

CFPS 37

(Call for Papers Submission number 37)

Proposal for a scalable extensibility mechanism

Submitted by: Smith, Richard

Created: 2013-04-12

URL: Most recent version: <http://fhiso.org/files/cfp/cfps37.pdf>
This version: http://fhiso.org/files/cfp/cfps37_v1-0.pdf

Description: Proposal to address certain extensibility requirements by using XML namespaces, but without requiring the use of XML.

Keywords: extensibility, vendor extensions, forwards compatibility, decentralised development, XML namespaces, URIs

Abstract

This paper examines the importance of extensibility in a genealogical standard and identifies three key principles for an extensibility mechanism. It examines current practice in GEDCOM and how it fails these principles. The extensibility mechanisms used in some other fields are also considered, and in particular XML's. The paper proposes the adoption of XML Namespaces to provide for extensibility in a FHISO standard, notwithstanding the fact that the FHISO may adopt a non-XML serialisation format. Such a scheme requires the use of URIs to identify vendor-specific namespaces, and the paper concludes with discussion on how these might be used.

1 Introduction

There several reasons why a new genealogical standard should have a well-defined mechanism for extension. These reasons can be divided into three broad groups:

- accommodating vendor-specific extensions;
- anticipating the need for forwards compatibility with a later standard; and
- recognising the modularity required in certain expert systems.

The first case, that of vendor-specific extensions, is the simplest. Experience in any fields, not just genealogy, suggests that vendors will find reasons for wanting to extend a standard. The business need to provide their product with a unique selling point is a frequent reason. Such extensions might be to store additional metadata relevant to that vendors' application. An example can be found in the common _UID extension to GEDCOM [1]. This example also highlights a problem with such extensions: they are used incompatibly by different vendors, with different formatting or semantics assumed in different applications. Such naming conflicts can cause problems migrating data between applications.

But good support for extensions is particularly important to a genealogical standard because the size and complexity of the field. No standard can feasibly hope to standardise everything, and therefore vendors are likely to experiment with a new features, perhaps with a view that they may eventually be standardised. The GEDCOM 5.5 EL extensions that improve GEDCOM 5.5's handling of places is an example of that [2]. In the future, the FHISO, or a successor organisation, may decide to standardise certain of these vendor extensions; and the FHISO may also wish to develop the standard itself in new ways. Irrespective of how such developments arise, if a proposed FHISO standard gains traction within the community, it seems certain that the first version will be succeeded by a second version.

Finally, certain areas are inherently open-ended and an extensible, modular approach best suits them. The world has seen scores, probably hundreds, of cal-

endar systems. It is implausible to suggest that the FHISO can standardise each of these, but it is also unreasonable to expect vendors to only ever support the standard ones. An Icelandic vendor may well want to support the old Icelandic calendar, for instance. Personal names have many different possible components, and a standard should accommodate the tagging of matronymics or Japanese post-nomous names, even though they may not be standardised. Similar consideration apply to geopolitical data, such as the types of subnational divisions.

2 Principles

Avoid name conflicts

There should be no danger of conflict between extensions made by different vendors. Nor should a vendor extension conflict with future extensions made by the FHISO or its successors. This should also apply to the likes of calendars: one vendor's implementation of the old Icelandic calendar may not be compatible with another vendor's.

Easy reuse of extensions

Where one vendor reuses an extension first introduced by another vendor, it should be able to indicate that this is the case. This should be possible without action or consent from the original implementation. (This is a technical requirement as this paper does not consider legal issues associated with patents or copyright.)

Decentralised development of extensions

The logistical barrier to creating an extension should be low. An individual developer wishing to develop an extension should be able to do so *in vacuo*, without registering the extension with a central authority such as the FHISO.

3 Current practice

Where extensions have been made to GEDCOM outside the official standard, there is an informal convention that the tags associated with extension are prefixed with an underscore. This avoids name conflict between vendor extensions and future standardisation, but as the example of `_UID` demonstrated, it does not prevent conflict between vendors [1].

In certain relevant ISO standards, such as ISO 639 language codes, allocation of identifiers is overseen by a central registration authority (the Library of Congress for ISO 639), and new codes are only allocated after application to and review by the registration authority [3]. Private extensions are permitted with a 'x-' prefix, but there is no means for preventing conflict between extensions.

