Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Federal Department of Home Affairs FDHA
Swiss Federal Archives SFA
Unit Innovation and Preservation

Hartwig Thomas, 4. December 2013

Document version 1.01

SIARD Suite
Data Type Mapping for MySQL

Published by:

Swiss Federal Archives
Archivstrasse 24
3003 Bern
Switzerland

1 Introduction

The Swiss Federal Archives developed the database archival called SIARD (Software Independent Archiving of Relational Databases) within the framework of the ARELDA (ARchivierung ELektronischer DAten) project. The SIARD format is used for long-term archiving of relational database content.

On behalf of the Swiss Federal Archives, Enter AG implemented the software SIARD Suite which supports converting database content from live proprietary database systems to the normalized SIARD format as well as uploading database content in SIARD format to such a live database system.

Unfortunately most real database systems to not support the SQL:1999 standard fully, on which the SIARD format is based. Therefore SIARD Suite needs to normalize/denormalize the data types during the conversion process.

This document specifies, how the MySQL data are converted to the SIARD format and how SIARD data are converted to MySQL on upload.

The conversions are *idempotent*. I.e. after the initial download any number of up- and download can be executed without changing the data types or values.

2 Mapping of MySQL Data Types

2.1 MySQL => SIARD

See also http://dev.mysql.com/doc/refman/5.0/en/data-types.html and http://dev.mysql.com/doc/refman/5.5/en/data-types.html

MySQL	*JDBC* *(java.sql.Types)*	*SQL:1999 (SIARD)*	*XML*
CHAR	CHAR(1)	CHARACTER(1)	xs:string
CHAR(n)	CHAR(n)	CHARACTER(n)	xs:string
VARCHAR(n)	VARCHAR(n)	CHARACTER VARYING(n)	xs:string
TINYTEXT (up to 255)	VARCHAR(255)	CHARACTER VARYING(255)	xs:string
TEXT (up to 65‘535)	LONGVARCHAR (65‘535)	CHARACTER LARGE OBJECT	clobType
MEDIUMTEXT (up to 16‘777‘215)	LONGVARCHAR (16‘777‘215)	CHARACTER LARGE OBJECT	clobType
LONGTEXT (up to 2‘147‘483‘647)	LONGVARCHAR (2‘147‘483‘647)	CHARACTER LARGE OBJECT	clobType
TINYINT (1 Byte)	TINYINT(3)	SMALLINT	xs:integer
SMALLINT (2 Bytes)	SMALLINT(5)	SMALLINT	xs:integer
MEDIUMINT (3 Bytes)	INTEGER(7)	INTEGER	xs:integer
INT (4 Bytes)	INTEGER(10)	INTEGER	xs:integer
BIGINT (8 Bytes)	BIGINT(19)	NUMERIC(19)	xs:decimal
DECIMAL	DECIMAL(10)	DECIMAL(10)	xs:decimal
DECIMAL(n)	DECIMAL(n)	DECIMAL(n)	xs:decimal

MySQL	*JDBC* *(java.sql.Types)*	*SQL:1999 (SIARD)*	*XML*
DECIMAL(p,q)	DECIMAL(p,q)	DECIMAL(p,q)	xs:decimal
NUMERIC	DECIMAL(10)	DECIMAL(10)	xs:decimal
NUMERIC(n)	DECIMAL(n)	DECIMAL(n)	xs:decimal
NUMERIC(p,q)	DECIMAL(p,q)	DECIMAL(p,q)	xs:decimal
FLOAT	REAL(12)	REAL	xs:float
FLOAT(p)	REAL(12)	REAL	xs:float
FLOAT(p,q)	REAL(p,q)	FLOAT(p)	xs:float
DOUBLE	DOUBLE(22)	DOUBLE PRECISION	xs:float
DOUBLE(p,q)	DOUBLE(p,q)	DOUBLE PRECISION	xs:float
BIT	BIT(1)	BOOLEAN	xs:boolean
BIT(n)	BIT(n)	BIT(n)	xs:hexBinary
BINARY(n)	BINARY(n)	BIT(8*n)	xs:hexBinary
VARBINARY(n)	VARBINARY(n)	BIT VARYING(8*n)	xs:hexBinary
TYNIBLOB (up to 255)	BINARY(255)	BIT VARYING(2040)	xs:hexBinary
BLOB (up to 65‘535)	LONGVARBINARY(65‘535)	BINARY LARGE OBJECT	blobType
MEDIUMBLOB (up to 16‘777‘215)	LONGVARBINARY(16‘777‘215)	BINARY LARGE OBJECT	blobType
LONGBLOB (up to 2‘147‘483‘647)	LONGVARBINARY(2‘147‘483‘647)	BINARY LARGE OBJECT	blobType
DATETIME	TIMESTAMP(19)	TIMESTAMP	xs:dateTime
TIMESTAMP	TIMESTAMP(19)	TIMESTAMP(19)	xs:dateTime
DATE	DATE(10)	DATE	xs:date
TIME	TIME(8)	TIME	xs:time

