fhiso

CFPS 65
(Call for Papers Submission number 65)

A Gradual Path to Standardised Citations

Submitted by: Tychonievich, Luther

Created: 2013-04-26

URL: Most recent version: http://fhiso.org/files/cfp/cfps65.pdf
This version: http://fhiso.org/files/cfp/cfps65_v1-0.pdf

Description: Citations are a mix of out-references and cross-references. These can be standardised independently of a Citations Names Authority.

Keywords: sources, citations, names, outref, xref, cross-references, xlink, pointers, variants

A Gradual Path to Standardised Citations

Luther A. Tychonievich

26 April 2013

Abstract: *Citations, meaning references to real-world documents and sources, are a fundamental component of family history data. I propose a three-part standard for citations that separates the distinct concerns of a data model, a serialisation format, and a standardised set of citation types and fields. This separation also allows for pre-standard citations to use the the standard data model and serialisation until they can be converted to a standards-compliant form.*

1 Definition and Introduction

Citations are essential to family history and genealogy information because the ultimate source of almost every element considered in any history is some record, artefact, conversation, memory, hunch, or other real-world thing. Citations in some form are important in describing and communicating what these real-world things are independent of their specific contributions to research.

The word "citation", like many words in family history, is used to mean different things by different people. I will explicitly address the following five definitions:

1. A pointer from something believed or asserted to the source or reasoning behind that belief or assertion. See Section 5
2. A digital cross-reference pointing to another element within the same digital tool or file. See Section 5.
3. A digital out-reference pointing to some phenomenon, artefact, or other thing external to the digital tool or file in question. See Section 4.1
4. A print cross-reference pointing to a bibliography. See Section 5.
5. A print out-reference typically in the form of a bibliography entry. See Section 4.3.

The bulk of this paper will address definition 3, which I will call **OutRef**s for the purpose of disambiguation. This focus is motivated by the observation that

OutRefs, combined with the well-understood cross-references (definition 2, also called **XRef**s) allow us to achieve all of the other definitions.

The OutRefs I will focus on are descriptions of real-world things, as opposed to digital things in other tools. Existing identification schemes such as Handles [1], DOIs [2], or URIs [3] can be used to handle digital things from other tools and can be easily included into my conversation.

One common characteristics of OutRefs is that they identify things primarily by publication, location, or origin rather than by content, appearance, or meaning. Nothing in this paper is restricted to this common use---the techniques discussed herein could identify resources by content instead or as well if desired---but it is supported and the only case discussed explicitly herein.

Digital OutRefs may exist independent of any definition citations using them. This independent existence is supported in every sourcing tool of which I am aware; for example, LaTeX calls these entities "bibitems"; Word calls them "sources"; BibTeX and LibreOffice call them "entries"; etc.

2 Why Standardise Citations?

In most extant family history tools, the only audience for a citation is a human. In general, computers either cannot or do not verify or act on citations in any direct way. Hence, one might argue that the only standard needed is "understandable to a human."

However, there are multiple reasons, both human and computational, why citation standards are valuable. These include:

- Structured information is easier to translate and localize than is free text.
- It requires less mental effort to read uniformly-formatted citations than mixed style citations.
- Structured citations can be automatically validated for completeness.
- Structured citations allow computer-facilitated collaboration between researchers working off the same sources.
- Search assistant tools might be devised that intelligently suggest records based on the set already cited if those citations are understood by the tools.
- Citation understanding might be an asset for tools that evaluate the strength of and/or gaps in reasoning.

The above list is not exhaustive, but is enough to motivate the need for standardised citations.

3 Handling Not-Standardised Citations

Significant numbers of existing family history datasets contain citations. Since these are not currently in a standardised format, it is likely that, for any given standard form, at least some of these citations cannot be easily and automatically converted to that form. Thus, no matter what the standard selected, there will be a non-trivial number of extant citations that do not match the standard.

To minimize the transitional cost associated with a standardised citation form, it is desirable to allow for the coexistence of standard-compliant and substandard OutRefs in the same database or record. Hence a standard should include the ability to store standardised content, non-standard fields, and uninterpreted text.

4 Three-Part Citation Standard

I propose we adopt a three-part citation standard. The first part is a general data model for representing OutRefs, one that might be incorporated into a larger data model or adopted independently. The second part is a names authority group to agree on canonical names of the various citation fields of various kinds of cited material. The third part is a (set of) standard serialisation formats for converting the data model to and from files and network traffic.

4.1 OutRef Data Model

Am OutRef may be characterized as an identifier of the type of thing it is citing and an unordered set of fields describing the particular thing of that type being cited.

4.1.1 Type

The type of a citation specifies what kind of document, artefact, or other thing is being described. The set of types will likely types include “book”, “census”, “conversation”, “web site”, “deed”, “obituary”, etc. The specified set of types used will be determined by the Citation Names Authority (see Section 4.3).

