

eCH-0165 SIARD-Formatspezifikation

Name	SIARD-Formatspezifikation
Standard-Nummer	eCH-0165
Kategorie	Standard
Reifegrad	Implementiert
Version	1.0
Status	Genehmigt
Genehmigt am	2013-03-06
Ausgabedatum	2013-03-21
Ersetzt Standard	--
Sprachen	Deutsch (Original), Französisch (Übersetzung)
Autoren	<p>Hedi Bruggisser, Staatsarchiv Thurgau, hedi.bruggisser@tg.ch</p> <p>Georg Buechler, KOST, georg.buechler@kost.admin.ch</p> <p>Alain Dubois, Staatsarchiv Wallis, alain.dubois@admin.vs.ch</p> <p>Martin Kaiser, KOST, martin.kaiser@kost.admin.ch</p> <p>Lambert Kansy, Staatsarchiv Basel Stadt, lambert.kansy@bs.ch</p> <p>Markus Lischer, Staatsarchiv Luzern, markus.lischer@lu.ch</p> <p>Claire Röthlisberger-Jourdan, KOST, claire.roethlisberger@kost.admin.ch</p> <p>Hartwig Thomas, Enter AG, hartwig.thomas@enterag.ch</p> <p>Andreas Voss, Schweizerisches Bundesarchiv, andreas.voss@bar.admin.ch</p>
Herausgeber / Vertrieb	<p>Verein eCH, Mainaustrasse 30, Postfach, 8034 Zürich T 044 388 74 64, F 044 388 71 80 www.ech.ch / info@ech.ch</p>

Zusammenfassung

Dieses Dokument enthält die Spezifikation des SIARD-Dateiformats. SIARD steht für *Software-Independent Archival of Relational Databases*. Das Format wurde vom Schweizerischen Bundesarchiv entwickelt. Es handelt sich um eine normative Beschreibung eines Dateiformats für die langfristige Erhaltung von relationalen Datenbanken.

Das SIARD-Format basiert auf Standards – u. a. auf den ISO-Normen Unicode und XML und den Industriestandards SQL1999 und ZIP. Die Verwendung international anerkannter Standards zielt darauf hin, die langfristige Erhaltung von und den Zugang zu dem weitverbreiteten relationalen Datenbankmodell zu gewährleisten.

Inhaltsverzeichnis

1	Status des Dokuments	5
2	Einleitung	5
2.1	Struktur des Dokuments	5
2.1.1	Aufbau Kapitel	5
2.1.2	ID Anforderungen	6
2.1.3	Unterscheidung zwischen Muss- und Kann-Anforderungen.....	6
2.1.4	Notation Ordner, Dateien und Ordnerstrukturen	6
2.2	Adressaten/Zielgruppe.....	7
2.3	Ausgangslage.....	7
2.4	Abgrenzungen	7
3	Allgemeine Anforderungen / Grundsätze	9
3.1	Verwendung von Standards.....	9
3.2	Datenbanken als Unterlagen	9
3.3	Zeichensätze und Zeichen.....	9
3.4	Bezeichner und reguläre Bezeichner	10
4	Anforderungen an die Formatstruktur	11
4.1	Aufbau der SIARD-Archivdatei.....	11
4.2	Struktur der SIARD-Archivdatei	11
4.3	Korrespondenz zwischen Metadaten und Tabellendaten	14
5	Anforderungen an die Metadaten	18
5.1	Metadaten auf der Ebene Datenbank	18
5.2	Metadaten auf der Ebene Schema	19
5.3	Metadaten auf der Ebene Tabelle	20
5.4	Metadaten auf der Ebene Spalte	21
5.5	Metadaten des Primärschlüssels	22
5.6	Metadaten der Fremdschlüssel.....	22
5.7	Referenz-Metadaten	23
5.8	Metadaten des Kandidatenschlüssels.....	23
5.9	Metadaten der Check-Einschränkung.....	23
5.10	Metadaten auf der Ebene Trigger	24
5.11	Metadaten auf der Ebene View	24

5.12 Metadaten auf der Ebene Routine	25
5.13 Metadaten der Parameter	26
5.14 Metadaten auf der Ebene des Benutzers	26
5.15 Metadaten auf der Ebene Rolle	26
5.16 Metadaten auf der Ebene der Privilegien	27
6 Anforderungen an die Tabellendaten.....	28
6.1 Tabellen-Schemadefinition	28
6.2 Tabellendaten.....	29
7 Version und Gültigkeit der Spezifikation	31
8 Change-Management-Prozess.....	31
9 Haftungsausschluss/Hinweise auf Rechte Dritter.....	31
10 Urheberrechte.....	32
Anhang A – Mitarbeit & Überprüfung.....	33
Anhang B – Abkürzungen und Glossar	34
Anhang C – Nachweis der verwendeten Standards.....	36
Anhang D – XML-Schemadefinitionen	36
D.1 metadata .xsd.....	36
D.2 Beispiel für metadata .xml.....	37
D.3 Beispiel für die XML-Schemadefinition einer Tabelle: table0.xsd.....	42
D.4 Beispiel für die Tabellendaten einer Tabelle: table0.xml	42

1 Status des Dokuments

Das vorliegende Dokument wurde vom Expertenausschuss am 06.03.2013 **genehmigt**. Es hat für das definierte Einsatzgebiet im festgelegten Gültigkeitsbereich normative Kraft.

2 Einleitung

2.1 Struktur des Dokuments

2.1.1 Aufbau Kapitel

Jedes Kapitel in dieser Spezifikation ist nach demselben Muster aufgebaut. Nach einer kurzen Einleitung werden die Anforderungen in einer Tabelle aufgeführt.

ID	Beschreibung Anforderung	M/K
Enthält die ID der Anforderung	Enthält den Anforderungstext	Definiert, ob es sich um eine Muss- oder Kann-Anforderung handelt

Eine Anforderung wird häufig durch Empfehlungen, Hinweise und Beispiele erläutert. Empfehlungen, Hinweise und Beispiele sind speziell gekennzeichnet.

ID	Beschreibung Anforderung	M/K
A_3.1-1	<p>Anforderungstext</p> <p>Beispiel Beispieltext</p> <p>Hinweis Hinweistext</p> <p>Empfehlung <i>Empfehlungstext ist kursiv gesetzt.</i></p>	M

2.1.2 ID Anforderungen

Die Anforderungen sind über eine ID eindeutig identifizierbar.

ID
A_3.1-1

Diese ID ist nach dem folgenden Muster aufgebaut:

- A_ Buchstabe + _ identifiziert Hauptkapitel
 - A_ = Allgemeine Anforderungen / Grundsätze
 - T_ = Anforderungen an die Tabellendaten
 - M_ = Anforderungen an die Metadaten
 - P_ = Anforderungen an die Paketstruktur

- 3.1-1 Die Nummer beginnt mit der Angabe des Kapitels (Gruppierung der Anforderungen zum gleichen Thema), die Zahl hinter dem Bindestrich wird durchnummeriert und kennzeichnet so alle Anforderungen des Kapitels.

2.1.3 Unterscheidung zwischen Muss- und Kann-Anforderungen

Jede Anforderung ist entweder eine Muss- oder eine Kann-Anforderung. Dies wird mit einem Buchstaben kenntlich gemacht, der auf die Verbindlichkeit verweist:

Abkürzung	Bedeutung
M	Muss-Anforderung Diese Anforderung muss erfüllt sein, um eine gültige SIARD-Datei zu erhalten.
K	Kann-Anforderung Diese Anforderung sollte erfüllt sein. Sie vereinfacht das Handling im Sinne von <i>Best Practice</i> .

2.1.4 Notation Ordner, Dateien und Ordnerstrukturen

Für die Notation von Ordnern, Dateien etc. werden die folgenden Symbole und Parameter verwendet.

Symbol	Bedeutung
/	Ordner
header/	Ein Ordner mit dem Namen «header»
xy.txt	Datei (mit Datei-Endung «txt»)
dir1/	Beispiel-Ordner (in roter Farbe)
abc.pdf	Beispiel-Dateien (in roter Farbe)
...	Platzhalter für Dateien oder Ordner, die für die Erklärung nicht relevant sind.

Symbol	Bedeutung
[]	Platzhalter für einen Ausdruck oder einen Basistyp wie «string», «integer» etc.
<xx>	Platzhalter für beliebige Zeichenkette

2.2 Adressaten/Zielgruppe

Dies ist ein technisches Dokument für IT-Spezialisten, die im Bereich der dauerhaften Archivierung von relationalen Datenbanken tätig sind.

2.3 Ausgangslage

Die Bezeichnung SIARD steht für Software-Independent Archival of Relational Databases (engl. für „software-unabhängige Archivierung von relationalen Datenbanken“). Es handelt sich um ein offenes Dateiformat zur dauerhaften Archivierung von relationalen Datenbanken in Form von Textdaten, basierend auf XML, die in eine Containerdatei (SIARD-Archiv) gepackt werden¹.

Dauerhafte Archivierung meint die grundsätzlich unbegrenzte Aufbewahrung der in SIARD-Dateien gespeicherten Informationen unter Erhalt des Bitstroms sowie der Fähigkeit, die Daten menschenlesbar und verständlich zu interpretieren und darzustellen.

Wenn Struktur und Inhalt einer relationalen Datenbank ins SIARD-Format übersetzt werden, wird es später jederzeit möglich sein, auf die Daten der Datenbank zuzugreifen, selbst wenn die ursprüngliche Datenbanksoftware nicht mehr verfügbar oder nicht mehr lauffähig sein wird. Dies wird erreicht, indem für das SIARD-Format geeignete Standards verwendet wurden, die international breit abgestützt sind. Diese langfristige Interpretierbarkeit der Datenbankinhalte beruht im Wesentlichen auf den beiden Standards XML und SQL:1999.

2.4 Abgrenzungen

Es ist festzuhalten, dass das SIARD-Format nur das Langzeitspeicherformat für eine spezielle Sorte von digitalen Unterlagen (relationale Datenbanken) darstellt und somit völlig unabhängig von Paketstrukturen wie SIP (Submission Information Package), AIP (Archival Information Package) und DIP (Dissemination Information Package) des OAIS-Modells konzipiert ist.

Es wird davon ausgegangen, dass eine Datenbank im SIARD-Format als Teil eines Informationspakets zusammen mit anderen Unterlagen (Dokumentation, für das Verständnis der Datenbank relevante Geschäftsunterlagen, ...) archiviert wird.

¹ Das Datenbank-Archivierungsformat SIARD ist zu unterscheiden von der Applikation SIARD Suite. Diese wurde vom Schweizerischen Bundesarchiv BAR entwickelt, um SIARD-Dateien zu erzeugen, zu editieren und wieder in Datenbankumgebungen zu importieren.