In a field as self-contained as the enumeration of languages and with the resources of Library of Congress, it is reasonable to hope that most languages will be identified and tagged, and the need for private extensions therefore rare. Genealogy, as a whole, is not as self-contained, nor are the resources of the FHISO comparable to those of the Library of Congress. Consequentially, the need for private extension is therefore much greater as is the risk of conflict between extensions.

Many languages based on XML have solved this problem using XML Namespaces [4]. Element names (and in some circumstances, other names) have an optional prefix, separated from the main part of the name by a colon. Such a name is referred to as a QName, and `x:name` and `name` are example with and without a prefix. The prefix is short-hand for a lengthier unique identifier (a URI, specifically) to which the prefix must be bound in a specified manner.

The XML Namespaces mechanism for using QNames as tags and binding prefixes to URIs has been reused in several formats that are not based on XML. The World Wide Web Consortium's CURIE standard for compact URIs uses it [5], as does the Turtle language [6].

4 Proposal

This paper proposes that a future genealogical standard should mandate that any extensions should be made in a vendor-specific namespace. Each namespace must be formally identified by a URI and references to vendor-specific concepts must be via a prefixed QName, as defined by the XML Namespaces specification [4]. The serialisation format adopted must provide syntax for binding prefixes to namespace URIs, and in the case that an XML serialisation is adopted by the FHISO, the namespace binding mechanism in XML Namespaces must be used.

This paper does not seek to influence what the choice of serialisation format should be, and the proposal to adopt one mechanism from XML does not make it necessary to use XML for serialisation. For example, the present GEDCOM format could be extended with a new XMLNS header tag to bind namespaces, and prefixed QNames used as GEDCOM tags.

Where extensible tag names are required for the likes of calendars, it is suggested that they too should be QNames. Any standardised calendars can be unprefixes, while non-standard ones can go in a vendor namespace.

For consistency, this paper proposes that the FHISO allocate a namespace URI to the default namespace (i.e. the namespace of FHISO-defined terms). Whether any form of declaration is required to use the default namespace in a non-XML format is a implementation detail of that serialisation format, and is not specified here.

5 Namespace URIs

The use of URIs to identify namespaces has several advantages, not least of which is that it is already standardised [7]. Many URI schemes (and particularly the most common http scheme) use domain names, and mechanisms already exist for the allocation of domain names. For developers who do not have a suitable domain name, sites such as <http://purl.org/> exist to provide people with persistent URIs. In the event that a developer has good reason not to want to use a domain-based URI for the namespace, a mechanism exists for using UUIDs as URIs [8]. However, this paper proposes that a FHISO standard should recommend (but not mandate) the use of http URIs.

No requirement is made by the XML Namespaces standard that the namespace URI can be used to fetch any document or resource. It is simply an identifier. Nevertheless the URI is commonly used as a means of fetching documentation on the namespace, and this paper proposes that the FHISO recommends (but not mandate) that, if the vendor wishes to make documentation available on the extensions provided in the namespace, it should be accessible from that URI.

References

- [1] Tamura Jones, 2012, *The _UID tag — Common GEDCOM Extension* (blog entry), http://www.tamurajones.net/The_UIDTag.xhtml
- [2] GenWiki website, 2009, *Gedcom 5.5EL*, http://wiki-en.genealogy.net/Gedcom_5.5EL
- [3] Internet Engineering Task Force, 2009, *Tags for Identifying Languages* (RFC 5646), <http://tools.ietf.org/html/rfc5646>
- [4] World Wide Web Consortium, 2009, *Namespaces in XML 1.0 (Third Edition)*, <http://www.w3.org/TR/REC-xml-names/>
- [5] World Wide Web Consortium, 2010, *CURIE Syntax 1.0 — A syntax for expressing Compact URIs*, <http://www.w3.org/TR/curie/>
- [6] World Wide Web Consortium, 2013, *Turtle — Terse RDF Triple Language*, <http://www.w3.org/TR/turtle/>
- [7] Internet Engineering Task Force, 2005, *Uniform Resource Identifier (URI): Generic Syntax* (RFC 3986), <http://www.ietf.org/rfc/rfc3986>
- [8] Internet Engineering Task Force, 2005, *A Universally Unique Identifier (UUID) URN Namespace* (RFC 4122), <http://www.ietf.org/rfc/rfc4122>