I expect that at any given point in time there will be an accepted hierarchy of types; perhaps a “census” is a subtype of a “government survey” or the like. However, I also suspect that the structure of the hierarchy will change from time to time as more appropriate organisations are proposed. I hence suggest that every citation type be given a single unique identifier. If, however, the names authority determines that the hierarchy is stable and part of the standard then a list of types (from most general to most specific) may be used instead.

A special type should be introduced for OutRefs that have not yet been converted to standardised form and whose proper type has thus not been identified. I suggest some version of “pre-standard” as the name of that type because the more common term in computing, “legacy,” has a different meaning in family history.

It should be possible to create OutRefs whose type is not any of the standard types. This will allow for citation to types of resources that have not yet been approved by the names authority. I suggest that these extension types should contain a mandatory field indicating their non-standard format so that they may be distinguished from OutRefs that have an incorrect type by mistake instead of intent.

4.1.2 Fields

An OutRef is primarily a set of fields. Each field is a key:value pair. The keys will most likely be ASCII text (though unicode might also be appropriate) taken from a type-specific set defined by the Citation Names Authority (see Section 4.3), while the values might be any suitable standardised datatype. For example, a citation to a book might contain a field with the key "author" whose value is a list of Personal Name objects.

The fields of any given type may be characterized in four classes:

Mandatory: Without this field the OutRef is incomplete. For example, the year will likely be mandatory for a census.

Optional: The meaning of the field is known, but not all OutRefs of this type will have it. For example, the editor will likely be optional for a book.

Annotative: This field is optional and describes something other than the citation itself. For example, an "accessed" field might be an annotation for an archival record (but might be mandatory for a web document).

It is common in family history to annotate individual fields of a citation. For example, to cite this page of this document I might include a mandatory field "page" with value "4" as well as an annotative field "page-source" with value "typeset at bottom of page as '4 of 8'". I suggest that a standard set of field annotation naming schemes (perhaps suffixes like "-source", "-note", "-alternative", etc.) be selected by the Citation Names Authority (see Section 4.3).

Extra: Any field not in the above three categories is "extra," meaning it is not part of the standard fields for the type in question. Extra fields would most likely arise from transfer of pre-standard citations into standard OutRef form or from vendor-specific extensions to the standard. The keys and values of extra fields are unconstrained.

Pre-standard type OutRefs would most likely contain a mandatory field with a key like "citation representation" so that they may be displayed prior to being converted to standard form.

Variant Datatypes as Field Values
It may be the case that a single field might be able to admit multiple datatypes as its value. For example, it might be decided that an "author" field may have a UTF-8 string or a UTF-16 string or a name object or a list object containing strings and name objects. For non-extra fields, variant datatypes can be avoided by design (e.g., we could define all "author" fields to be (possibly singleton) lists of name objects), but including variants might still be desirable. Extra fields, being individually outside the standard, will have unknown datatypes to tools other than the originating tool. If they are to be interpretable by such tools, variant datatypes are required.

Multiple datatypes can be stored in the same field using some form of disjoint union, variant, or dynamically-typed value. These should be included in the data model and the serialisation scheme. Many implementation of these concepts exist in computing, so they do not pose a technical challenge, but they do add complexity to software. A decision to introducing variants in standardised fields or to require the ability to interpret other tools' extra fields should only be made if the expected benefits justify the software engineering effort variants represent.

Multiset Fields
There can be a case made for allowing multiple fields in a single OutRef all sharing the same key. One might easily picture several "author-note" fields, for example, each documenting a different note about the author. However, multisets introduce a variety of technical and interpretive challenges and I would suggest that single keys whose values are sets are preferable in almost every instance.

4.2 Citation Names Authority

A standing committee ought to be formed to standardise the types of OutRefs that FHISO recognizes. For each type, the committee ought to identify the sets of mandatory and optional fields for that type and the kinds of values that each field may contain.

As I lack the background to speak to the likely types and fields such a committee might identify, I say no more about it here.

4.3 OutRef Serialisation

In the abstract data model, an OutRef is a (type, fields) pair, where the fields element of this par is a set of key:value pairs. There are many ways this might be serialized. For example, an OutRef pointing to this paper might be serialized as any of the following (making up the unlikely type "fhisocfps" for expository purposes):

```
<fhisocfp author="Tychonievich, Luther A." date="2013-04-26"
cfps="64" title="A Gradual Path to Standardised Citations" />
```
(XML tag style)

```
<citation type="fhisocfp">
  <author>Tychonievich, Luther A.</author>
  <date>2013-04-26</date> <cfps>64</cfps>
  <title>A Gradual Path to Standardised Citations</title>
</citation>
```
(XML element style)

```
{ "type":"fhisocfp", "author":"Tychonievich, Luther A.",
"date":"2013-04-26", "cfps":"64",
"title":"A Gradual Path to Standardised Citations" }
```
(JSON style)

```
@fhisocfp { cfps64, author={Tychonievich, Luther A.},
year={2013}, month={04}, day={24}, cfps={64},
title={A Gradual Path to Standardised Citations} }
```
(BibTeX style)

This set is of course not complete. The actual serialisation format(s) selected should match similar selections made for other data serialisations.

