiligten Stellen effizienter zu gestalten und zu normieren, indem Datenexport und Dokumentation automatisiert und durch einen einheitlichen Workflow gesteuert werden. Ein drittes Ziel war eine **bessere und flexiblere Benutzbarkeit**: Die

unabhängig von einer spezifischen Datenbanksoftware archivierten Datenbanken sollten mit tolerierbarem manuellem Aufwand (bevorzugt automatisiert) wieder in eine beliebige relationale Datenbank-Applikation geladen und mit einer Standardfunktionalität benutzt werden können.

Was wurde umgesetzt?

Eine Vorgabe zur Lösung dieser Probleme war, dass als Archivformat zur Beschreibung der logischen Datenbankstruktur SQL-3 (ISO/IEC 9075) als offen und vollständig dokumentiertes, nicht herstellerabhängiges und trotzdem datenbanknahes und von der Industrie breit unterstütztes Format verwendet werden muss. Damit das Archivformat eindeutig ist und damit später konsistente Migrationen in zukünftige, neue Archivformate ermöglicht, sollte es sich jedoch um „reines“ (resp. generisches) SQL-3 handeln, das sich exakt an den ISO-Standard hält.

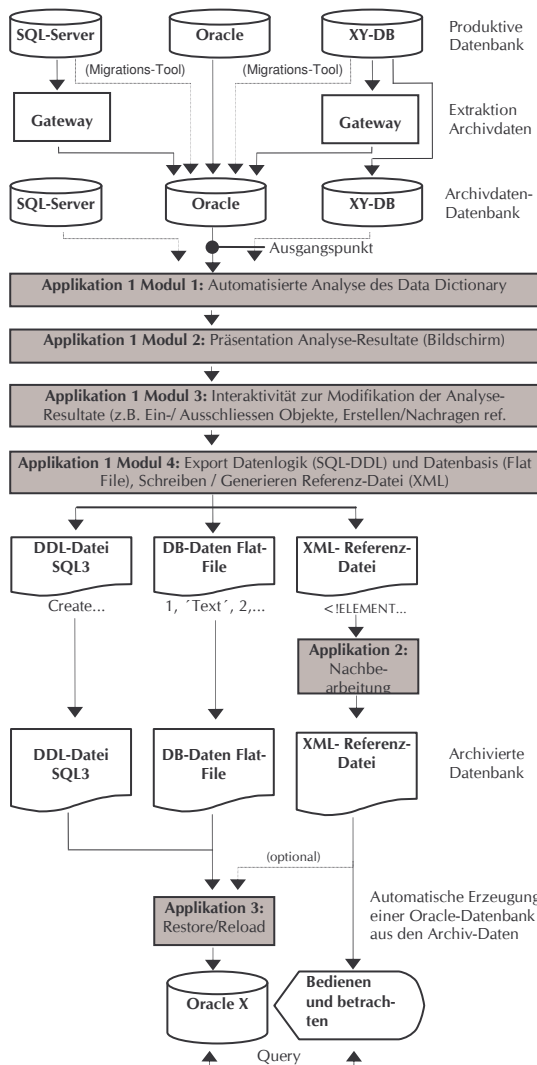
Dies stellte insofern ein Problem dar, als die meisten Hersteller von Datenbanksystemen den ISO-Standard nicht genau einhalten resp. ihre Produkte durch herstellerspezifische Features ergänzen (z. B. eigene Datentypen). Vorgesehen war der Einsatz von SQL-3 für die Kodierung der Datenlogik (Datenbankstruktur), von normierten „Flat-Files“ für die eigentlichen Primärdaten pro Datenbanktabelle sowie von XML für die Kodierung von beschreibenden Kontext-Metadaten, welche nicht in der Datenbank hinterlegt sind.

Aus diesem Grund wurde zuerst in einer Machbarkeitsanalyse und später durch die Entwicklung eines Prototypen überprüft, wie erheblich die zu erwartenden Schwierigkeiten und Verluste an Information und Authentizität sind, welche sich aus diesen herstellerspezifischen SQL-„Flavours“ für die Umsetzung der Lösungsziele ergeben würden.

