From XML to RDF in the Orlando Project

John Simpson
Text Mining & Visualization
For Literary History / INKE
University of Alberta
Email: john.simpson@ualberta.ca

Susan Brown
Department of English
University of Alberta /
University of Guelph
Email: sibrown@ualberta.ca

INKE Research Group
Modelling & Prototyping Team
Canada

***Abstract*—This short paper offers a case study in the challenges associated with the translation of content from a particularly rich set of XML tags in a boutique digital humanities project–*The Orlando Project*–into RDF.**

***Index Terms*—RDF; XML; Orlando; conversion**

I. Introduction

Research projects and content providers in the humanities such as libraries and museums are increasingly incorporating semantic web technologies [1, 5]. As part of this process, many ontologies initially developed for other contexts are being translated into semantic-web-ready forms to enable the leveraging of existing metadata in a semantic web context [4, 8]. XML schemas in particular are targeted for such translation, and production of RDF based on existing XML markup is increasing, with the W3C offering a conversion tool meant to facilitate such translations [9].

Yet such translation is fraught with challenge, particularly when rigid ontological hierarchies common in the sciences are asserted on humanities data [6]. The migration of information to formats ready for the semantic web reopens important questions about cultural artifacts that were less relevant in previous formats. In particular the use of RDF to describe the world and to link aspects of it together mobilizes the hierarchical structures of XML in ways that may have unforeseen consequences for knowledge representation and result in erroneous assertions.

This short paper offers a summary of technical challenges associated with the translation of content from rich XML in a boutique digital humanities project–*The Orlando Project*–into RDF.

II. About Orlando

Orlando: Women's Writing in the British Isles from the Beginnings to the Present is a born-digital reference and research resource comprised of more than 1,300 detailed biocritical entries, capturing over 27,000 individual people [2]. The prose within each entry has been marked up using XML to identify information related to authors' lives and writing careers, contextual material, timelines, sets of internal links, and bibliographies [3]. Besides being leveraged to both structure and search the documents within *Orlando*, the XML encoding is now being used to assist in the automatic conversion of RDF from *Orlando's* structured text.

The *Orlando* XML tag set is extensive, comprising 186 individual tags, 95 attributes, and 507 fixed attribute values that may be applied multiple times. These tags can be divided into four categories: provenance, structural, context, and content. Provenance tags capture information about who has contributed to an entry, how, and when. Structural tags capture the formal properties of the document, such as <DIV>s and <P>s. Context tags provide meta-level descriptions of the structure of the entries. For example, many entries have a discussion about family and so there is a context tag that wraps the entirety of this section. The content tags capture the properties of the particular things that are said about particular people, such as tags for birth events and gender.

A set of conventions, documented extensively in the project's internal production system and used when authors are tagging the documents as they write, accompanies the tag set, providing further structure. Such conventions include tagging an author's name most times it is used within a new <DIV> tag within any entry and wrapping any locations referenced in three layers of location tags (settlement, region, and broader geographic area).

III. Conversion Method

This combination of tags and conventions allows for a great deal of information to be extracted from *Orlando* through an automated process that leverages these structures and then responds by creating RDF triples.

This extraction is performed by a Python script that uses regular expressions to directly target structures in the document created by the intersection of XML and the tagging conventions. Each correctly matched expression produces a single RDF triple for an entry about an author.

This approach deviates from standard approaches to the conversion of XML documents which use tools such as XSLT or XQUERY to target items within the Document Object Model. The principal reasons for this deviation are that the scripting method employed allows project members to quickly write single lines of regular expressions to target new RDF triples rather than a combination of XPATH, regular expressions, and a translation language. It also reduces the amount of redundant code and therefore the likelihood of errors.

The Python script reads through a list of commands that look like the following:

givenName|<GIVEN>([\w]+)</GIVEN>

Two pieces of information are shared here and broken up by the pipe character. The first is the name of the predicate to be extracted (in this case, *givenName*). The second is the regular expression used to capture the given name of the primary individual for whom a given entry is about. It amounts to looking for the opening and closing tags that surround a given name and then capturing all the text within. Since the target of the entry is known before executing this code, the result is the return of all the components for an RDF triple. The program then assigns each element a URI and outputs a line of RDF/XML.

Statements like the one shared previously work only because of the combination of tags and conventions used in the entries. To see this just consider how problematic it would be if *every* person mentioned in an entry had their given name tagged. Fortunately, this is not the case. By convention, only the person who is the primary focus of the entry has such properties tagged, allowing for triples about them to be easily extracted.

