SIARD Format 2.0 – Final Version

Feedback

The „final“ version of the SIARD Format 2.0 was agreed upon by the E-ARK project on July, 4th, 2015. A number of interested parties was then asked to provide feedback for the planned standard. The feedback would be addressed to Anders Bo Nielsen on behalf of E-ARK and to Hartwig Thomas on behalf of the Swiss Federal Archives, who acted in a consulting role for the E-ARK project.

This document collects feedback, contains comments, and suggests resulting changes. Each feedback item was usually mailed. Most of them were answered (commented on) by Hartwig Thomas. Some of them were subsequently clarified. Some of them led to suggestions of changes to the SIARD Format 2.0 standard.

In this document each feedback item has the following parts:

- Title
- Date of first mail
- Sender of first mail
- Content (paraphrased including all subsequent clarification)
- Comment
- Suggested Changes to SIARD Format 2.0 suggested by Hartwig Thomas

Time in SIARD Specification

Date: July, 10th, 2015

Sender: Luis Faria

Content: The regular expression in the metadata XSD does not permit decimal places.

Comment: The example contains decimal places and strangely the validator used, did not complain …

Changes: Correct the regular expressions to contain decimal places.

Message digest ...

Date: July, 23rd, 2015

Sender: Luis Faria

Content: The message digest should not be mandatory.

Comment: The message digest is not mandatory any more in SIARD Format 2.0

Changes: none

… and database info

Date: July 23rd, 2015

Sender: Luis Faria

Content: It is suggested, that in addition the the full textual database product description at the download time, the numeric version and release number of the database product should be stored in the SIARD archive.

Then, if the version and release number were identical, one could use the „original data type“ on upload.

Comment: This information is already contained in the database product description. Saving it in addition in a more „structured“ way, does not appear to serve any purpose. It is also not recommended, to use the „original data type“ on upload. This will not even work for the same version and release. (E.g. 'varchar(max)' is detected as 'varchar(2147483647)' on SQL Server, which cannot be used for upload …)

Changes: none

Lack of expressiveness on some SIARD fields

Date: Aug. 4th, 2015

Sender: Luis Faria

Content: It is considered problematic that some very proprietary settings of some DBMSs are not preserved in the SIARD archive. An example is given, where MySQL supports „INSTEAD OF“ for TRIGGERs in addition to „BEFORE“ and „AFTER“, whereas SQL:1999 – and therefore the SIARD Format 1.0 – only supports „BEFORE“ and „AFTER“.

Comment: The goal is to preserve primary data, not code for TRIGGERs. For some major SQL objects (VIEWs, ROUTINEs) fields that can contain the original code text are part of the standard. TRIGGERs and CHECK CONTRAINTs, however, are not deemed to be sufficiently important to preserve, because they only describe what happens, when the data change.

However, because SIARD Format 2.0 has moved the support from SQL:1999 to SQL:2008 it was remarked, that the SQL standard was amended and now supports INSTEAD OF TRIGGERs. Therefore it makes sense, to add that option, which is not to be considered proprietary any more, when it is supported by the standard.

As a generic guideline it is recommended, to map the proprietary DBMS to a standard database conforming to the SQL:2008 standard as well as possible without losing any primary data, and not insist on being able to reconstruct the proprietary constructs of every database system.

Changes: Add INSTEAD OF to TRIGGER object definition.

SIARD and time zones

Date: Aug. 4th, 2015

Sender: Luis Faria

Content: SIARD Format 2.0 should not only store UTC time stamps but also preserve time zones. Otherwise this information is lost.

Comment: This is a very difficult problem because the SQL standard has defined the time zone support in a contradictory way. Most DBMSs appear to store the time stamps in UTC and convert it from and to the client session's time zone, when it is entered and displayed.

If different users can enter different time zone offsets on entering time data and this offset is stored somewhere, and if it were possible to extract the stroed time zone offset, which may be different for each row of such a column, then it would make sense to extend the standard of the SIARD Format 2.0 to include xs:dateTime and xs:time fields, that are not given in UTC. However, as long as no DBMS permits extraction of the original time zone offset entered, it cannot be stored either and cannot be considered as „lost“ information.

Changes: none, unless an example for the extraction of stored TZ offsets is presented.