Ähnlich wie eine XML-basierte Word- oder Mail-Datei eine interne Dateistruktur mit Metadaten, Primärdaten und verschiedenen Hilfsdaten enthält, enthält auch eine archivierte relationale Datenbank im SIARD-Format neben den eigentlichen Tabellendaten auch eigene Metadaten, welche die Unterlage näher beschreiben – ohne Rücksicht auf den Metadatenkatalog, den ein Archiv in seinen OAIS-Paketen erfasst.

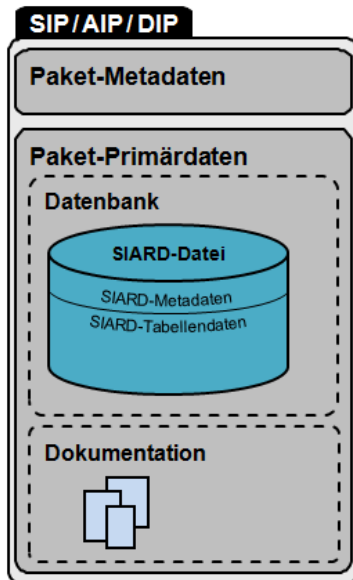


Abb. 1: Symbolbild eines Informationspakets mit einer enthaltenen SIARD-Datei

3 Allgemeine Anforderungen / Grundsätze

3.1 Verwendung von Standards

Um die Interpretierbarkeit der Datenbankinhalte über lange Zeiträume zu gewährleisten, beruht das SIARD-Format im Wesentlichen auf den beiden ISO-Standards XML sowie SQL:1999.

ID	Beschreibung Anforderung	M/K
A_3.1-1	Sämtliche Datenbankinhalte werden in einer Kollektion von XML-Dateien gemäss ISO/IEC 19503:2005 gespeichert. Schemadefinitionen und SQL-Code müssen jeweils SQL:1999-konform sein gemäss ISO/IEC 9075. Einzige Ausnahme sind BLOB- und CLOB-Daten (Binary Large Objects und Character Large Objects) ab einer bestimmten Speichergrösse (siehe T_6.2-4), die in separaten binären Dateien gespeichert, aber in den XML-Dateien referenziert werden.	M

3.2 Datenbanken als Unterlagen

Eine relationale Datenbank wird wie eine einzige zu archivierende Unterlage behandelt, damit die Bezüge (Referenzen) zwischen den Daten einzelner Tabellen erhalten bleiben.

ID	Beschreibung Anforderung	M/K
A_3.2-1	Eine relationale Datenbank wird in einer einzigen SIARD-Datei archiviert.	M
A_3.2-2	Die Primärdaten einer relationalen Datenbank werden vollständig in einer SIARD-Datei archiviert. Das heisst, dass jede SELECT-Abfrage auf den originalen und auf den archivierten Daten dieselben Resultate ergibt.	M

3.3 Zeichensätze und Zeichen

ID	Beschreibung Anforderung	M/K
A_3.3-1	Generell werden alle Daten im Unicode-Zeichensatz gemäss ISO 10646 gespeichert.	M
A_3.3-2	Beim Extrahieren aus Datenbanken, welche andere Zeichensätze unterstützen, wird die Abbildung in die entsprechenden Unicode-Zeichensätze vorgenommen. Aus diesem Grund müssen die nationalen Zeichenketten-Typen (NCHAR, NVARCHAR, NCLOB) aus dem Datenbank-Produkt generell in nicht-nationale (CHAR, VARCHAR bzw. CLOB) übersetzt werden. Diese Konvention wird von XML unterstützt, unabhängig davon, ob eine XML-Datei im UTF-8 Format oder im UTF-16-Format gespeichert wird.	M
A_3.3-3	In den XML-Dateien des SIARD-Formats werden alle Zeichen, welche in der XML-Syntax eine spezielle Bedeutung haben, durch Einheitenreferenzen ersetzt und zwar in allen Feldern vom Typ xs:string. Zusätzlich werden die Unicode-Steuerzeichen 0-31 und 127-159 mit Hilfe des Solidus ("") codiert, damit die Gültigkeit der XML-Datei garantiert bleibt.	M

ID	Beschreibung Anforderung	M/K																						
A_3.3-4	<p>Zeichen, die nicht in UNICODE dargestellt werden können (Codes 0-8, 14-31, 127-159), sowie das Escapezeichen '\ ' und mehrere aufeinanderfolgende Leerschläge werden mit Escape als \u00<xx> in XML dargestellt. Anführungszeichen, Kleiner und Et-Zeichen werden in XML als Einheitsreferenzen dargestellt.</p> <table border="1"> <thead> <tr> <th>Ursprüngliche Zeichen</th> <th>Zeichen im SIARD- Format</th> </tr> </thead> <tbody> <tr> <td>0 bis 8</td> <td>\u0000 bis \u0008</td> </tr> <tr> <td>14-31</td> <td>\u000E bis \u001F</td> </tr> <tr> <td>32</td> <td>\u0020, falls mehrere aufeinanderfolgen</td> </tr> <tr> <td>"</td> <td>&quot;</td> </tr> <tr> <td>&</td> <td>&amp;</td> </tr> <tr> <td>'</td> <td>&apos;</td> </tr> <tr> <td><</td> <td>&lt;</td> </tr> <tr> <td>></td> <td>&gt;</td> </tr> <tr> <td>\</td> <td>\u005c</td> </tr> <tr> <td>127 bis 159</td> <td>\u007F bis \u009F</td> </tr> </tbody> </table>	Ursprüngliche Zeichen	Zeichen im SIARD- Format	0 bis 8	\u0000 bis \u0008	14-31	\u000E bis \u001F	32	\u0020, falls mehrere aufeinanderfolgen	"	"	&	&	'	'	<	<	>	>	\	\u005c	127 bis 159	\u007F bis \u009F	M
Ursprüngliche Zeichen	Zeichen im SIARD- Format																							
0 bis 8	\u0000 bis \u0008																							
14-31	\u000E bis \u001F																							
32	\u0020, falls mehrere aufeinanderfolgen																							
"	"																							
&	&																							
'	'																							
<	<																							
>	>																							
\	\u005c																							
127 bis 159	\u007F bis \u009F																							

3.4 Bezeichner und reguläre Bezeichner

In SQL:1999 gibt es reguläre Bezeichner² ohne Leerschläge und Sonderzeichen, für welche Gross- und Kleinschreibung unwichtig ist, die aber in Grossbuchstaben gespeichert werden, und Bezeichner in Anführungszeichen³, für welche die Schreibweise eindeutig ist, und die auch Sonderzeichen enthalten dürfen. Diese werden in Ausdrücken von doppelten Anführungszeichen umrahmt.

Was ein Sonderzeichen ist, bestimmt der SQL-Standard. Was die Grossbuchstabenversion eines Buchstabens ist, wird vom Unicode-Standard bestimmt.

In den Metadaten wird ein regulärer Bezeichner in Grossbuchstaben gespeichert, während ein begrenzter (delimitierter) Bezeichner in Anführungszeichen gespeichert wird. Der SQL:1999-Standard hält fest: Sobald ein Bezeichner ein Zeichen enthält, das er als regulärer Bezeichner nicht enthalten darf, handelt es sich um einen begrenzten Bezeichner.

ID	Beschreibung Anforderung	M/K
A_3.4-1	Generell werden alle Bezeichner im Unicode-Zeichensatz gespeichert.	M
A_3.4-2	Reguläre Bezeichner sind in Grossbuchstaben und ohne Anführungszeichen.	M
A_3.4-3	Begrenzte (delimitierte) Bezeichner sind in Anführungszeichen gesetzt.	M

² „Regulärer Bezeichner“, engl.: *identifier*. Ein SQL:1999-Bezeichner muss mit einem Buchstaben (A-Z) oder dem Tiefstrich (_) beginnen, gefolgt von Buchstaben (A-Z), Ziffern (0-9) oder Tiefstrich (_), maximal 128 Zeichen.

³ „Bezeichner in Anführungszeichen“ bzw. „begrenzter (delimitierter) Bezeichner“, engl.: *delimited identifier*.

4 Anforderungen an die Formatstruktur

4.1 Aufbau der SIARD-Archivdatei

Die SIARD-Archivdatei wird als ZIP-Archiv realisiert.

ID	Beschreibung Anforderung	M/K
A_4.1-1	Die SIARD-Datei wird als ein einziges unkomprimiertes ZIP-Archiv gemäss der von der Firma PkWare publizierten Spezifikation, Version 6.3.2 gespeichert ⁴ .	M
A_4.1-2	Die SIARD-Datei ist nicht passwortgeschützt oder verschlüsselt.	M
A_4.1-3	Für das ZIP-Archiv sind beide Ausprägungen erlaubt, ZIP32 und ZIP64.	M
A_4.1-4	Das ZIP-Archiv hat die Dateierweiterung ".siard".	M

4.2 Struktur der SIARD-Archivdatei

Eine im SIARD-Format archivierte relationale Datenbank besteht aus zwei Komponenten: den Metadaten, welche die Struktur der archivierten Datenbank beschreiben, und den Tabellendaten, welche die Tabelleninhalte repräsentieren. Die Metadaten geben weiterhin an, welche Tabellendaten wo im Archiv zu finden sind.

ID	Beschreibung Anforderung	M/K
P_4.2-1	<p>Die Tabellendaten befinden sich im Ordner <code>content/</code> und die Metadaten im Ordner <code>header/</code>. Weitere Ordner oder Dateien sind nicht erlaubt.</p> <p>Beispiel Aufbau der SIARD-Datei (schematisch)</p> <pre>Northwind.siard content/ header/</pre>	M

⁴ ZIP-Dateien wurden ursprünglich von Phil Katz definiert und sind heute als De-facto-Standard sehr weit verbreitet. Die aktuelle Version 6.3.2 der von der Firma PkWare publizierten Spezifikation findet man unter <http://www.pkware.com/documents/casestudies/APPNOTE.TXT>.

