Repräsentationen und Versionen in eCH-0160 und xIsadg

# Problemstellung

In eCH-0160 besteht die Möglichkeit, mit jedem abstrakten Dokument eine oder mehrere Dateien verbinden. Die genaueren Umstände und Anwendungsfälle für diese 1:n-Beziehung sind in der Dokumentation nicht näher erläutert. Wahrscheinlich war an ein Dokument bestehend aus einzelnen TIFF-Dateien (Seiten) gedacht.

Die inhaltlichen Metadaten zu einem Dokument sind mit der Entität *Dokument* verbunden (Titel, Autor, Erscheinungsform, Dokumenttyp, Entstehungszeitraum etc.); die mit dem Dokument verbundene Entität *Datei* besitzt neben rein technischen Metadaten (Originalname, Prüfsumme) ein Metadatum *Eigenschaft*, das auch als Eigenschaften–Werte Paar ausgebildet sein kann.

Damit ist zwar eine flexible Attribuierung von Datei möglich, ein späteres maschinelles Auslesen der Information aber schwierig.

Bei der Entwicklung von xIsadg wurde eine klare Trennung von inhaltlichen Metadaten (Abbildung in xIsadg) und technischen Metadaten (Abbildung in PREMIS) angedacht. Die Trennlinie zwischen den zwei Bereichen wurde in der Unterscheidung *Dokument* (abstrakt) und *Datei(en)* (technische Repräsentation eines Dokuments) gesehen. Es zeigt sich aber jetzt, dass auch inhaltliche Metadaten, das sind Metadaten, die zum Verständnis des *Dokuments* notwendig sind, auf Ebene *Datei(en)* anfallen. Es muss also konkret eine Möglichkeit geschaffen werden, *Datei:Eigenschaften* ebenfalls korrekt in xIsadg abzubilden, wenn möglich mit einer rückwärtskompatiblen Erweiterung des Standards.

# Konkretes Problem *(Bsp. SIP\_20190415\_2719\_31.zip)*

Das Staatsarchiv erhält Ablieferungen aus dem GEVER-System (CMI axioma) mit Versionen zu den einzelnen Dokumenten. Die Versionen sind als Dateien ausgebildet und in *Datei:Eigenschaften* entsprechend gekennzeichnet. Neben den verschiedenen Versionen gibt es auch den Fall, dass bereits in der Ablieferung verschiedene Repräsentationen des gleichen Dokuments als Dateien vorliegen (Word- und PDF-Datei).

In diesem Beispiel sehen wir vier Versionen des Dokuments "1 HofbrückeText Versionen", jeweils mit zwei Datei-Repräsentationen (DOCX und PDF). Die PDF-Repräsentation wurde offenbar automatisch beim Speichern der jeweiligen Version erstellt. ID d00015 ist Version 1 als DOCX-Datei.

Daneben haben wir ein Dokument "1 HofbrückeText", offenbar die finale Version, ebenfalls mit zwei Datei-Repräsentationen (DOCX und PDF).

Die in *Datei:Eigenschaften* gespeicherten Informationen: "Version", "Renderer", "EingechecktVon", "CheckInDate" können heute nicht auf Ebene Datei in xIsadg (*isadg:secondaryDataLocator*) festgehalten werden, sondern müssten in der verlinkten PREMIS-Datei untergebracht werden. Da die Information in Datei:Eigenschaften applikationsspezifisch kodiert ist ("Version", "Renderer", "EingechecktVon", " CheckInDate") können wir sie nicht in eine generische PREMIS Datei konvertieren, sondern müssen diese Informationen, so wie sie geliefert wurden, in xIsadg festhalten.

<datei id="d000015">

<name>1 HofbrueckeText.docx</name>

<originalName>1 HofbrückeText.docx</originalName>

<pruefalgorithmus>SHA-512</pruefalgorithmus>

<pruefsumme>316808539743b77e29e27009517be48afe5c0585c3e4d958e741d3dbc894baac684fdc593c58da3d9525cba9ffebee0be74a7efc0a2cd621a8f284826e0500cb</pruefsumme>

<eigenschaft name="Version">1</eigenschaft>

<eigenschaft name="Renderer">00000000000000000000000000000000</eigenschaft>

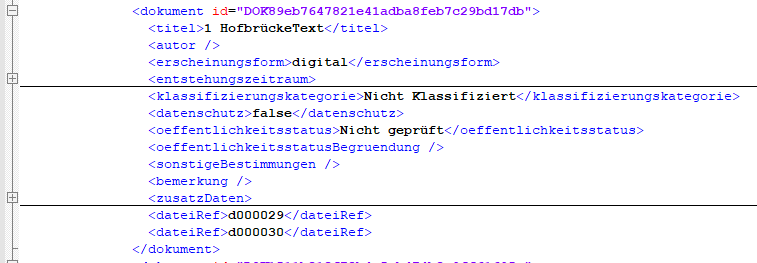
<eigenschaft name="EingechecktVon">Egloff Gregor; GREGOR.EGLOFF@LU.CH</eigenschaft>