Der Vorschlag wurde durch die Machbarkeitsanalyse positiv beurteilt. In der Prototypen-Phase beschränkte man sich auf das Erzeugen der Datenbank-Archive und deren Reload, d. h. das Rückladen der oben beschriebenen, rein text-basierten Datenbank-Archive in relationale Datenbanken, mit Oracle 8i. Ausserdem wurde ein umfangreiches Dokument erstellt, das die Abweichungen des Oracle-SQL Syntax gegenüber dem ISO-Standard (SQL-3) analysiert. Dieses Dokument wurde später noch mit dem Fokus auf Microsoft SQL-Server erweitert.

März 2003 - 2/4

Während der Realisierungsphase wurde die als Framework definierte Schnittstelle für die Data-Dictionary-Analyse auch noch für Microsoft SQL-Server implementiert. Neben diesen



zwei „Experten-Modi“ für Oracle und MS SQL-Server stellt ein zusätzlicher „generischer Modus“ eine produktunabhängige Implementierung dieser Schnittstelle mit Hilfe des JDBC-Metadaten-API (Datenbankschnittstelle für die Programmiersprache Java) zur Verfügung. Dieser hat gegenüber den Experten-Modi zwar einen reduzierten Archivierungsumfang, ermöglicht dafür aber grundsätzlich das Archivieren aus beliebigen Datenbankenprodukten, für welche ein JDBC-API verfügbar ist (inkl. Microsoft-Access).

Auch das Dokumentieren der Metadaten, die Applikation zur Beschreibung des originalen Entstehungs- und Verwendungszusammenhangs der Datenbank sowie die Möglichkeit zur Integration von externen Dokumenten (z. B. Handbücher oder Entwicklungsdokumentationen im Format TIFF oder PDF) ins Datenbank-Archiv wurden implementiert. Zudem wurde die Funktionalität des Reload Tools stark erweitert. Entstanden sind drei Applikationen, deren Zusammenspiel in folgender Grafik gezeigt wird:

Applikation 1: Archivierung

Analyse der Datenbankstruktur und grafische Darstellung als GUI-Baumstruktur. Mögliche Integritätsverletzungen und Abweichungen vom ISO-Standard SQL-3 werden angezeigt. Durch Ausschliessen, Modifizieren und Erzeugen von Objekten können allfällige Mängel behoben werden. (Änderungen werden zwecks Nachweisbarkeit in einem Logbuch aufgezeichnet). Erst jetzt ist das Erzeugen des Datenbank-Archivs möglich.

Applikation 2: Nachbearbeitung

Dokumentation des erstellten Datenbank-Archivs: Eingabe von Zusatzinformationen in definierte Metadatenfelder (XML), z. B. Klartext-Auflösung von Keywords sowie Codelisten, Freitextbeschreibung zum Inhalt von Tabellen, ergänzende Angaben, z.B. zur abliefernden Amtsstelle und zur nicht archivierten Applikationslogik; Überprüfung der Konsistenz: erst wenn alle Mussfelder ausgefüllt sind, kann das Archiv abgeschlossen werden.

Applikation 3: Reload

Reload eines Datenbank-Archivs in eine Oracle-Instanz. Möglichkeit zum Abfragen der Primärdaten, inkl. Absetzen beliebiger SQL-Queries. Alle technischen Metadaten sowie die in Applikation 2 erfassten beschreibenden Metadaten zu den einzelnen Objekten werden automatisch angezeigt (XML). Auch das Laden mehrerer Datenbank-Archive ist möglich (inkl. Queries über mehrere Archive).