ID	Beschreibung Anforderung	M/K
P_4.2-2	<p>Im Ordner <code>content/</code> befinden sich ein oder mehrere Schema-Ordner, welche die einzelnen Tabellen-Ordner enthalten. Andere Ordner oder Dateien sind nicht erlaubt.</p> <p>Beispiel Aufbau der SIARD-Datei (schematisch)</p> <pre data-bbox="432 510 1318 817"> Northwind.siard content/ schema0/ table0/ table1/ table2/ ... schema1/ table0/ ... </pre> <p>Empfehlung <i>Es wird empfohlen, die Schema- und Tabellenordner zu normalisieren und anstelle des eigentlichen Namen z.B. <code>schema0/</code> und <code>table0/</code> zu verwenden (siehe Einschränkungen unter P_4.2-5).</i></p>	M
P_4.2-3	<p>In den einzelnen Tabellen-Ordnern ist eine XML- und eine XSD-Datei enthalten, wobei die Namen (Ordnerbezeichnung und beide Dateinamen) identisch sein müssen. Weitere Ordner oder Dateien sind mit Ausnahme von BLOB- und CLOB-Ordnern samt deren Inhalt (BIN-oder TXT-Dateien) nicht erlaubt.</p> <p>Beispiel Aufbau der SIARD-Datei (schematisch)</p> <pre data-bbox="432 1272 1318 1709"> Northwind.siard content/ schema0/ table0/ table0.xml table0.xsd lob4⁵/ record0.bin record1.bin table1/ table1.xml table1.xsd ... </pre> <p>Empfehlung <i>Es wird empfohlen, die lob-Ordner und lob-Dateien zu normalisieren und anstelle des eigentlichen Namens z.B. <code>lob4/</code> und <code>record0.bin</code> oder <code>record0.txt</code> zu verwenden.</i></p>	M

⁵ Bei diesem Beispiel enthält die Spalte 4 zusätzliche lob-Dateien, die entsprechend in `lob4/` abgelegt werden.

ID	Beschreibung Anforderung	M/K
P_4.2-4	<p>Im Ordner <code>header/</code> müssen die Dateien <code>metadata.xml</code> und <code>metadata.xsd</code> vorhanden sein. Weitere Dateien, zum Beispiel Stylesheets, sind erlaubt.</p> <p>Beispiel Aufbau der SIARD-Datei (schematisch)</p> <pre data-bbox="432 479 1318 696"> Northwind.siard content/ ... header/ metadata.xml metadata.xsd ... </pre>	M
P_4.2-5	<p>Alle Datei- und Ordnernamen müssen wie folgt aufgebaut sein: Der Name muss mit einem Buchstaben [a-z respektive A-Z] beginnen und darf anschliessend nur folgende Zeichen enthalten:</p> <ul style="list-style-type: none"> • a-z • A-Z • 0-9 • - • . (darf nur für die Trennung zwischen Namen und Extension verwendet werden) <p>Empfehlung <i>Die Länge der Datei- und Ordnernamen sollte möglichst 20 Zeichen nicht überschreiten, damit Schwierigkeiten mit zu grossen Pfadlängen unter Windows vermieden werden können.</i></p>	M

4.3 Korrespondenz zwischen Metadaten und Tabellendaten

<p>P_4.3-1</p>	<p>Die in metadata.xml vorgegebene Struktur muss identisch sein mit jener im Ordner content/.</p> <p>Beispiel</p> <div style="display: flex; justify-content: space-around;"> <div data-bbox="437 443 946 1402" style="border: 1px solid #add8e6; padding: 10px; width: 45%;"> <p style="text-align: center; font-weight: bold;">SIARD-Metadaten</p> <pre> ... <dbname>northwind</dbname> ... <schemas> <schema> <name>admin</name> <folder>schema0</folder> <tables> <table> <name>Products</name> <folder>table0</folder> ... </table> <table> <name>Shippers</name> <folder>table1</folder> ... </table> <table> <name>Orders</name> <folder>table2</folder> ... </table> <table> <name>Categories</name> <folder>table3</folder> ... </table> <table> <name>Customers</name> <folder>table4</folder> </pre> </div> <div data-bbox="978 443 1305 1402" style="border: 1px solid #add8e6; padding: 10px; width: 45%;"> <p style="text-align: center; font-weight: bold;">content-Struktur</p> </div> </div>	<p>M</p>
----------------	--	----------

<p>P_4.3-2</p>	<p>Die in metadata.xml angegebene Anzahl der Spalten einer Tabelle muss identisch sein mit jener der entsprechenden Datei table[Zahl].xsd.</p> <p>Beispiel</p> <div style="display: flex; justify-content: space-around;"> <div data-bbox="432 387 743 999" style="border: 1px solid #add8e6; padding: 5px;"> <p style="text-align: center;">SIARD-Metadaten</p> <pre> ... <dbname>northwind</dbname> ... <schemas> <schema> <name>admin</name> <folder>schema0</folder> <tables> <table> <name>Products</name> <folder>table0</folder> <description> <columns> <column> <column> <column> <column> <column> <column> <column> <column> <column> <column> </columns> <primaryKey> <foreignKeys> <rows>77</rows> </table> ... </pre> </div> <div data-bbox="778 387 1302 999" style="border: 1px solid #add8e6; padding: 5px;"> <p style="text-align: center;">content – table0.xsd</p> <pre> <?xml version="1.0" encoding="UTF-8"?> <xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema" > <xs:element name="table"> <xs:complexType name="rowType"> <xs:sequence> <xs:element name="c1" type="xs:integer"/> <xs:element minOccurs="0" name="c2" type="xs:string"/> <xs:element minOccurs="0" name="c3" type="xs:integer"/> <xs:element minOccurs="0" name="c4" type="xs:integer"/> <xs:element minOccurs="0" name="c5" type="xs:string"/> <xs:element minOccurs="0" name="c6" type="xs:decimal"/> <xs:element minOccurs="0" name="c7" type="xs:integer"/> <xs:element minOccurs="0" name="c8" type="xs:integer"/> <xs:element minOccurs="0" name="c9" type="xs:integer"/> <xs:element minOccurs="0" name="c10" type="xs:boolean"/> </xs:sequence> </xs:complexType> </xs:schema> </pre> </div> </div>	<p>M</p>
----------------	---	----------

P_4.3-3

Die Datentyp-Angaben zu den Spaltendefinitionen in `metadata.xml` müssen identisch sein mit jenen der entsprechenden Datei `table[Zahl].xsd`.

M

Die SQL:1999-Datentypen werden in den Schemadateien `table[Zahl].xsd` gemäss der folgenden Tabelle in XML-Datentypen umgewandelt.

SQL:1999	XML
BINARY LARGE OBJECT	blobType ⁶
BIT VARYING(...)	xs:hexBinary
BIT(...)	xs:hexBinary
BOOLEAN	xs:boolean
CHARACTER LARGE OBJECT	clobType ⁶
CHARACTER VARYING(...)	xs:string
CHARACTER(...)	xs:string
DATE	xs:date
DECIMAL(...)	xs:decimal
DOUBLE PRECISION	xs:float
FLOAT(...)	xs:float
INTEGER	xs:integer
INTERVAL ⁷	
NATIONAL CHARACTER LARGE OBJECT	clobType ⁶
NATIONAL CHARACTER VARYING(...)	xs:string
NATIONAL CHARACTER(...)	xs:string
NUMERIC(...)	xs:decimal
REAL	xs:float
SMALLINT	xs:integer
TIME	xs:time
TIME WITH TIME ZONE ⁷	
TIMESTAMP	xs:dateTime
TIMESTAMP WITH TIME ZONE ⁷	

Beispiel

SIARD-Metadaten

```

...
<name>Products</name>
<folder>table0</folder>
<description/>
<columns>
  <column>
    <name>ProductID</name>
    <type>INTEGER</type>
    <typeOriginal>COUNTER</typeOriginal>
    <nullable>false</nullable>
    <description/>
  </column>
  <column>
    <name>ProductName</name>
    <type>CHARACTER VARYING(40)</type>
    <typeOriginal>VARCHAR(40)</typeOriginal>
    <nullable>true</nullable>
  </column>
...