SIARD archives outside a ZIP64 file

Date Sept. 4th, 2015

Sender: Andrew Wilson on behalf of Riksarkivet Sweden

Content: SIARD Format 2.0 should possibly not be contained in a ZIP file but in individual „table files“, that can be referenced from METS and/or PREMIS.
Comment: Experience with the prototype of SIARD has shown us, that the integrity of a database is much more likely to be preserved, when the whole database is contained in a single container. The problem of referencing files within a ZIP also appears in DOCX files, which are ZIP files containing any number of image files and other material besides the document's text. METS and PREMIS are able to handle referencing the whole document (DOCX) als well as some sub documents (e.g. images) contained in it and nobody is likely to suggest that DOCX files should be stored outside a ZIP file for long-term preservation. The same applies to SIARD files.

Changes: none

SIARD archives in TAR files

Date Sept. 4th, 2015

Sender: Andrew Wilson on behalf of Riksarkivet Sweden

Content: It is suggested to use TAR or TGZ (TAR and gzip) format for SIARD archives.

Comment: This format was considered, when the container format was chosen for SIARD 1.0. We found, that it was not as clearly standardized as ZIP and was less likely to function as a general exchange format. A major factor was the lack of a „directory“ in the TAR format which permits opening and editing metadata very fast even in huge files. Another reason for chosing the ZIP64 format was the universal adoption of the ZIP format in other standards (OOXML for DOCX, XLSX, PPTX; ODF for ODT, ODS, ODP, …) which makes the long-term availability of tools for reading it very probable. Allowing many variants of file formats would weaken the usefulness of the standard without adding to its power.

Changes: none

SIARD archives to be split

Date Sept. 4th, 2015

Sender: Andrew Wilson on behalf of Riksarkivet Sweden

Content: It is suggested that it should be possible to split SIARD archives into two or more SIARD files, because SIARD files can become very large even if LOBs are stored separately. This leads to large SIPs and AIPs.

Comment: The SIARD standard should not be too closely concerned with particular hardware- or platform-related limitations. In practive it is always possible to split large files for practical reasons (e.g. using the UNIX split command) without impact for the file format.

In practice we have found that even very large databases have not become impractical to handle if the LOBs were stored separately. We have not yet encountered a database anywhere that was larger than 20 GB.

What is true about the SIARD format is also true for SIPs and AIPs. Their format should not be dependent on hardware- or platform-related limitations. It is always possible to devise a way of splitting large SIPs or AIPs for practical reasons without impact for its structure definition.

Changes: none

Uniquely identifying each export of a database

Date Sept. 4th, 2015

Sender: Andrew Wilson on behalf of Riksarkivet Sweden

Content: It is suggested one should have a unique identifier for each database export.

Comment: Such an identifier is part of the proposed SIARD Format 2.0 standard. The message digest uniquely identifies the primary data of the database better than a UUID/GUID. It is more useful than the UUID/GUID because it is garanteed to be the same when the primary data are the same.

Changes: none

User-defined metadata extensions

Date Sept. 4th, 2015

Sender: Andrew Wilson on behalf of Riksarkivet Sweden

Content: It is suggested that a „generic“ XML element should be included in the metadata definition which would permit user-defined metadata extensions.

Comment: We have found that user-defined extensions (e.g. in the TIF format) very much impair the exchangeability of standard files. Nobody is prevented from storing structured XML in a „description“ field. But that is not part of the standard and thus outside its scope.

Changes: none

Separating type and value in messageDigest

Date Sept. 4th, 2015

Sender: Andrew Wilson on behalf of Riksarkivet Sweden

Content: It is suggested that type and value of a message digest should be put into two separate fields.

Comment: If the SIARD standard were completely new then we might consider such a change. Today, however, every program that reads SIARD files must do the little amount of parsing separating the type of the digest from its value for SIARD Format 1.0. The amount of processing needed is very small. Therefore we propose to leave the digest definition as it is.

Changes: none

file: URIs referencing folders in exotic file systems

Date Sept. 4th, 2015

Sender: Andrew Wilson on behalf of Riksarkivet Sweden

Content: It is suggested/asked, that file: URIs in the *lobFolder* elements should be able to reference file in non-standard object store systems or in cloud storage solutions.

Comment: The file: URI is defined independent of the concrete file system used. It can reference files on any platform and thus also in object store systems or in cloud storage solutions.

Changes: none

Message Digests for LOB files should also accommodate SHA-256.

Date Sept. 4th, 2015

Sender: Andrew Wilson on behalf of Riksarkivet Sweden

Content: It is suggested that the message digests for lob files should be split into two separate fields (like the message digest for the whole database; see above). Also beside MD5 and SHA-1 the possibility SHA-256 should be supported.