Technologiehighlights

Alle im Datenbank-Archiv vorhandenen Dateien sind reine Textdateien (UTF-16/Unicode). Die XML-Dateien besitzen ein XML-Schema und sind damit validierbar. Diverse Aufgaben werden mit XSLT gelöst. Da die Applikationen auf unterschiedlichen Plattformen (Windows, Linux, Solaris) lauffähig sein müssen, wurden sie mit der Programmiersprache Java entwickelt. Für das GUI wurden die Java-eigenen Klassenbib-

liotheken (Swing) verwendet. Die Datenbankanbindungen laufen über JDBC. Zusätzlich zu den Experten-Modi für Oracle und MS SQL-Server können dynamische (ohne Codeänderungen) Experten-Modi für die Archivierung aus anderen Datenbankprodukten angebunden werden.

Was ist der Nutzen für den Kunden?

Durch die Lösung können Daten aus unterschiedlichsten herstellereigenen Quelldatenbanken sehr viel authentischer, vollständiger und konsistenter langzeit-archiviert werden als bisher. Das neue Archivformat (reines SQL-3 nach ISO-Standard) ist zudem normiert, vollständig dokumentiert und herstellerunabhängig. Dies ermöglicht eine langfristige Lagerbarkeit der Datenbankarchive ohne grossen Unterhalt sowie im Bedarfsfall zukünftige Migrationen in ein neues Archivformat. Damit können die langfristigen Aufwände für Konversionen und Migrationen erheblich reduziert werden. Aktuell sind Migrationszyklen von voraussichtlich 10 bis 20 Jahren notwendig, welche effizient durchgeführt werden können.

Durch die Lösung werden die Datenintegrität und die Datenqualität bei der Archivierung verbessert, d.h. fehlerhafte,

fehlende und widersprüchliche Angaben können minimiert werden. Die Übernahme der Daten aus den Quellsystemen der Verwaltungsstellen des Bundes ins Bundesarchiv wird durch die Lösung weitgehend automatisiert und vereinheitlicht und damit kostengünstiger. Durch die Lösung wird die Benutzbarkeit von archivierten Datenbanken für Kunden des Bundesarchivs erheblich vereinfacht und verbessert.

Systeme

- Solaris 7 / 8
- Red Hat Linux ab 7
- Windows NT / 2000 / XP

Datenbanken

- Oracle 7 / 8 / 9
- Microsoft SQL-Server 7 / 2000
- Microsoft Access

Entwicklungstools und Bibliotheken

- Eclipse Platform 2.0
- J2SDK (Java 2 Software Development Kit) 1.4, inkl. Swing
- JDBC (Java Database Connection) 3.0
- AXP (JavaTM API for XML Processing) 1.2

Glossar

Codetabellen	Abkürzungen (Codes) und ihre Beschreibung
Data-Dictionary	Gesamte Metadaten der Datenbank
Framework	Gerüst einer Applikation, spezifische Implementationen werden aus dem Framework heraus aufgerufen.
GUI	Graphical User Interface - Grafische Benutzeroberfläche
Keywordlisten	Schlüsselworttabellen
Klassenbibliothek (Swing)	Funktionsbibliothek für GUIs
Kontext-Metadaten	Daten, welche den Aufbau einzelner DB-Objekte beschreiben
Objekte	Datenbankobjekte (Schemas, Tabellen, Columns...)
Queries	Datenabfragebeschreibung
SQL-DDL	Structured Query Language / Data Definition Language / DDL ist ein Teil von SQL
Syntax	(Programmier-) Sprach-Deklaration
Wertbereiche	Min./Max-Wert
XSLT	eXtensible Stylesheet Language for Transformation – Deklarationsprache zur Umwandlung von XML-Dateien in andere Formate.

Trivadis AG
 Elisabethenanlage 9
 CH-4051 Basel
 Phone-No. +41 61 279 97 55
 Fax +41 61 279 97 56

Trivadis GmbH
 Industriestrasse 4
 D-70565 Stuttgart
 Phone-No. +49 711 90 36 32 30
 Fax +49 711 90 36 32 59

Trivadis GmbH
 Millennium Tower, Handelskai 94-96
 A-1200 Wien
 Phone-No. +43 1 332 35 31
 Fax +43 1 332 35 34

März 2003 - 4/4