```

content – table0.xsd

```

<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema" >
  <xs:element name="table">
    <xs:complexType name="rowType">
      <xs:sequence>
        <xs:element name="c1" type="xs:integer"/>
        <xs:element minOccurs="0" name="c2" type="xs:string"/>
        <xs:element minOccurs="0" name="c3" type="xs:integer"/>
        <xs:element minOccurs="0" name="c4" type="xs:integer"/>
        <xs:element minOccurs="0" name="c5" type="xs:string"/>
        <xs:element minOccurs="0" name="c6" type="xs:decimal"/>
        <xs:element minOccurs="0" name="c7" type="xs:integer"/>
        <xs:element minOccurs="0" name="c8" type="xs:integer"/>
        <xs:element minOccurs="0" name="c9" type="xs:integer"/>
        <xs:element name="c10" type="xs:boolean"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

⁶ Zu den XML-Datentypen `blobType` und `clobType` siehe A_3.1-1.

⁷ Die SQL:1999-Datentypen `INTERVAL`, `TIME WITH TIME ZONE`, und `TIMESTAMP WITH TIME ZONE` werden in dieser SIARD-Version noch nicht unterstützt.

<p>P_4.3-4</p>	<p>Die Nullable-Angaben zu den Spaltendefinitionen in metadata.xml müssen identisch sein mit jenen der entsprechenden Datei table[Zahl].xsd.</p> <p>Beispiel</p> <div style="display: flex; justify-content: space-around;"> <div style="border: 1px solid #ccc; padding: 5px; width: 45%;"> <p style="text-align: center;">SIARD-Metadaten</p> <pre> ... <name>Products</name> <folder>table0</folder> <description/> <columns> <column> <name>ProductID</name> <type>INTEGER</type> <typeOriginal>COUNTER</typeOriginal> <nullable>false</nullable> <description/> </column> <column> <name>ProductName</name> <type>CHARACTER VARYING(40)</type> <typeOriginal>VARCHAR(40)</typeOriginal> <nullable>true</nullable> </column> ... </pre> </div> <div style="border: 1px solid #ccc; padding: 5px; width: 45%;"> <p style="text-align: center;">content – table0.xsd</p> <pre> <?xml version="1.0" encoding="UTF-8" ?> <xs:schema attributeFormDefault="unqualified" elementFormDefault="unqualified"> <xs:element name="table"> <xs:complexType name="rowType"> <xs:sequence> <xs:element name="c1" type="xs:integer"/> <xs:element minOccurs="0" name="c2" type="xs:string"/> <xs:element minOccurs="1" name="c3" type="xs:integer"/> <xs:element minOccurs="1" name="c4" type="xs:integer"/> <xs:element minOccurs="1" name="c5" type="xs:string"/> <xs:element minOccurs="1" name="c6" type="xs:decimal"/> <xs:element minOccurs="1" name="c7" type="xs:integer"/> <xs:element minOccurs="1" name="c8" type="xs:integer"/> <xs:element minOccurs="1" name="c9" type="xs:integer"/> <xs:element name="c10" type="xs:boolean"/> </xs:sequence> </xs:complexType> </xs:element> </xs:schema> </pre> </div> </div> <p>Hinweis</p> <p>Die SQL:1999-Notation "<nullable>true</nullable>" wird in XML zu "minOccurs="0"". "<nullable>false</nullable>" entspricht "minOccurs="1"" in XML; da dies jedoch der Standardwert ist, wird er oftmals weggelassen.</p>	<p>M</p>
<p>P_4.3-5</p>	<p>Die Spaltenreihenfolge in metadata.xml muss identisch sein mit jener der entsprechenden table[Zahl].xsd.</p>	<p>M</p>
<p>P_4.3-6</p>	<p>Die Anzahl Zeilen einer Tabelle in metadata.xml muss in den von der entsprechenden table[Zahl].xsd spezifizierten Bereich hineinpassen.</p> <p>Die Anzahl Zeilen einer Tabelle in metadata.xml muss identisch sein mit der Anzahl Zeilen in der entsprechenden table[Zahl].xml.</p> <p>Beispiel</p> <div style="display: flex; flex-direction: column; align-items: flex-start;"> <div style="border: 1px solid #ccc; padding: 5px; width: 45%; margin-bottom: 10px;"> <p style="text-align: center;">SIARD-Metadaten</p> <pre> ... <dbname>northwind</dbname> ... <schemas> <schema> <name>admin</name> <folder>schema0</folder> <tables> <table> <name>Products</name> <folder>table0</folder> <description/> <columns> <primaryKey> <foreignKeys> </table> </table> <rows>77</rows> </table> </schemas> </schema> ... </pre> </div> <div style="display: flex; justify-content: space-around; width: 100%;"> <div style="border: 1px solid #ccc; padding: 5px; width: 45%;"> <p style="text-align: center;">content – table0.xsd</p> <pre> <?xml version="1.0" encoding="UTF-8" ?> <xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" target="http://www.admin.ch/xmlns/siard/1.0/schema0/table0.xsd"> <xs:element name="table"> <xs:complexType name="rowType"> <xs:sequence> <xs:element maxOccurs="unbounded" minOccurs="0" name="row" type="rowType"/> </xs:sequence> </xs:complexType> </xs:element> </xs:schema> </pre> </div> <div style="border: 1px solid #ccc; padding: 5px; width: 45%;"> <p style="text-align: center;">content – table0.xml</p> <pre> <?xml version="1.0" encoding="utf-8" ?> <table xmlns:schemaLocation="http://www.admin.ch/xmlns/siard/1.0/schema0/table0.xsd table0.xsd" xmlns="http://www.admin.ch/xmlns/siard/1.0/schema0/table0.xsd" xmlns:xi="http://www.w3.org/2001/XMLSchema-instance"> <row><c1>1</c1><c2>Chai</c2><c3>1</c3><c4>1</c4> ... <c10>false</c10></row> <row><c1>2</c1><c2>Chang</c2><c3>1</c3><c4>1</c4> ... <c10>false</c10></row> <row><c1>3</c1><c2>Aniseed Syrup</c2><c3>1</c3><c4>2</c4> ... <c10>false</c10></row> ... <row><c1>75</c1><c2>Rhonbräu Klosterbier</c2><c3>12</c3><c4>1</c4> ... <c10>false</c10></row> <row><c1>76</c1><c2>Lakkalikööri</c2><c3>23</c3><c4>1</c4> ... <c10>false</c10></row> <row><c1>77</c1><c2>Frankfurter grüne Soße</c2><c3>12</c3><c4>2</c4> ... <c10>false</c10></row> </table> </pre> </div> </div> </div> <p>Empfehlung</p> <p>Es wird empfohlen, dass in table[Zahl].xsd der Bereich 0 bis unendlich (maxOccurs="unbounded" minOccurs="0") verwendet wird. So werden Probleme bei der Validierung von table[Zahl].xml gegenüber table[Zahl].xsd vermieden.</p>	<p>M</p>

5 Anforderungen an die Metadaten

Die Metadaten im SIARD-Archiv speichern die Struktur der archivierten Datenbank und geben an, welche Tabellendaten wo im Archiv zu finden sind.

Sämtliche Metadaten werden in einer einzigen Datei `metadata.xml` im Ordner `header/` versammelt. Die Datei ist im Gegensatz zu einer relationalen Datenbank hierarchisch aufgebaut.

Für die Datei `metadata.xml` existiert die Schemadefinition `metadata.xsd`, welche ebenfalls im Ordner `header/` abgelegt ist.

ID	Beschreibung Anforderung	M/K
M_5.0-1	Die Schemadefinition <code>metadata.xsd</code> ist für die Datei <code>metadata.xml</code> verbindlich einzuhalten. Das heisst, <code>metadata.xml</code> muss gegenüber <code>metadata.xsd</code> positiv validiert werden können.	M

Die Inhalte der einzelnen Ebenen werden im Folgenden definiert.

5.1 Metadaten auf der Ebene Datenbank

Die Datei `metadata.xml` enthält folgende globalen Angaben auf der Ebene Datenbank:

ID	Beschreibung Anforderung	M/K
M_5.1-1	Alle Metadaten, die in <code>metadata.xsd</code> auf der Ebene Datenbank als Muss bezeichnet sind, müssen entsprechend ausgefüllt sein.	M

Die folgenden Datenbank-Metadaten werden in der Datei `metadata.xml` gespeichert:

Bezeichnung	Bedeutung	M/K
version	SIARD-Format-Version	M
dbname	Kurze Bezeichnung der Datenbank	M
description	Beschreibung der Bedeutung und des Inhalts der Datenbank als Ganzes.	K
archiver	Name der Person, welche die Archivierung der Tabellendaten aus der Datenbank durchführte	K
archiverContact	Kontaktdaten (Telefon, E-Mail) zur Person, welche die Archivierung der Tabellendaten aus der Datenbank durchführte	K
dataOwner	Eigentümer der Daten in der Datenbank; die Institution oder Person, welche zum Zeitpunkt der Archivierung das Recht besitzt, Lizenzrechte an der Nutzung der Daten zu vergeben und die für die Einhaltung gesetzlicher Auflagen wie Datenschutzrichtlinien verantwortlich ist	M

Bezeichnung	Bedeutung	M/K
dataOriginTimespan	Entstehungszeitraum der Daten in der Datenbank; eine ungefähre Zeitangabe als Text	M
producerApplication	Name und Version der Anwendung, welche die SIARD-Datei heruntergeladen hat.	K
archivalDate	Archivierungsdatum; Datum der Archivierung der Tabellendaten	M
messageDigest	<p>hexadezimaler Message-Digest-Code über den Ordner <code>content/</code> mit Präfix, welches den Typ des Digest-Algorithmus angibt ("MD5" oder "SHA1")</p> <p>Empfehlung <i>Wird die Option MessageDigest verwendet, muss folgendes umgesetzt werden: Die Verzeichnisse <code>content</code> und <code>header</code> werden als separate (leere) Einträge <code>content/</code> und <code>header/</code> in der ZIP-Datei gespeichert. Damit die Integrität der Primärdaten überprüft werden kann, ist es notwendig, dass der Eintrag des <code>header</code>-Verzeichnisses erst nach allen Primärdaten und vor allen anderen Metadateneinträgen eingefügt wird. Der unten erwähnte MessageDigest wird von Offset 0 bis zum Offset des <code>header/</code>-Eintrags berechnet.</i></p>	K
clientMachine	DNS-Name des (Client-)Rechners, auf welchem die Archivierung durchgeführt wurde	K
databaseProduct	Datenbank-Produkt und Version, aus welchem die Archivierung der Tabellendaten erfolgte	K
connection	Verwendeter Connection String für die Archivierung der Tabellendaten	K
databaseUser	Datenbank-UserId des Benutzers des SIARD-Werkzeugs für das Archivieren der Tabellendaten aus der Datenbank	K
schemas	Liste der Schemas in der Datenbank	M
users	Liste der Datenbank-Benutzer	M
roles	Liste der Datenbank-Rollen	K
privileges	Liste der Privilegien für Benutzer und Rollen	K

5.2 Metadaten auf der Ebene Schema

Die Schema-Metadaten werden wie schon die globalen Angaben zur Datenbank in der Datei `metadata.xml` archiviert.

ID	Beschreibung Anforderung	M/K
M_5.2-1	Alle Metadaten, die in <code>metadata.xsd</code> auf der Ebene Schema als Muss bezeichnet sind, müssen entsprechend ausgefüllt sein.	M

Die folgenden Schema-Metadaten werden in der Datei `metadata.xml` gespeichert:

Bezeichnung	Bedeutung	M/K
name	Schemaname in der Datenbank	M
folder	Name des Schemaordners unter <code>content/</code> im SIARD-Archiv	M
description	Beschreibung der Bedeutung und des Inhalts des Schemas	K
tables	Liste der Tabellen in der Datenbank	M
views	Liste der in der Datenbank gespeicherten Queries (Abfragen)	K
routines	Liste der Routinen (früher Stored Procedures genannt) im Schema	K

5.3 Metadaten auf der Ebene Tabelle

Die Metadaten auf der Ebene Tabelle werden wie schon die globalen Angaben zur Datenbank und die Schema-Metadaten in der Datei `metadata.xml` archiviert.

ID	Beschreibung Anforderung	M/K
M_5.3-1	Alle Metadaten, die in <code>metadata.xsd</code> auf der Ebene Tabelle als Muss bezeichnet sind, müssen entsprechend ausgefüllt sein.	M

Die folgenden Tabellen-Metadaten werden in der Datei `metadata.xml` gespeichert:

Bezeichnung	Bedeutung	M/K
name	Tabellenname im Schema	M
folder	Name des Tabellenordners im Schemaordner	M
description	Beschreibung der Bedeutung und des Inhalts der Tabelle	K
columns	Liste der Spalten der Tabelle	M
primaryKey	Primärschlüssel der Tabelle	K
foreignKeys	Liste der Fremdschlüssel der Tabelle	K
candidateKeys	Liste der Kandidatenschlüssel der Tabelle	K
checkConstraints	Liste der Einschränkungen der Tabelle	K
triggers	Liste der Triggers der Tabelle	K
rows	Anzahl Datensätze	M

5.4 Metadaten auf der Ebene Spalte

Die Metadaten auf der Ebene Spalte werden wie schon die globalen Angaben zur Datenbank, die Schema-Metadaten und die Metadaten auf der Ebene Tabelle in der Datei `metadata.xml` archiviert.

ID	Beschreibung Anforderung	M/K
M_5.4-1	Alle Metadaten, die in <code>metadata.xsd</code> auf der Ebene Spalte als Muss bezeichnet sind, müssen ausgefüllt sein.	M

Die folgenden Spalten-Metadaten werden in der Datei `metadata.xml` gespeichert:

Bezeichnung	Bedeutung	M/K
name	Spaltenname in der Tabelle Innerhalb der gleichen Tabelle muss der Spaltenname eindeutig sein.	M
folder	Name des LOB- Ordners im Tabellenordner Hinweis Der optionale LOB-Ordnername wird nur für Spalten der Large-Object-Typen (z.B BLOB oder CLOB) benötigt. Die Dateien, welche die Large-Object-Felder repräsentieren, werden in diesen Ordnern angelegt und heissen <code>record0.txt</code> , <code>record1.txt</code> , bzw. <code>record0.bin</code> , <code>record1.bin</code> ... Diese werden in der Daten-XML-Datei referenziert.	K
