

Selector Language

Authors Luther A. Tychonievich

Created 2013-03-15

Title Selector Language for Family History Conclusions

Version 1.0

Type Other Standards

Language English

Description There should be a standard way to express things like “two people, one the second cousin of the other, both from Prussia.”

Keywords selector, template, relationship, attribute, conclusion, regular expression, XPath

Selector Language for Family History Conclusions

Luther A. Tychonievich

12 March 2013

Abstract: *There should be a standard way to express things like “brother of an aunt” or “second cousin by marriage” or “two people, one the ancestor of the other, both from Prussia.”*

When generating reports, answering queries, and otherwise inspecting a data set it is useful to have an unambiguous selector language over the data. Examples of successful selector languages include XPath for XML [3], CSS Selectors [2], and regular expressions for text (see, e.g., [1]).

The data to be matched in genealogy and family history has a different structure than other data applications. In general it is a graph where both nodes and edges may have different types and attributes. I am unaware of any existing selector language or template design that can easily accommodate the kinds of queries we might pose over a genealogical conclusion.

1 Partial Requirements

A selector language ought to be able to reference at least the following:

- People, places, and events
- Attributes of people, places, and events, such as names and dates
- Multiple elements with particular relationships
- Partially-specified attributes and relationships
- Variable numbers of nodes (e.g., “ancestor” rather than just “grandparent”)
- Internal back-references (e.g., “two people with the same name”)

Additionally, it should specify how to refer to each matched component, both single- and set-valued; for example, given a selector encoding “two people with the same name, one the ancestor of the other” it should be possible to refer to the ancestor, to the descendant, and to the set of intervening people who connect them.

2 Ideas about Implementation

XPath has a reasonable attribute matching language that might be extended to genealogy.

Regular expressions over relationships might handle much, but not all, of the expressive power needed for relationships. In particular, some kind of “and” and “not” operators are needed in addition to regular expressions’ existing “or” operator “|”. An example of where “and” and “or” are needed is in expressing “aunt”:

(mother|father){2}((son)(wife)|daughter) “and not” mother

It is my belief that, no matter how data is modelled internally, the selector language ought to be over conclusions. Data models that support storing steps in reaching a conclusion should still apply the selector to the conclusion reached, not to the path used to reach it.

References

- [1] Perl 5 Porters. *Perlre: Perl regular expressions*. Official documentation for the Perl programming language. <http://perldoc.perl.org/perlre.html>
- [2] World Wide Web Consortium. *Selectors Level 3*. W3C Recommendation. <http://www.w3.org/TR/css3-selectors/>
- [3] World Wide Web Consortium. *XML Path Language (XPath) 1.0*. W3C Recommendation. <http://www.w3.org/TR/xpath/>