Comment: As the attribute messageDigest for LOBs was not present in SIARD Format 1.0 it makes sense to split it into two fields: digestType and messageDigest. In the proposed standard SHA-256 was already included as a possibility for message digests of the whole database. It should also be included for the message digests of LOB files.

From the practical point of view of a real archive it is unimportant, whether one digest is claimed to be more „secure“ than another. If one is already using MD5 there is no need to change to a more „secure“ algorithm.

Changes: Add attribute „digestType“ with possible values „MD5“, „SHA-1“ and „SHA-256“ in 6.2 and in the XSDs and the examples.

Remove doubtful text

Date: Sept. 7th, 2015

Sender: Bruno Ferreira (later annotated by H. Silva)

Document: https://docs.google.com/document/d/1-nsL2JNm01_dE_-rVVNqTGeuVNhcN7NyHiF-hdwS2y4/edit#heading=h.w1q77sepg2fa

Content: In G_3.3-1, G_3.4-1 and G_3.5-1 the clause „in general“ may lead to doubt, whether the requirement needs to be fulfilled.

Comment: The clause „in general“ should be removed in these three places.

Changes: remove „in general“ from G3_3-1, G_3.4-1 and G_3.5-1.

Is „should“ mandatory or optional?

Date: Sept. 7th, 2015

Sender: Bruno Ferreira (later annotated by H. Silva)

Document: https://docs.google.com/document/d/1-nsL2JNm01_dE_-rVVNqTGeuVNhcN7NyHiF-hdwS2y4/edit#heading=h.w1q77sepg2fa

Content: In P_4.2-2 and P_4.2-5 mandatory requirements have a recommendation. It is not clear, whether „should“ means „must“.

Comment: (Actually the numbering has changed to P_4.2-3 and P_4.2-6.)

P_4.3-3 does not contain „should“. P_4.2-6 does say that the maximum length of file names should not exceed 200 characters, where possible. The „where possible“ and the title „recommendation“ make it sufficiently clear that these requirements are not mandatory. The reason given – problems unzipping the file in a Windows environment – makes it sufficiently clear that it is useful today to stick to this recommendation but that it may become unimportant, when Windows permits path names with more than 240 characters.

Changes: none

„attributer“ instead of „attribute“?

Date: Sept. 7th, 2015

Sender: Bruno Ferreira (later annotated by H. Silva)

Document: https://docs.google.com/document/d/1-nsL2JNm01_dE_-rVVNqTGeuVNhcN7NyHiF-hdwS2y4/edit#heading=h.w1q77sepg2fa

Content: In section 5.4 the identifier „name“ is spelled „attributer“ instead of „attribute“.

Comment: Typing mistake.

Changes: correct typing mistake.

What is the meaning of „externalBase“?

Date: Sept. 7th, 2015

Sender: Bruno Ferreira (later annotated by H. Silva)

Document: https://docs.google.com/document/d/1-nsL2JNm01_dE_-

rVVNqTGeuVNhcN7NyHiF-hdwS2y4/edit#heading=h.w1q77sepg2fa

Content: In column-level metadata (section 5.6) the „externalBase on the database level“ is referenced.What does it mean?

Comment: In an earlier version what is now called „lobFolder“ on the database level was called „externalBase“. This change has not been carried through correctly everywhere.

Changes: replace „externalBase“ by „lobFolder in 5.6.

Clarification for relative *lobFolder* elements needed

Date: Sept. 7th, 2015

Sender: Bruno Ferreira (later annotated by H. Silva)

Document: https://docs.google.com/document/d/1-nsL2JNm01_dE_-rVVNqTGeuVNhcN7NyHiF-hdwS2y4/edit#heading=h.w1q77sepg2fa

Content: In column-level metadata (section 5.6) it is specified, that relative URIs refers to the value of the lobFolder element of the enclosing ROW or UDT. Columns only show up in table definitions.

Comment: It is true that columns only show up in table – and view – definitions. Therefore the reference here to ROW and UDT elements is confusing.

Changes: Remove the reference to enclosing ROW and UDT types in 5.6.

Is there really a need to have the folder identifier?

Date: Sept. 7th, 2015

Sender: Bruno Ferreira (later annotated by H. Silva)

Document: https://docs.google.com/document/d/1-nsL2JNm01_dE_-rVVNqTGeuVNhcN7NyHiF-hdwS2y4/edit#heading=h.w1q77sepg2fa

Content: In column-level metadata (section 5.6) both *folder* and *lobFolder* elements are specified for LOB columns. Are both necessary? Can they be used simultaneously? Shouldn't one of them be mandatory?