<eigenschaft name="CheckInDate">15.04.2019 12:01:00</eigenschaft>

</datei>







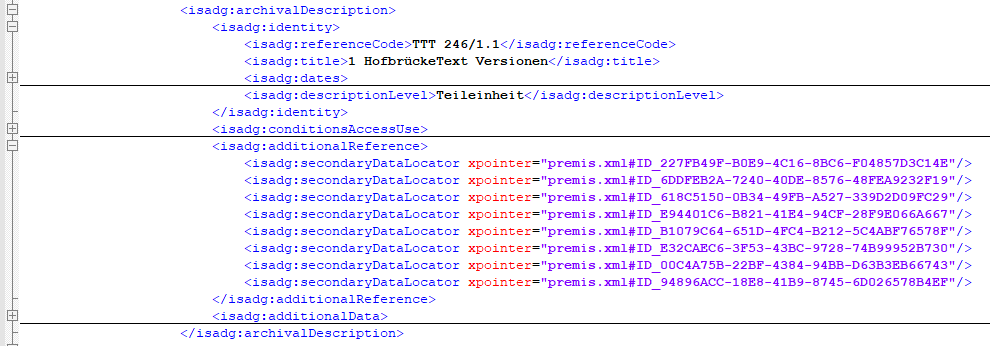
# Übernahme von Dateieigenschaften aus eCH-0160 SIP in xIsadg

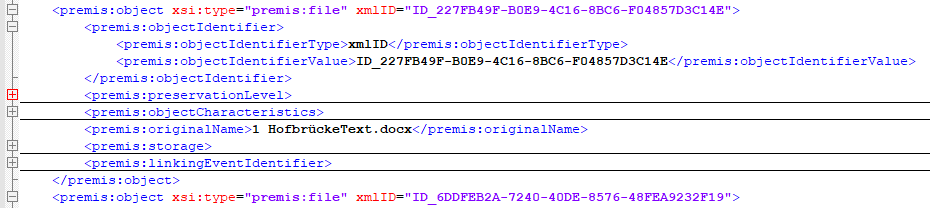
Eine erste Fragestellung lautet: Wie können die Dateieigenschaften aus eCH-0160 korrekt in xIsadg und in einem xIsadg-basierten AIP festgehalten werden? Die zweite Frage wäre: Wie können Versionen, Repräsentationen, Multidateidokumente und Dokumentaspekte in eCH-0160 SIP korrekt gespeichert und abgeliefert werden?

Wir konzentrieren uns hier auf die erste Frage. Die zweite Frage muss in einem anderen Rahmen beantwortet werden.

Dateieigenschaften in eCH-0160, auch wenn als Eigenschaften-Werte-Paare[[1]](#footnote-1) strukturiert, sind im Prinzip Freitextinformationen inhaltlicher und nicht technischer Art. Sie gehören also in den xIsadg-Bereich und nicht (weil unstrukturiert) in die zur Datei gehörige PREMIS-Datei.

*arelda:dateiRef* entspricht funktional *isadg:secondaryDataLocator* und verweist via XPointer auf einen entsprechenden Eintrag in der PREMIS-Datei. Die Eigenschaften müssen also hier als Erweiterung von *isadg:secondaryDataLocator* oder im entsprechenden *premis:object*-Element untergebracht werden.





Folgende drei Lösungen stehen zur Verfügung (in der Reihenfolge, in der diese von der AG scopeOAIS in der Sitzung vom 11. März favorisiert wurden):

## Erweiterung von *isadg:secondaryDataLocator* mit *additionalData*

Diese Erweiterung wäre kein vollständiges *isadg:additionalData-*Element, sondern müsste nur einen Metadaten-Wrapper für Key-Value-Paare enthalten. Die Möglichkeit, auf spezifische andere XML-Schema-Daten zu verweisen, ist ja durch den *secondaryDataLocator schon gegeben.*

<isadg:secondaryDataLocator xpointer="premis.xml#ID\_227FB49F-B0E9-4C16-8BC6-F04857D3C14E">

<isadg:additionalData>

<isadg:mdWrap>

<isadg:property key="Version" value="1"/>

<isadg:property key="Renderer" value="00000000000000000000000000000000"/>

<isadg:property key="EingechecktVon" value="Egloff Gregor; GREGOR.EGLOFF@LU.CH"/>

<isadg:property key="CheckInDate" value="15.04.2019 12:01:00"/>

</isadg:mdWrap>

</isadg:additionalData>

</isadg:secondaryDataLocator>

<isadg:secondaryDataLocator xpointer="premis.xml#ID\_6DDFEB2A-7240-40DE-8576-48FEA9232F19"/>

<isadg:secondaryDataLocator xpointer="premis.xml#ID\_618C5150-0B34-49FB-A527-339D2D09FC29"/>

**…**

Möglicherweise würden wir besser auf *isadg:additionalData* verzichten, um keine Verwirrung mit dem schon bestehenden *additionalData* Block von xIsadg zu erzeugen. Hier also eine Variante nur mit einem *mdWrap*-Element (Metadaten-Wrapper):