type	SQL:1999-Typ der Spalte	M
typeOriginal	originaler Spaltentyp Hinweis Da die verschiedenen sich SQL-konform nennenden Datenbank-Programme sehr unterschiedliche Datentypen zulassen, wird hier neben dem SQL:1999-Typ auch der <i>originale</i> Typ aufgeführt. Für jedes das SIARD-Format unterstützende Datenbank-Programm ist eine Übersetzung der proprietären Typen zu SQL:1999-Typen bei der entsprechenden Applikation zu definieren und zu dokumentieren.	K
defaultValue	Standardwert der Spalte	K
nullable	Eintrag nicht erforderlich	K
description	Beschreibung der Bedeutung und des Inhalts der Spalte	K

5.5 Metadaten des Primärschlüssels

ID	Beschreibung Anforderung	M/K
M_5.5-1	Die Metadaten des Primärschlüssels einer Tabelle können in der Datei <code>metadeta.xml</code> archiviert werden	K

Die folgenden Primärschlüssel-Metadaten werden in der Datei `metadata.xml` gespeichert, sofern ein Primärschlüssel archiviert wird:

Bezeichnung	Bedeutung	M/K
name	Name des Primärschlüssels	M
column	Liste der Spalten des Primärschlüssels	M
description	Beschreibung der Bedeutung und des Inhalts des Primärschlüssels	K

5.6 Metadaten der Fremdschlüssel

ID	Beschreibung Anforderung	M/K
M_5.6-1	Die Metadaten der Fremdschlüssel innerhalb einer Tabelle können in der Datei <code>metadata.xml</code> archiviert werden	K

Die folgenden Fremdschlüssel-Metadaten werden in der Datei `metadata.xml` gespeichert, sofern ein Fremdschlüssel archiviert wird:

Bezeichnung	Bedeutung	M/K
name	Name des Fremdschlüssels	M
referencedSchema	Schema der referenzierten Tabelle	M
referencedTable	Referenzierte Tabelle Hinweis Der referenzierte externe Tabellename kann vom Typ <code>tabelle</code> oder <code>schema.tabelle</code> sein. Dabei sind delimitierte Bezeichner in Anführungszeichen gesetzt.	M
reference	Referenz (Liste von Spalten und referenzierten Spalten)	M
matchType	Matchtyp (FULL, PARTIAL oder SIMPLE)	K
deleteAction	Löschaktion, z.B.: CASCADE Hinweis Die Lösch- und Änderungsaktion enthalten die vom SQL:1999-Standard zugelassenen Aktionen.	K

Bezeichnung	Bedeutung	M/K
updateAction	Änderungsaktion, z.B.: SET DEFAULT	K
description	Beschreibung der Bedeutung und des Inhalts des Fremdschlüssels	K

5.7 Referenz-Metadaten

ID	Beschreibung Anforderung	M/K
M_5.7-1	Die Metadaten der Referenzen, welche beim Fremdschlüssel verwendet werden, können in der Datei <code>metadata.xml</code> archiviert werden	K

Die folgenden Referenzen-Metadaten werden in der Datei `metadata.xml` gespeichert, sofern ein Fremdschlüssel archiviert wird:

Bezeichnung	Bedeutung	M/K
column	Name der Spalte	M
referenced	Name der referenzierten Spalte	M

5.8 Metadaten des Kandidatenschlüssels

ID	Beschreibung Anforderung	M/K
M_5.8-1	Die Metadaten des Kandidatenschlüssels einer Tabelle können in der Datei <code>metadata.xml</code> archiviert werden	K

Die folgenden Kandidatenschlüssel-Metadaten werden in der Datei `metadata.xml` gespeichert, sofern ein Kandidatenschlüssel archiviert wird:

Bezeichnung	Bedeutung	M/K
name	Name des Kandidatenschlüssels	M
column	Liste der Spalten des Kandidatenschlüssels	M
description	Beschreibung der Bedeutung und des Inhalts des Kandidatenschlüssels	K

5.9 Metadaten der Check-Einschränkung

Die Check-Einschränkung besteht aus einer zu prüfenden Bedingung. Diese ist als Ausdruck vom Typ BOOLEAN (mit Wert *true*, *false* oder *unknown*) in SQL:1999-Syntax angegeben.

ID	Beschreibung Anforderung	M/K
M_5.9-1	Die Metadaten der Check-Einschränkung einer Tabelle können in der Datei <code>metadata.xml</code> archiviert werden	K

Die folgenden Check-Einschränkung-Metadaten werden in der Datei `metadata.xml` gespeichert, sofern eine Check-Einschränkung archiviert wird:

Bezeichnung	Bedeutung	M/K
name	Name der Check-Einschränkung	M
condition	Bedingung der Check-Einschränkung	M
description	Beschreibung der Bedeutung und des Inhalts der Check-Einschränkung	K

5.10 Metadaten auf der Ebene Trigger

ID	Beschreibung Anforderung	M/K
M_5.10-1	Die Metadaten des Triggers einer Tabelle können in der Datei <code>metadata.xml</code> archiviert werden	K

Die folgenden Trigger-Metadaten werden in der Datei `metadata.xml` gespeichert, sofern ein Trigger archiviert wird:

Bezeichnung	Bedeutung	M/K
name	Triggernamen in der Tabelle	M
actionTime	BEFORE oder AFTER	M
triggerEvent	INSERT, DELETE, UPDATE [OF <trigger column list>]	M
aliasList	<old or new value alias list>	K
triggeredAction	<triggered action>	M
description	Beschreibung der Bedeutung und des Inhalts des Triggers	K

5.11 Metadaten auf der Ebene View

ID	Beschreibung Anforderung	M/K
M_5.11-1	Die Metadaten der View eines Schemas können in der Datei <code>metadata.xml</code> archiviert werden	K

Die folgenden View-Metadaten werden in der Datei `metadata.xml` gespeichert, sofern eine View archiviert wird:

Bezeichnung	Bedeutung	M/K
name	Name der View im Schema	M

Bezeichnung	Bedeutung	M/K
columns	Liste der Spaltennamen der View Hinweis Die Metadaten der Spalten einer View sind identisch strukturiert wie diejenige einer Tabelle.	M
query	SQL:1999-Abfrage, welche die View definiert	K
queryOriginal	Originale SQL-Abfrage, welche die View definiert Hinweis Da die verschiedenen sich SQL-konform nennenden Datenbank-Programme sehr unterschiedliche Abfrage-Syntax zulassen, wird hier neben der SQL:1999-Abfrage auch die originale Abfrage aufgeführt. Für jedes das SIARD-Format unterstützende Datenbank-Programm ist eine Übersetzung der proprietären Abfragesyntax zu SQL:1999-Typen bei der entsprechenden Applikation zu definieren und zu dokumentieren.	K
description	Beschreibung der Bedeutung und des Inhalts der View	K

5.12 Metadaten auf der Ebene Routine

ID	Beschreibung Anforderung	M/K
M_5.12-1	Die Metadaten der Routine eines Schemas können in der Datei <code>metadata.xml</code> archiviert werden	K

Die folgenden Routine-Metadaten werden in der Datei `metadata.xml` gespeichert, sofern eine Routine archiviert wird:

Bezeichnung	Bedeutung	M/K
name	Routinename im Schema	M
description	Beschreibung der Bedeutung und des Inhalts der Routine	K
source	originaler Quellcode der Routine (VBA, PL/SQL, JAVA) Hinweis Da viele Datenbank-Programme über proprietäre Routinen verfügen, die nicht in eine SQL:1999-konforme Abfrage transformiert werden können, kann hier der originale Quellcode der Routine (z.B. in PL/SQL bei Oracle-Datenbanken, VBA bei MS Access Modulen) archiviert werden.	K
body	SQL:1999-konformer Quellcode der Routine	K
characteristic	Charakteristik der Routine	K
returnType	Rückgabebetyp der Routine (sofern es sich um eine Funktion handelt)	K
parameters	Liste der Parameter	K

5.13 Metadaten der Parameter

ID	Beschreibung Anforderung	M/K
M_5.13-1	Die Metadaten der Parameter, welche bei der Routine verwendet werden, können in der Datei <code>metadata.xml</code> archiviert werden	K

Die folgenden Parameter-Metadaten werden in der Datei `metadata.xml` gespeichert, sofern eine Routine archiviert wird:

Bezeichnung	Bedeutung	M/K
name	Name des Parameters	M
mode	Mode des Parameters (IN, OUT oder INOUT)	M
type	SQL:1999-Typ des Parameters	M
typeOriginal	originaler Parametertyp Hinweis Wie bei den Spaltenbeschreibungen kann hier der <i>originale</i> – proprietäre – Parametertyp angegeben werden.	K
description	Beschreibung der Bedeutung und der Funktion der Routine	K

5.14 Metadaten auf der Ebene des Benutzers

ID	Beschreibung Anforderung	M/K
M_5.14-1	Die Metadaten der Benutzer können in der Datei <code>metadata.xml</code> archiviert werden	K

Die folgenden User-Metadaten werden in der Datei `metadata.xml` gespeichert:

Bezeichnung	Bedeutung	M/K
name	Name des Benutzers	M
description	Beschreibung der Bedeutung und der Funktion des Benutzers	K

5.15 Metadaten auf der Ebene Rolle

ID	Beschreibung Anforderung	M/K
M_5.15-1	Die Metadaten der Rolle können in der Datei <code>metadata.xml</code> archiviert werden	K

Die folgenden Role-Metadaten werden in der Datei `metadata.xml` gespeichert:

Bezeichnung	Bedeutung	M/K
name	Name der Rolle	M
admin	Administrator der Rolle (Benutzer oder Rolle)	M
description	Beschreibung der Bedeutung und der Funktion der Rolle	K

5.16 Metadaten auf der Ebene der Privilegien

ID	Beschreibung Anforderung	M/K
M_5.16-1	Die Metadaten der Privilegien können in der Datei <code>metadata.xml</code> archiviert werden	K

Die folgenden Privilegien-Metadaten werden in der Datei `metadata.xml` gespeichert:

Bezeichnung	Bedeutung	M/K
type	ingeräumtes Privileg (z.B. SELECT)	M
object	Objekt, auf welches das Privileg anzuwenden ist	K
grantor	Berechtigter, der das Privileg einräumt	M
grantee	Empfänger des Privilegs (Benutzer oder Rolle)	M
option	Grant-Option (ADMIN oder GRANT)	K
description	Beschreibung der Bedeutung und der Funktion des Grants	K

6 Anforderungen an die Tabellendaten

Wie bereits beschrieben, befinden sich die Tabellendaten einer archivierten relationalen Datenbank im Ordner `content/` in der Dokument-Root des SIARD-Archivs. Sie werden dort in dem jeweiligen Schema- und Tabellenordner abgelegt.

Die Tabellendaten sind jeweils in einer XML-Datei gespeichert. Pro Tabelle wird eine XML-Schemadefinition erzeugt, welche das XML-Speicherformat der Tabellendaten angibt. Entsprechend existiert für jede Tabelle die Datei `table[Zahl].xml` zur Schemadefinition `table[Zahl].xsd`.

ID	Beschreibung Anforderung	M/K
D_6.0-1	Die Gesamtheit der Tabellendaten (Primärdaten) muss den Konsistenzanforderungen von SQL:1999 entsprechen. Eine SIARD-Datei, die zwar syntaktisch gegen die verschiedenen XSDs validiert, aber semantisch gegen den SQL-Standard verstösst, ist nicht konform zu der vorliegenden Formatbeschreibung. Insbesondere müssen die Tabellenwerte den Einschränkungen der SQL-Typen in den Metadaten entsprechen. Ausserdem müssen die in den Metadaten gespeicherten Primär-, Kandidaten- und Fremdschlüsselbedingungen und die Nullabilitätsbedingungen alle erfüllt sein.	M
T_6.0-2	Die Schemadefinition <code>table[Zahl].xsd</code> ist für die Datei <code>table[Zahl].xml</code> verbindlich einzuhalten. Das heisst, <code>table[Zahl].xml</code> muss gegenüber <code>table[Zahl].xsd</code> positiv validiert werden können.	M

6.1 Tabellen-Schemadefinition

Die Datei `table[Zahl].xsd` enthält folgende Schemadefinitionen zu einer Tabelle:

ID	Beschreibung Anforderung	M/K
T_6.1-1	Pro Tabelle muss eine XML-Schemadefinition existieren, welche das XML-Speicherformat der Tabellendaten angibt.	M

ID	Beschreibung Anforderung	M/K
T_6.1-2	<p>Diese Schemadefinition spiegelt die SQL-Schema-Metadaten der Tabelle wider und gibt an, dass die Tabelle als Sequenz von Zeilen gespeichert wird, welche eine Sequenz von Spalteneinträgen mit verschiedenen XML-Typen enthalten. Der Name des Tabellen-Tags ist <i>table</i>, derjenige des Datensatz-Tags ist <i>row</i>, die Spalten-Tags heissen <i>c1</i>, <i>c2</i>, ... (bzw. <i>siard:c1</i> ..., wenn der Namespace ausgeschrieben würde).</p> <p>Beispiel</p> <pre> content-table0.xsd <?xml version="1.0" encoding="UTF-8"?> <xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" targetNamespace=" "http://www.admin.ch/xmlns/siard/1.0/schema0/table0.xsd" xmlns=" "http://www.admin.ch/xmlns/siard/1.0/schema0/table0.xsd" xmlns:xs="http://www.w3.org/2001/XMLSchema"> <xs:element name="table"> <xs:complexType> <xs:sequence> <xs:element maxOccurs="unbounded" minOccurs="0" name="row" type="rowType"/> </xs:sequence> </xs:complexType> </xs:element> <xs:complexType name="rowType"> <xs:sequence> <xs:element name="c1" type="xs:integer"/> <xs:element minOccurs="0" name="c2" type="xs:string"/> <xs:element minOccurs="0" name="c3" type="xs:integer"/> <xs:element minOccurs="0" name="c4" type="xs:integer"/> <xs:element minOccurs="0" name="c5" type="xs:string"/> <xs:element minOccurs="0" name="c6" type="xs:decimal"/> <xs:element minOccurs="0" name="c7" type="xs:integer"/> <xs:element minOccurs="0" name="c8" type="xs:integer"/> <xs:element minOccurs="0" name="c9" type="xs:integer"/> <xs:element name="c10" type="xs:boolean"/> </xs:sequence> </xs:complexType> </xs:schema> </pre>	M

6.2 Tabellendaten

Die Datei `table[Zahl].xml` enthält die Tabellendaten zu dieser Tabelle:

ID	Beschreibung Anforderung	M/K
T_6.2-1	Pro Tabelle müssen die Tabellendaten jeweils in einer XML-Datei gespeichert sein.	M
T_6.2-2	<p>Die Datei <i>table</i> besteht aus <i>row</i>-Elementen, welche die Daten einer Zeile unterteilt in die verschiedenen Spalten (<i>c1</i>, <i>c2</i> ...) enthalten.</p> <p>Beispiel</p> <pre> content-table0.xml <?xml version="1.0" encoding="utf-8"?> <table xsi:schemaLocation="http://www.admin.ch/xmlns/siard/1.0/schema0/table0.xsd table0.xsd" xmlns="http://www.admin.ch/xmlns/siard/1.0/schema0/table0.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"> <row><c1>1</c1><c2>Chai</c2><c3>1</c3><c4>1</c4> ... <c10>>false</c10></row> <row><c1>2</c1><c2>Chang</c2><c3>1</c3><c4>1</c4> ... <c10>>false</c10></row> <row><c1>3</c1><c2>Aniseed Syrup</c2><c3>1</c3><c4>2</c4> ... <c10>>false</c10></row> ... <row><c1>75</c1><c2>Rhönbräu Klosterbier</c2><c3>12</c3><c4>1</c4> ... <c10>>false</c10></row> <row><c1>76</c1><c2>Lakkalikööri</c2><c3>23</c3><c4>1</c4> ... <c10>>false</c10></row> <row><c1>77</c1><c2>Frankfurter grüne Soße</c2><c3>12</c3><c4>2</c4> ... <c10>>false</c10></row> </table> </pre>	M

ID	Beschreibung Anforderung	M/K
T_6.2-3	<p>Enthält eine Zelle einer Spalte keinen Wert, kann diese weggelassen werden oder leer sein.</p> <p>Beispiel</p> <div data-bbox="432 443 1302 689" style="border: 1px solid #add8e6; padding: 5px;"> <p style="text-align: center;">content-table1.xml</p> <pre style="font-family: monospace; font-size: 0.9em;"> <?xml version="1.0" encoding="utf-8"?> <table xsi:schemaLocation="http://www.admin.ch/xmlns/siard/1.0/schema0/table1.xsd table1.xsd" xmlns="http://www.admin.ch/xmlns/siard/1.0/schema0/table1.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"> <row><c1>1</c1><c2>Speedy Express</c2></row> <row><c1>2</c1><c2>United Package</c2><c3></row> <row><c1>3</c1><c2>Federal Shipping</c2><c3></c3></row> </table> </pre> </div>	

7 Version und Gültigkeit der Spezifikation

Die Spezifikation liegt in der Version 1.0 vor. Die Inhalte der Spezifikation werden periodisch durch die eCH-Fachgruppe *Digitale Archivierung* überprüft und wenn nötig angepasst.

8 Change-Management-Prozess

Das Change-Management dieses Standards folgt [eCH-0150], Szenario 3. Als *Change Manager* dient die Fachgruppenleitung, als *Change Board* ein von der Fachgruppe mandatierter Ausschuss oder eine Themengruppe.

9 Haftungsausschluss/Hinweise auf Rechte Dritter

eCH-Standards, welche der Verein **eCH** dem Benutzer zur unentgeltlichen Nutzung zur Verfügung stellt, oder welche **eCH** referenziert, haben nur den Status von Empfehlungen. Der Verein **eCH** haftet in keinem Fall für Entscheidungen oder Massnahmen, welche der Benutzer auf Grund dieser Dokumente trifft und / oder ergreift. Der Benutzer ist verpflichtet, die Dokumente vor deren Nutzung selbst zu überprüfen und sich gegebenenfalls beraten zu lassen. **eCH**-Standards können und sollen die technische, organisatorische oder juristische Beratung im konkreten Einzelfall nicht ersetzen.

In **eCH**-Standards referenzierte Dokumente, Verfahren, Methoden, Produkte und Standards sind unter Umständen markenrechtlich, urheberrechtlich oder patentrechtlich geschützt. Es liegt in der ausschliesslichen Verantwortlichkeit des Benutzers, sich die allenfalls erforderlichen Rechte bei den jeweils berechtigten Personen und/oder Organisationen zu beschaffen.

Obwohl der Verein **eCH** all seine Sorgfalt darauf verwendet, die **eCH**-Standards sorgfältig auszuarbeiten, kann keine Zusicherung oder Garantie auf Aktualität, Vollständigkeit, Richtigkeit bzw. Fehlerfreiheit der zur Verfügung gestellten Informationen und Dokumente gegeben werden. Der Inhalt von **eCH**-Standards kann jederzeit und ohne Ankündigung geändert werden.

Jede Haftung für Schäden, welche dem Benutzer aus dem Gebrauch der **eCH**-Standards entstehen ist, soweit gesetzlich zulässig, wegbedungen.

10 Urheberrechte

Wer **eCH**-Standards erarbeitet, behält das geistige Eigentum an diesen. Allerdings verpflichtet sich der Erarbeitende, sein betreffendes geistiges Eigentum oder seine Rechte an geistigem Eigentum anderer, sofern möglich, den jeweiligen Fachgruppen und dem Verein **eCH** kostenlos zur uneingeschränkten Nutzung und Weiterentwicklung im Rahmen des Vereinszweckes zur Verfügung zu stellen.

Die von den Fachgruppen erarbeiteten Standards können unter Nennung der jeweiligen Urheber von **eCH** unentgeltlich und uneingeschränkt genutzt, weiterverbreitet und weiterentwickelt werden.

eCH-Standards sind vollständig dokumentiert und frei von lizenz- und/oder patentrechtlichen Einschränkungen. Die dazugehörige Dokumentation kann unentgeltlich bezogen werden.

Diese Bestimmungen gelten ausschliesslich für die von **eCH** erarbeiteten Standards, nicht jedoch für Standards oder Produkte Dritter, auf welche in den **eCH**-Standards Bezug genommen wird. Die Standards enthalten die entsprechenden Hinweise auf die Rechte Dritter.

Anhang A – Mitarbeit & Überprüfung

Hedi Bruggisser, Staatsarchiv Thurgau

Georg Büchler, KOST

Alain Dubois, Staatsarchiv Wallis

Martin Kaiser, KOST

Lambert Kansy, Staatsarchiv Basel Stadt

Markus Lischer, Staatsarchiv Luzern

Claire Röthlisberger-Jourdan, KOST

Hartwig Thomas, Enter AG

Andreas Voss, Schweizerisches Bundesarchiv

Anhang B – Abkürzungen und Glossar

Begriff	Beschreibung
AIP	Archival Information Package: AIP entstehen gemäss OAIS aus SIP im Laufe des Archivierungsprozesses der digitalen Unterlagen. AIP stellen diejenige Form der Informationspakete dar, in welcher die digitalen Unterlagen im digitalen Magazin gespeichert werden.
Aktenbildner	Bezeichnung der Stelle bzw. Organisationseinheit, welche die Unterlagen gebildet und geführt hat.
Archiv	<ol style="list-style-type: none"> 1. Institution/Stelle, die Archivgut erfasst, aufbewahrt, konserviert und zugänglich macht. 2. Archivierte Unterlagen einer Organisation. 3. Gebäude oder Institution, das/die für die Archivierung von Unterlagen gebaut oder hergerichtet wurde. 4. Begriff für eine Datei, die andere Dateien beinhaltet. Vgl. auch Archivdatei und als Synonym Containderdatei.
Archivgut	Als Archivgut gelten Unterlagen, die vom Archiv zur Aufbewahrung übernommen worden sind oder von anderen Stellen nach den gleichen Grundsätzen selbständig archiviert werden.
BAR	Schweizerisches Bundesarchiv
Datenbank	<p>Eine "Datenbank" besteht normalerweise aus einem oder mehreren Datenbank-Schemas sowie definierten Zugriffsrechten einzelner Benutzer und Rollen auf gewisse Teile der Datenbank. In SQL:1999 können Benutzer (Users) und Rollen (Roles) Träger von Berechtigungen (Privilegien) sein.</p> <p>Eine relationale Datenbank besteht somit aus einer Menge strukturierter Datenbankobjekte (z.B. Schema, View etc.) sowie den Tabelleninhalten.</p> <p>Ein Datenbankschema ist eine Art Namespace-Präfix. Ein Datenbankkatalog enthält die Metadaten aller Schemas im Katalog. Die Ebene Katalog in SQL:1999 entspricht der „Unterlage Datenbank“, die man mit SIARD in ein Archivformat umwandeln kann.</p>
Dauerhafte Archivierung / Langzeitarchivierung	Bezeichnung für die grundsätzlich unbegrenzte Aufbewahrung und die Erhaltung der dauerhaften Verfügbarkeit von digitalen Informationen. Neben der Erhaltung des Bitstroms der archivierten Information fällt darunter auch die Fähigkeit, denselben menschenlesbar und verständlich jederzeit interpretieren und darstellen zu können.
DIP	Dissemination Information Package: Ein DIP ist gemäss OAIS der Behälter für diejenigen Dossiers, welche von einem Benutzer in einem Bestellvorgang bestellt werden.
DNS	Domain Name System, eine verteilte Datenbank, die den Namensraum im Internet verwaltet.

Begriff	Beschreibung
Dossier	Als Dossier gilt die Gesamtheit (Kollektiv) der Unterlagen zu einem Geschäft. Grundsätzlich entspricht ein Dossier einem Geschäft. Durch Zusammenfassen artverwandter Geschäfte bzw. durch Aufteilung von Dossiers in Subdossiers kann diese Grundstruktur aber den jeweiligen Bedürfnissen angepasst werden. Die Dossierbildung erfolgt auf der Grundlage des Ordnungssystems.
Informationspaket	Ein konzeptioneller Container, der sich aus optionaler Inhaltsinformation und optional dazugehörigen Erhaltungsmetadaten zusammensetzt. Zu diesem Informationspaket gehört Verpackungsinformation, welche die Inhaltsinformation und die Paketbeschreibung voneinander abgrenzt und identifiziert sowie die Suche nach der Inhaltsinformation ermöglicht.
MD5	Message-Digest Algorithm 5
Metadaten	Metadaten können als «Informationen über die Primärdaten» (Daten über Daten) bezeichnet werden, da sie einen beschreibenden Charakter haben.
OAIS	Open Archival Information System, ISO 14721:2003. Das OAIS beschreibt als Referenzmodell ein Archiv als Organisation, in der Menschen und Systeme mit der Aufgabenstellung zusammenwirken, Informationen zu erhalten und einer definierten Nutzerschaft verfügbar zu machen.
Primärdaten	Primärdaten sind die Daten, welche die inhaltliche Substanz von Unterlagen ausmachen. Innerhalb einer SIARD-Datei nehmen die Tabellendaten die Funktion von Primärdaten ein.
Routinen	SQL-Routinen (auch unter der Bezeichnung Stored Procedures bekannt) sind vor allem zum Verständnis der View-Abfragen wichtig, bei welchen sie in Teilausdrücken vorkommen können.
Schemas	Schemas sind Behälter der Tabellen, Views und Routinen.
SHA1	sicherer Hash-Algorithmus (Secure Hash Algorithm)
SIP	Submission Information Package: SIP sind gemäss OAIS Informationspakete, die von den aktenbildenden Stellen an das Archiv übermittelt werden. Sie enthalten die digitalen Unterlagen (Primärdaten und Metadaten).
Tabellen	Tabellen bestehen aus einer Tabellendefinition mit Feldern, die jeder Spalte der Tabelle einen Namen und einen Typ zuordnen, aus Datensätzen, welche die eigentlichen Tabellendaten enthalten, aus einem optionalen Primärschlüssel, aus Fremdschlüsseln, welche die referenzielle Integrität sicherstellen, aus Kandidatenschlüsseln, welche zur Identifizierung eines Datensatzes dienen, und aus Einschränkungen, welche die Konsistenz garantieren. Optional können zu einer Tabelle so genannte Triggers (Auslöser) definiert sein.
Unterlagen	Unterlagen sind alle aufgezeichneten Informationen, unabhängig vom Informationsträger, welche bei der Erfüllung öffentlicher Aufgaben empfangen oder erstellt worden sind, sowie alle Hilfsmittel und ergänzenden Daten, die für das Verständnis dieser Informationen und deren Nutzung notwendig sind.

Begriff	Beschreibung
UTF	Unicode Transformation Format
Views	Views sind in der Datenbank gespeicherte Standardabfragen. Das Abfrageresultat ist eine Tabelle, welche ebenfalls Felder und Datensätze enthält.
XSD	XML Schema Definition

Anhang C – Nachweis der verwendeten Standards

eCH-0150	eCH-0150 Change und Release Management von eCH-Standards http://www.ech.ch/
SQL:1999	ISO/IEC 9075(1-4,9-11,13,14):2011: Information technology -- Database languages – SQL http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=53681
Unicode	Unicode 6.1.0 Unicode, Inc. http://www.unicode.org/versions/Unicode6.1.0/ (entspricht ISO/IEC 10646:2012: Information technology -- Universal Coded Character Set (UCS) http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=56921)
XML	Extensible Markup Language (XML), 1.1 (Second Edition) W3C Recommendation 16 August 2006, edited in place 29 September 2006 http://www.w3.org/TR/2006/REC-xml11-20060816/ (entspricht ISO/IEC 19503:2005: Information technology -- XML Metadata Interchange (XMI), http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=32622)
ZIP	.ZIP File Format Specification, Version 6.3.3 September 1, 2012 PKWARE Inc. http://www.pkware.com/documents/casestudies/APPNOTE.TXT

Anhang D – XML-Schemadefinitionen

D.1 metadata.xsd

Die XML-Schemadefinition `metadata.xsd` definiert die Struktur der Datei `metadata.xml` im Ordner `header/`. Sie ist separat als Beilage zur vorliegenden Spezifikation veröffentlicht.

D.2 Beispiel für metadata.xml

Eine zum XML-Schema für SIARD konforme Metadatenbeschreibung einer Datenbank sieht beispielsweise so aus:

```
<?xml version="1.0" encoding="UTF-8"?>
<siardArchive xmlns="http://www.bar.admin.ch/xmlns/siard/1.0/metadata.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="1.0"
xsi:schemaLocation="http://www.bar.admin.ch/xmlns/siard/1.0/metadata.xsd metadata.xsd">
  <dbname>jdbc:oracle:thin:@dbhost.enternet.ch:1521:SIARD1</dbname>
  <dataOwner>SIARD</dataOwner>
  <dataOriginTimespan>Fri May 16 11:21:39 CEST 2008</dataOriginTimespan>
  <archivalDate>2008-05-16</archivalDate>
  <messageDigest>MD5B9FB4FA23EFC27F10957533D747A4300</messageDigest>
  <clientMachine>blue2400.entereg.ch</clientMachine>
  <databaseProduct>
    Oracle Oracle9i Enterprise Edition Release 9.2.0.1.0 -
    Production\u000AWith the Partitioning, OLAP and Oracle Data Mining
    options\u000AJServer Release 9.2.0.1.0 - Production
  </databaseProduct>
  <connection>jdbc:oracle:thin:@dbhost.enternet.ch:1521:SIARD1</connection>
  <databaseUser>SIARD</databaseUser>
  <schemas>
    <schema>
      <name>SIARD</name>
      <folder>schema0</folder>
      <tables>
        <table>
          <name>TABLETEST</name>
          <folder>table0</folder>
          <description/>
          <columns>
            <column>
              <name>NID</name>
              <type>DECIMAL(38,0)</type>
              <typeOriginal>NUMBER</typeOriginal>
              <nullable>>false</nullable>
            </column>
            <column>
              <name>SNAME</name>
              <type>CHARACTER VARYING(31)</type>
              <typeOriginal>VARCHAR2</typeOriginal>
              <nullable>>true</nullable>
            </column>
            <column>
              <name>TSCREATED</name>
              <type>DATE</type>
              <typeOriginal>DATE</typeOriginal>
              <nullable>>false</nullable>
            </column>
          </columns>
          <primaryKey>
            <name>TABLETEST_PK</name>
            <column>NID</column>
            <column>TSCREATED</column>
          </primaryKey>
          <candidateKeys>
            <candidateKey>
              <name>"Unique2"</name>
              <column>SNAME</column>
              <column>TSCREATED</column>
            </candidateKey>
            <candidateKey>
              <name>UNIQUE1</name>
              <column>TSCREATED</column>
            </candidateKey>
          </candidateKeys>
          <rows>2</rows>
        </table>
      </tables>
    </schema>
  </schemas>
  <table>
    <name>"TableTest1"</name>
  </table>
</siardArchive>
```