Comment: The new *lobFolder* element was introduced as a „file:“-URI to help externalizing LOBs. The old *folder* element of SIARD Format 1.0 just contained the path of the LOB folder inside the SIARD file. The *folder* element is obsoleted by the new *lobFolder* element but is retained to ensure that SIARD archives conforming to 1.0 are still recognized as valid SIARD files.

Neither the *folder* element nor the *lobFolder* element are mandatory for every column as they only apply to LOB columns and may be missing as mentioned in the specification.

The description needs to be clarified.

Changes: clarify description

„parameters“ or „attributes“?

Date: Sept. 7th, 2015

Sender: Bruno Ferreira (later annotated by H. Silva)

Document: https://docs.google.com/document/d/1-nsL2JNm01_dE_-rVVNqTGeuVNhcN7NyHiF-hdwS2y4/edit#heading=h.w1q77sepg2fa

Content: In the metadata.xsd in the complex type routine the list of parameters is called „attributes“. Shouldn't it be „parameters“ as in SIARD Format 1.0?

Comment: This is a copy/paste error.

Changes: replace erroneous „attributes“ by „parameters“.

Contradictory qualifications in Specification and XSD

Date: Sept. 7th, 2015

Sender: Bruno Ferreira (later annotated by H. Silva)

Document: https://docs.google.com/document/d/1-nsL2JNm01_dE_-rVVNqTGeuVNhcN7NyHiF-hdwS2y4/edit#heading=h.w1q77sepg2fa

Content: In section 6.2 attributes file and length are defined as optional and in metadata.xsd as mandatory. Which is correct?

Comment: In either case these attributes were not really defined in SIARD Format 1.0. They should be optional in both places.

Changes: Remove „use='required'“ from table0.xsd and table1.xsd.

Unclear status of *timeType*

Date: Sept. 7th, 2015

Sender: Bruno Ferreira (later annotated by H. Silva)

Document: https://docs.google.com/document/d/1-nsL2JNm01_dE_-rVVNqTGeuVNhcN7NyHiF-hdwS2y4/edit#heading=h.w1q77sepg2fa

Content:In section 6.3 a timeType of metadata.xml is not mentioned.

Comment: The section 6.3 only refers to the SQL requirement of having dates between 0001 and 9999. The timeType is not concerned with this restriction as it does not contain a year.

Changes: none

timeType without milliseconds

Date: Sept. 7th, 2015

Sender: Bruno Ferreira (later annotated by H. Silva)

Document: https://docs.google.com/document/d/1-nsL2JNm01_dE_-rVVNqTGeuVNhcN7NyHiF-hdwS2y4/edit#heading=h.w1q77sepg2fa

Content:In *table0.xsd* and *table1.xsd* the *timeType* does not contain fractional seconds.

Comment: My automatic validation did not catch this discrepancy!

Changes: add fractional seconds to timeType according to SQL and XML standards.

„Restriction id enforced“

Date: Sept. 7th, 2015

Sender: Bruno Ferreira (later annotated by H. Silva)

Document: https://docs.google.com/document/d/1-nsL2JNm01_dE_-rVVNqTGeuVNhcN7NyHiF-hdwS2y4/edit#heading=h.w1q77sepg2fa

Content: In section 6.3 „restriction id enforced“ should read „restriction is enforced“.

Comment: Typo.

Changes: Correct typo.

XSD Version in namespace?

Date: Sept. 7th, 2015

Sender: Bruno Ferreira (later annotated by H. Silva)

Document: https://docs.google.com/document/d/1-nsL2JNm01_dE_-rVVNqTGeuVNhcN7NyHiF-hdwS2y4/edit#heading=h.w1q77sepg2fa

Content: The namespace of SIARD Format 2.0 has been defined as

`http://www.bar.admin.ch/xmlns/siard/1.0/metadata.xsd`
which is unchanged from SIARD Format 1.0. Should it not be

`http://www.bar.admin.ch/xmlns/siard/2.0/metadata.xsd`
Comment: The argument for 1.0 is that „old“ files still conform to the new

definition. The argument for 2.0 is that there are changes in metadata.xsd and it would be nice to have a different URI for it – particularly if one wanted to publish the metadata.xsd under this URI. Also it makes sense to use „2.0“ (rather than „1.1“ for example), because it refers to/defines the SIARD Format 2.0.