Textual citations such as those that appear in bibliographies and printed reports (definition 5) are also serialisations. Examples of these human-centric textual serialisations include:

> Tychonievich, Luther A. (2013) "A Gradual Path to Standardised Citations." *FHISO Open Call for Papers 2013*. CFPS 64.
>
> L. A. Tychonievich. *A Gradual Path to Standardised Citations.* FHISO Open Call for Papers no. 64. 4 April 2013.
>
> Tychonievich, L. "A Gradual Path to Stnd. Citations." FHISO CFPS 64. 2013.

or any of a myriad similar citation styles. Such textual citation styles are generally inappropriate for digital transfer because they are almost universally lossy: they do not generally include the annotations or extra fields and often store multiple types in similar or even indistinguishable formats.

All serialisations, digitally complete or textually presentable, may be generated dynamically given a set of rules and the OutRefs discussed in Section 4.1.

5 Citing with XRefs and OutRefs

Having discussed citation definitions 3 and 5, I now address the other three definitions.

The easiest of these definition to describe is cross-references or XRefs (definition 2). Cross-references are one of the foundational building blocks of computing and have many well-understood implementations (e.g., address and pointer [4], xml:id and xlink:href [5], xpath [6], title and fragment identifier [7], in-document

URI [3], etc.). Any of these extant implementations is suitable for XRefs used in citation.

Definition 1 was "a pointer from something believed or asserted to the source or reasoning behind that belief or assertion." This may be accomplished attaching to each belief or assertion an XRefs to its source or reasoning, which may be an internal element of the digital model or may be an OutRef.

Finally, definition 4 was "a print cross-reference pointing to a bibliography." These are human-centric serialisations of XRefs, and can be handled automatically in any of a myriad citation styles, such as "[1]", "[1]", "(Tyc13)", "(Tychonievich 2013a)", etc.

6 Sub-citations

In many instances it may be useful to cite both a larger document and one or more smaller parts of that document. I suggest this may be accomplished by an optional "within" field in an OutRef which contains an XRef to another OutRef. For example, to cite both this document and this page of this document I might send the following two OutRefs (in XML tag style)

```
<fhisocfp author="Tychonievich, Luther A." date="2013-04-26"
cfps="64" title="A Gradual Path to Standardised Citations"
xml:id="doc1" />
<fhisocfp page="7" within="doc1" xml:id="doc1p7" />
```

Note, however, that such an extension is merely a data compression shorthand; the meaning is equivalent to sending both OutRefs in full.

```
<fhisocfp author="Tychonievich, Luther A." date="2013-04-26"
cfps="64" title="A Gradual Path to Standardised Citations"
xml:id="doc1" />
<fhisocfp author="Tychonievich, Luther A." date="2013-04-26"
cfps="64" title="A Gradual Path to Standardised Citations"
page="7" xml:id="doc1p7" />
```

Subcitation presentation formats are also common in documents; for example, the above two citations might be displayed with only one bibliography entry and distinguished using in-text style as, e.g., [1] and [1 page 7]. Since within-style compression is unambiguous and fully automatable, the decision to *store* citations flatly or with subcitations is independent of the decision to *present* them flatly or with subcitation notation.

7 Conclusion

Citations are a fundamental element of genealogical data. Putting them in a standard computer-readable format will simplify and/or enable a wide range of computational and researcher tasks. Modeling using a mix of XRefs and OutRefs,

where OutRefs are a type and a set of named fields, allows for a general data model to handle a variety of different things that might be cited. Having a separate Citation Names Authority will allow the particular citation formats to be created and maintained with minimal impact on the software and information standards used. Additionally, a special type "pre-standard" will allow existing non-standardised citations to be maintained within the new standard data model until they are updated.

Acknowledgements

I am indebted to GeneJ for suggesting I work on this problem and for her helpful comments on earlier drafts of this paper.

References

[1] "Welcome to the Handle System", http://handle.net/. Retrieved 2013-04-26.

[2] "The DOI System", http://www.doi.org/. Retrieved 2013-04-26.

[3] URI Planning Interest Group. "URIs, URLs, and URNs: Clarifications and Recommendations 1.0", 2001-09-21.
http://www.w3.org/TR/uri-clarification/. Retrieved 2013-04-26.

[4] Lawson, Harold W. *PL/1 Language Specifications*. IBM Report C28-6571. New York, NY, 1965.

[5] DeRose, Steve and Maler, Eve and Orchard, David and Walsh, Norman. "XML XLink Language (XLink) Version 1.1", 2010-05-06.
http://www.w3.org/TR/xlink11/. Retrieved 2013-04-26.

[6] Clark, James and DeRose, Steve. "XML Path Language (XPath)", 1999-11-16.
http://www.w3.org/TR/xpath/. Retrieved 2013-04-26.

[7] Berners-Lee, Tim. "URI References: Fragment Identifiers on URIs", 1997-04.
http://www.w3.org/DesignIssues/Fragment.html. Retrieved 2013-04-26.