```

<folder>table1</folder>
<description/>
<columns>
  <column>
    <name>"nID"</name>
    <type>DECIMAL(38,0)</type>
    <typeOriginal>NUMBER</typeOriginal>
    <nullable>>false</nullable>
  </column>
  <column>
    <name>"sName"</name>
    <type>CHARACTER VARYING(31)</type>
    <typeOriginal>VARCHAR2</typeOriginal>
    <nullable>>true</nullable>
  </column>
  <column>
    <name>"tsCreated"</name>
    <type>DATE</type>
    <typeOriginal>DATE</typeOriginal>
    <nullable>>false</nullable>
  </column>
</columns>
<primaryKey>
  <name>"TableTest1_PK"</name>
  <column>"nID"</column>
</primaryKey>
<foreignKeys>
  <foreignKey>
    <name>"TableTest1_FK"</name>
    <referencedSchema>SIARD</referencedSchema>
    <referencedTable>TABLETEST</referencedTable>
    <reference>
      <column>"nID"</column>
      <referenced>NID</referenced>
    </reference>
    <reference>
      <column>"tsCreated"</column>
      <referenced>TSCREATED</referenced>
    </reference>
    <deleteAction>RESTRICT</deleteAction>
    <updateAction>CASCADE</updateAction>
  </foreignKey>
</foreignKeys>
<candidateKeys>
  <candidateKey>
    <name>UNIQUE3</name>
    <column>"tsCreated"</column>
  </candidateKey>
</candidateKeys>
<rows>0</rows>
</table>
<table>
  <name>TABLETEST2</name>
  <folder>table2</folder>
  <description/>
  <columns>
    <column>
      <name>NID</name>
      <type>DECIMAL(38,0)</type>
      <typeOriginal>NUMBER</typeOriginal>
      <nullable>>false</nullable>
    </column>
    <column>
      <name>SNAME</name>
      <type>CHARACTER VARYING(31)</type>
      <typeOriginal>VARCHAR2</typeOriginal>
      <nullable>>true</nullable>
    </column>
    <column>
      <name>SCLOB</name>
      <folder>lob3</folder>
      <type>CHARACTER LARGE OBJECT(4000)</type>
      <typeOriginal>CLOB</typeOriginal>
      <nullable>>false</nullable>
    </column>
  </columns>

```

```

        <name>TSCREATED</name>
        <type>DATE</type>
        <typeOriginal>DATE</typeOriginal>
        <nullable>>false</nullable>
    </column>
    <column>
        <name>COLBINARYLARGEOBJECT</name>
        <folder>lob5</folder>
        <type>BINARY LARGE OBJECT (4000)</type>
        <typeOriginal>BLOB</typeOriginal>
        <nullable>>true</nullable>
    </column>
    <column>
        <name>COLCHARACTER</name>
        <type>CHARACTER (2)</type>
        <typeOriginal>CHAR</typeOriginal>
        <nullable>>true</nullable>
    </column>
    <column>
        <name>COLDATE</name>
        <type>DATE</type>
        <typeOriginal>DATE</typeOriginal>
        <nullable>>true</nullable>
    </column>
    <column>
        <name>COLDECIMAL</name>
        <type>DECIMAL (2,1)</type>
        <typeOriginal>NUMBER</typeOriginal>
        <nullable>>true</nullable>
    </column>
    <column>
        <name>COLDOUBLEPRECISION</name>
        <type>DECIMAL (22,0)</type>
        <typeOriginal>NUMBER</typeOriginal>
        <nullable>>true</nullable>
    </column>
    <column>
        <name>COLFLOAT</name>
        <type>DECIMAL (22,0)</type>
        <typeOriginal>NUMBER</typeOriginal>
        <nullable>>true</nullable>
    </column>
    <column>
        <name>COLINTEGER</name>
        <type>DECIMAL (38,0)</type>
        <typeOriginal>NUMBER</typeOriginal>
        <nullable>>true</nullable>
    </column>
    <column>
        <name>COLNUMERIC</name>
        <type>DECIMAL (2,1)</type>
        <typeOriginal>NUMBER</typeOriginal>
        <nullable>>true</nullable>
    </column>
    <column>
        <name>COLREAL</name>
        <type>DECIMAL (22,0)</type>
        <typeOriginal>NUMBER</typeOriginal>
        <nullable>true</nullable>
    </column>
    <column>
        <name>COLSMALLINT</name>
        <type>DECIMAL (38,0)</type>
        <typeOriginal>NUMBER</typeOriginal>
        <nullable>true</nullable>
    </column>
</columns>
<primaryKey>
    <name>TABLETEST2_PK</name>
    <column>NID</column>
</primaryKey>
<rows>4</rows>
</table>
</tables>
<views>
    <view>

```

```

<name>"View1"</name>
<columns>
  <column>
    <name>"nID"</name>
    <type>DECIMAL(38,0)</type>
    <typeOriginal>NUMBER</typeOriginal>
    <nullable>>false</nullable>
  </column>
  <column>
    <name>"sName"</name>
    <type>CHARACTER VARYING(31)</type>
    <typeOriginal>VARCHAR2</typeOriginal>
    <nullable>>true</nullable>
  </column>
  <column>
    <name>"tsCreated"</name>
    <type>DATE</type>
    <typeOriginal>DATE</typeOriginal>
    <nullable>>false</nullable>
  </column>
</columns>
</view>
<view>
  <name>VIEW2</name>
  <columns>
    <column>
      <name>"nID"</name>
      <type>DECIMAL(38,0)</type>
      <typeOriginal>NUMBER</typeOriginal>
      <nullable>>false</nullable>
    </column>
    <column>
      <name>"sName"</name>
      <type>CHARACTER VARYING(31)</type>
      <typeOriginal>VARCHAR2</typeOriginal>
      <nullable>>true</nullable>
    </column>
    <column>
      <name>"tsCreated"</name>
      <type>DATE</type>
      <typeOriginal>DATE</typeOriginal>
      <nullable>>false</nullable>
    </column>
  </columns>
</view>
</views>
<routines>
  <routine>
    <name>IS_LE</name>
    <description>
      Standalone procedure or function
    </description>
    <returnType>NUMERIC</returnType>
    <parameters>
      <parameter>
        <name>S1</name>
        <mode>IN</mode>
        <type>CHARACTER VARYING</type>
        <typeOriginal>VARCHAR2</typeOriginal>
      </parameter>
      <parameter>
        <name>S2</name>
        <mode>IN</mode>
        <type>CHARACTER VARYING</type>
        <typeOriginal>VARCHAR2</typeOriginal>
      </parameter>
    </parameters>
  </routine>
  <routine>
    <name>"IsZero"</name>
    <description>
      Standalone procedure or function
    </description>
    <returnType>NUMERIC</returnType>
    <parameters>
      <parameter>

```



```

        <name>I</name>
        <mode>IN</mode>
        <type>DECIMAL(38,0)</type>
        <typeOriginal>NUMBER</typeOriginal>
    </parameter>
</parameters>
</routine>
</routines>
</schema>
</schemas>
<users>
  <user>
    <name>SIARD3</name>
  </user>
  <user>
    <name>SIARD1</name>
  </user>
  <user>
    <name>SIARD2</name>
  </user>
  <user>
    <name>SIARD</name>
  </user>
</users>
<roles>
  <role>
    <name>"siardrole2"</name>
    <admin/>
  </role>
  <role>
    <name>SIARDROLE3</name>
    <admin/>
  </role>
  <role>
    <name>SIARDROLE1</name>
    <admin/>
  </role>
  <role>
    <name>RESOURCE</name>
    <admin/>
  </role>
  <role>
    <name>CONNECT</name>
    <admin/>
  </role>
</roles>
<privileges>
  <privilege>
    <type>SELECT</type>
    <object>TABLE TABLETEST2</object>
    <grantor>SIARD</grantor>
    <grantee>SIARD2</grantee>
  </privilege>
  <privilege>
    <type>SELECT</type>
    <object>TABLE "TableTest1"</object>
    <grantor>SIARD</grantor>
    <grantee>SIARD1</grantee>
  </privilege>
  <privilege>
    <type>SELECT</type>
    <object>TABLE "TableTest1"</object>
    <grantor>SIARD</grantor>
    <grantee>SIARD3</grantee>
  </privilege>
  <privilege>
    <type>SELECT</type>
    <object>TABLE TABLETEST</object>
    <grantor>SIARD</grantor>
    <grantee>SIARD2</grantee>
  </privilege>
  <privilege>
    <type>UPDATE</type>
    <object>TABLE TABLETEST2</object>
    <grantor>SIARD</grantor>
    <grantee>SIARD2</grantee>

```

```

</privilege>
<privilege>
  <type>UPDATE</type>
  <object>TABLE TABLETEST2</object>
  <grantor>SIARD</grantor>
  <grantee>SIARD3</grantee>
</privilege>
<privilege>
  <type>SELECT</type>
  <object>TABLE "TableTest1"</object>
  <grantor>SIARD</grantor>
  <grantee>"siardrole2"</grantee>
</privilege>
</privileges>
</siardArchive>

```

D.3 Beispiel für die XML-Schemadefinition einer Tabelle: table0.xsd

Für jede Tabelle wird von SIARD eine XML-Schemadefinition erzeugt, welche den Spalten die richtigen XML-Datentypen zuordnet.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="http://www.bar.admin.ch/xmlns/siard/1.0/schema0/table0.xsd"
  attributeFormDefault="unqualified" elementFormDefault="qualified"
  targetNamespace="http://www.bar.admin.ch/xmlns/siard/1.0/schema0/table0.xsd">
  <xs:element name="table">
    <xs:complexType>
      <xs:sequence>
        <xs:element maxOccurs="unbounded" minOccurs="0" name="row" type="rowType">
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="rowType">
    <xs:sequence>
      <xs:element name="c1" type="xs:decimal"/>
      <xs:element minOccurs="0" name="c2" type="xs:string"/>
      <xs:element name="c3" type="xs:date"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>

```

D.4 Beispiel für die Tabellendaten einer Tabelle: table0.xml

Die Tabellendaten werden in einer XML-Datei gespeichert, die der XML-Schemadefinition der Tabelle genügt.

```

<?xml version="1.0" encoding="utf-8"?>
<table
  xsi:schemaLocation="http://www.bar.admin.ch/xmlns/siard/1.0/schema0/table0.xsd table0.xsd"
  xmlns="http://www.bar.admin.ch/xmlns/siard/1.0/schema0/table0.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <row><c1>1</c1><c2>First Name</c2><c3>2008-05-09</c3></row>
  <row><c1>2</c1><c2>Second Name</c2><c3>2008-05-10</c3></row>
</table>

```