Changes: Change the namespace to http://www.bar.admin.ch/xmlns/siard/2.0/metadata.xsd.

Wrong requirement ID

Date: Sept. 7th, 2015

Sender: Bruno Ferreira (later annotated by H. Silva)

Document: https://docs.google.com/document/d/1-nsL2JNm01_dE_-rVVNqTGeuVNhcN7NyHiF-hdwS2y4/edit#heading=h.w1q77sepg2fa

Content: After T_6.4-1 requirement T_4.2-2 should really be T_6.4-2.

Comment: Copy/Paste error from version 1.0.

Changes: Correct erroneous requirement number.

Representing „null“ values

Date: Sept. 7th, 2015

Sender: Bruno Ferreira (later annotated by H. Silva)

Document: https://docs.google.com/document/d/1-nsL2JNm01_dE_-rVVNqTGeuVNhcN7NyHiF-hdwS2y4/edit#heading=h.w1q77sepg2fa

Content: Clarification of NULL (SQL) values and nil (XML) values is needed.

Comment: NULL (SQL) values should not be confused with nil (XML) values. XML nil are „empty“ XML elements with a nil attribute which asks the processor to treat them as present although they are not. In SIARD there are no „empty“ values, because all NULL (SQL) values are represented by omitting the corresponding column or field element. One reason for this decision was to keep SIARD archive files small. Also it prevents the confusion between NULL and „“, which are – unfortunately – identified by Oracle databases. A column <c8></c8> (or – equivalently <c8 />) contains the empty string '' and is not NULL. A missing column c8 indicates, that its value is NULL.

Changes: Clarify the criticized section T_6.4-3 of the text.

Representation of NULL values optional?

Date: Sept. 7th, 2015

Sender: Bruno Ferreira (later annotated by H. Silva)

Document: https://docs.google.com/document/d/1-nsL2JNm01_dE_-rVVNqTGeuVNhcN7NyHiF-hdwS2y4/edit#heading=h.w1q77sepg2fa

Content: The definition T_6.4-3 is marked optional. Should it not be mandatory?

Comment: It should be mandatory.

Changes: Change T_6.4-3 to mandatory.

„Necessity“ instead of „necessaty“

Date: Sept. 7th, 2015

Sender: Bruno Ferreira (later annotated by H. Silva)

Document: https://docs.google.com/document/d/1-nsL2JNm01_dE_-rVVNqTGeuVNhcN7NyHiF-hdwS2y4/edit#heading=h.w1q77sepg2fa

Content: In 3.2.2.1 replace „necessaty“ by „necessity“.

Comment: This really refers to the document of Anders „Recommendation for Folder Structure for Binary Data for the SIARD Format 2.0 and should be corrected there.

Changes: Correct typo in correct document.

Clarify folder name standard

Date: Sept. 7th, 2015

Sender: Bruno Ferreira (later annotated by H. Silva)

Document: https://docs.google.com/document/d/1-nsL2JNm01_dE_-rVVNqTGeuVNhcN7NyHiF-hdwS2y4/edit#heading=h.w1q77sepg2fa

Content: In P_4.2-6 the dot character is not permitted in folder names. In the LOB recommendation it is used.

Comment: The requirement P_4.2-6 only refers to ZIP elements inside the SIARD archive. The „file:“ URI syntax is much more permissive and allows anything the external file system supports. The requirement P_4.2-6 does not apply to external „file:“ URIs, which may refer to any kind of file system.

Nevertheless it may be useful to stick to P_4.2-6 even when you store files externally, but you may not always be able to do so. Therefore H. Silva's comment makes sense, that the LOB recommendation should use underscore („_“) in folder names instead of dots („.“).

Changes: clarify P_4.2-6. Plossible replace „.“ by „_“ in LOB recommendation.

Call for a full sample database

Date: Sept. 7th, 2015

Sender: Bruno Ferreira (later annotated by H. Silva)

Document: https://docs.google.com/document/d/1-nsL2JNm01_dE_-rVVNqTGeuVNhcN7NyHiF-hdwS2y4/edit#heading=h.w1q77sepg2fa

Content: A full example with external LOBs and UDTs is needed.

Comment: Such an example could/should accompany the specification together with the table XSD and XML examples. Unfortunately currently Enter AG has not yet been granted the resources from the Swiss Federal Archives to create such an example. But surely other archives can furnish one?

Changes: Add full example to specification.

Oct. 27th, 2015 Hartwig Thomas