Not every case can be picked out by a single regular expression since not all relevant information is captured within a single XML tag. The place of death is a case in point. Here it is the hierarchy of the XML tags that is particularly helpful. Given this hierarchy the location of death can be extracted:

diedInRegion|<DEATH>.+?</DEATH>|
<PLACE.+?>([\w ,']+?)</PLACE>

Again, the line is divided up at the pipe characters, this time producing the predicate *diedInRegion* and two separate regular expressions. The first regular expression grabs the prose associated specifically with the death of the target of the entry and the second pulls out all the regions returned by the first expression. This recursive approach allows for the extraction to be carried out to many levels of precision, although in practice it has not been necessary to resort to more than three.

IV. Limitations of the Method

While the tool is effective overall, producing hundreds of thousands of triples with less than 80 lines of targeted regular expressions and about 100 lines of Python, it does suffer drawbacks. Principally these are an inability to target certain relationships and promiscuous captures.

Some relationships are simply unavailable for capture. The exact nature of various relationships between people within entries is a case in point. Given the markup, it is simply not possible to determine whether people named within the same section of an entry are related to each other apart from their relationship to the subject of the entry. Currently we deal with this by ignoring such possible connections. We are exploring the use of NER and relation extractors such as SONEX [7] to overcome this limitation in the long term.

The reliance of XML on a hierarchical structure rather than explicit relationship labels means that inferring relationships based on the location of proper names within context tags can lead to some curious results. In one entry, William Shakespeare is named in an *intimateRelationship* tag related to a woman in love with a Shakespearean actress, which results in the production of triples for both Shakespeare and the actual love object.

V. Provenance

Given the limitations of the extraction process it is important to make it clear within the RDF that each triple was produced by an automated extraction process. While this could be done via meta-level tags in the file that holds the RDF, this crucial fact would then be hidden from queries and inference agents. To avoid this, each triple is reified by assigning it a URI and then connecting its pieces with the predicates *rdf:object*, *rdf:predicate*, and *rdf:subject*.

VI. Conclusion

The digital humanities are turning to Linked Data in search of discoverability, interoperability, and scalability. However, such conversion is not a straightforward process, typically because the information that is valuable to humanists is not obviously compatible with XML hierarchies or simple RDF.

Acknowledgment

The authors would like to thank Jentery Sayers, Jon Bath, Adèle Barclay and other members of the *INKE* Modeling & Prototyping team. Members of *Orlando*, including editors Patricia Clements and Isobel Grundy were key to the creation of *Orlando* along with many research assistants, postdocs, and staff. Jeffery Antoniuk and other members of the *Canadian Writing Research Collaboratory* project (cwrc.ca) are building the infrastructure for our RDF. This project is supported by the *Text Mining and Visualization Project for Literary History* project, which is funded by the Social Sciences and Humanities Research Council of Canada, which also funds *INKE* (inke.ca).

References

[1] T. Blanke, G. Bodard, M. Bryant, S. Dunn, and M. Hedges, "Linked data for humanities researchThe SPQR experiment." in *6th IEEE International Conference on Digital Ecosystems Technologies (DEST)*, 2012 ©IEEE. DOI: 10.1109/DEST.2012.6227932

[2] S. Brown, P. Clements, and I. Grundy, eds., *Orlando: Women's Writing in the British Isles from the Beginnings to the Present*. Cambridge: Cambridge University Press, 2006-2013. http://orlando.cambridge.org

[3] S. Brown, S. Fisher, P. Clements, K. Binhammer, T. Butler, K. Carter, I. Grundy, and S. Hockey, "SGML and the Orlando Project: descriptive markup for an electronic history of women's writing," *Computers and the Humanities*, vol. 31, no. 4, pp. 271-85, 1997.

[4] M. H. Butler, J. Gilbert, A. Seaborne, K. Smathers, "Data conversion, extraction and record linkage using XML and RDF tools in Project SIMILE," HP Labs, Bristol, UK, 2004.

[5] J. Hunter, T. Cole, R. Sanderson, H. Van de Sompel, "The open annotation collaboration: A data model to support sharing and interoperability of scholarly annotations," in *Digital Humanities 2010*, London, United Kingdom, 2010, pp. 175-178.

[6] M. Konnikova, "Humanities aren't a science. Stop treating them like one," in *Literally Psyched: Scientific American Blog Network*, retrieved 13 January 2013. http://blogs.scientificamerican.com/literally-psyched/2012/08/10/humanities-arent-a-science-stop-treating-them-like-one/

[7] SONEX. https://sites.google.com/a/ualberta.ca/sonex/

[8] D. Van Deursen, C. Poppe, G. Martens, E. Mannens, and R. Walle, "XML to RDF conversion: a generic approach," in *International Conference on Automated solutions for Cross Media Content and Multi-channel Distribution (AXMEDIS'08)*, 2008 ©IEEE. DOI: 10.1109/AXMEDIS.2008.17

[9] W3C. ConverterToRdf. http://www.w3.org/wiki/ConverterToRdf