<isadg:secondaryDataLocator xpointer="premis.xml#ID\_227FB49F-B0E9-4C16-8BC6-F04857D3C14E">

<isadg:mdWrap>

<isadg:property key="Version" value="1"/>

<isadg:property key="Renderer" value="00000000000000000000000000000000"/>

<isadg:property key="EingechecktVon" value="Egloff Gregor; GREGOR.EGLOFF@LU.CH"/>

<isadg:property key="CheckInDate" value="15.04.2019 12:01:00"/>

</isadg:mdWrap>

</isadg:secondaryDataLocator>

<isadg:secondaryDataLocator xpointer="premis.xml#ID\_6DDFEB2A-7240-40DE-8576-48FEA9232F19"/>

<isadg:secondaryDataLocator xpointer="premis.xml#ID\_618C5150-0B34-49FB-A527-339D2D09FC29"/>

**…**

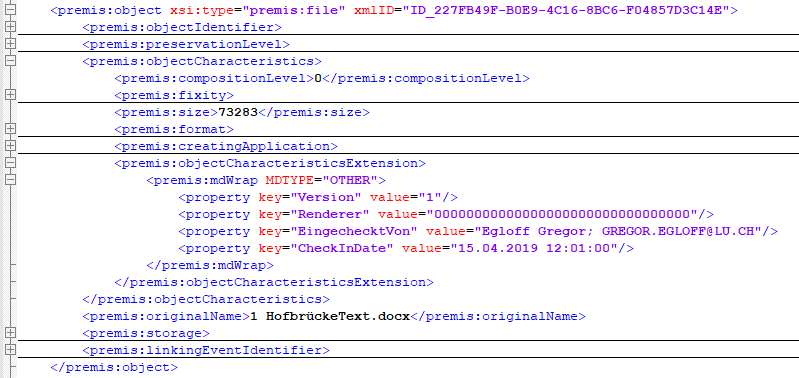
*Vorteil*: Nach der Logik von xIsadg und ISAD(G) befinden sich die Informationen am richtigen Ort.

*Nachteil*: Diese Lösung verlangt eine neue Version von xIsadg, das Schema wäre aber vollständig rückwärtskompatibel. Der Inhalt, da nicht in der xIsadg Syntax, ist nicht direkt maschinenlesbar.

## *premis:objectCharacteristicsExtension* verwenden

Die Dateieigenschaften aus eCH-0160 werden in das entsprechende premis:object, das dieser Datei zugeordnet ist, übernommen. Am ehesten kommt dafür das Element *premis:objectCharacteristicsExtension* in Frage:

Das würde dann so aussehen:



*Vorteil*: Es sind keine Änderungen an xIsadg v2.1 notwendig

*Nachteil*: *premis:objectCharacteristicsExtension* ist nicht eigentlich für diese Art Informationen gedacht[[2]](#footnote-2) und der Inhalt, da nicht PREMIS Syntax, ist nicht direkt maschinenlesbar.

## Für jede Datei eine eigene *isadg:archivalDescription*

Für jede Primärdatei wird eine eigene Verzeichnungseinheit (*archivalDescription)* erzeugt. Im Ansatz ist das im Beispiel mit den Dokumenten "1 HofbrückeText Versionen" und "1 HofbrückeText" bereits in eCH-0160 SIP angelegt.

Damit würde es möglich, die Metadaten zur Primärdatei in xIsadg-Syntax richtig abzubilden. Das würde natürlich voraussetzen, dass im Ingestfall die Dateieigenschaften richtig interpretiert werden können; werden sie nur als Key-Value-Paare in *isadg:additionalData* abgelegt, ist nicht viel gewonnen.

Wir sehen im Beispiel die geschachtelten *archivalDescription,* eine mit dem *descriptionLevel* Teileinheit, die andere mit dem *descriptionLevel* Repräsentation, von wo aus auf die PREMIS-Datei verwiesen wird. In dieser geschachtelten Verzeichnungseinheit "Repräsentation" befinden sich schliesslich im additionalData Wrapper die Dateieigenschaften.



*Vorteil*: Es sind keine Änderungen an xIsadg v2.1 notwendig

*Nachteil*: Es müssen beim Ingest zusätzliche Verzeichnungseinheiten angelegt werden, die einzig und alleine Dateivarianten beschreiben.

1. Die Eigenschaften-Werte-Paare sind nur im Zusammenhang mit der abliefernden Stelle/Applikation maschinell auswertbar. [↑](#footnote-ref-1)
2. <http://www.loc.gov/standards/premis/understanding_premis_german2017.pdf> Seite 8 "Der Erweiterungscontainer objectCharacteristicsExtension (objektMerkmaleErweiterung) bietet einen Platz für die Verzeichnung technischer Metadaten" und <https://www.loc.gov/standards/premis/v2/premis-2-2.pdf> Seite 73 "objectCharacteristicsExtension is used for additional object characteristics not covered by PREMIS, for instance format specific metadata that is defined externally. It is not a replacement for units specified in PREMIS" [↑](#footnote-ref-2)