MySQL	*JDBC (java.sql.Types)*	*SQL:1999 (SIARD)*	*XML*
YEAR	DATE(0)	DATE	xs:date

2.2 SIARD => MySQL

XML	*SQL:1999 (SIARD)*	*MySQL*
xs:decimal	NUMERIC	NUMERIC(10)
xs:decimal	NUMERIC(n)	NUMERIC(n)
xs:decimal	NUMERIC(p,q)	NUMERIC(p,q)
xs:decimal	DECIMAL	DECIMAL(10)
xs:decimal	DECIMAL(n)	DECIMAL(n)
xs:decimal	DECIMAL(p,q)	DECIMAL (p,q)

xs:hexBinary	BIT(n)	if (n < 8*68) BIT(n)
		else if (n <= 8*255) BINARY((n+7)/8)
		else if (n <= 8*65535) BLOB
		else if (n <= 8*16777215) MEDIUMBLOB
		else LONGBLOB

xs:hexBinary	BINARY LARGE OBJECT	if (n < 256) TINYBLOB (n)
		else if (n < 65536) BLOB
		else if (n < 16777216) MEDIUMBLOB
		else LONGBLOB

xs:boolean	BOOLEAN	BIT(1)
xs:string	CHARACTER	CHAR

xs:string	CHARACTER(n)	if (n < 256) CHAR(n)
		else if (n < 65536) TEXT
		else if (n < 16777216) MEDIUMTEXT
		else LONGTEXT

xs:string	NATIONAL CHARACTER	CHAR

XML	*SQL:1999 (SIARD)*	*MySQL*

xs:string	NATIONAL CHARACTER(n)	if (n < 256) CHAR(n)
		else if (n < 65536) TEXT
		else if (n < 16777216) MEDIUMTEXT
		else LONGTEXT

xs:string	CHARACTER LARGE OBJECT	if (n < 256) TINYTEXT (n)
		else if (n < 65536) TEXT
		else if (n < 16777216) MEDIUMTEXT
		else LONGTEXT

xs:string	NATIONAL CHARACTER LARGE OBJECT	if (n < 256) TINYTEXT (n)
		else if (n < 65536) TEXT
		else if (n < 16777216) MEDIUMTEXT
		else LONGTEXT

xs:date	DATE	DATE
xs:time	TIME(p)	TIME
xs:dateTime	TIMESTAMP(p)	DATETIME
xs:decimal	DECIMAL(n)	DECIMAL(n)
xs:decimal	DECIMAL(p,q)	DECIMAL (p,q)
xs:float	REAL	FLOAT(12)
xs:float	DOUBLE PRECISION	DOUBLE
xs:float	FLOAT(p,q)	FLOAT(p,q)
xs:integer	INTEGER	INTEGER
xs:integer	SMALLINT	SMALLINT

xs:hexBinary	BIT VARYING(n)	if (n < 65536) VARBINARY(n)
		else if (n < 16777216) MEDIUMBLOB
		else LONGBLOB

xs:string	CHARACTER VARYING(n)	if (n < 65536) VARCHAR (n)
		else if (n < 16777216) MEDIUMTEXT
		else LONGTEXT

XML	*SQL:1999 (SIARD)*	*MySQL*
xs:string	NATIONAL CHARACTER VARYING(n)	VARCHAR(n)

If a string is longer than 4000 characters then „clobType“ and „xs:string“ are replaced by an external reference to a text file.

If a binary array is longer than 2000 bytes then „blobType“ and „xs:hexBinary“ are replaced by an external reference to a binary file.

Characters that cannot be represented in UNICODE (Codes 0-8, 14-31, 127-159) as well as the escape character '\' and multiple space characters are escaped as \u00<xx> in XML. Less-than and ampersand characters are represented as entity references in XML.