**SPERBERG-McQUEEN — TLRR2**

<1 image>

**TLRR2: A Digital Revision of Trials in the Late Roman Republic: 149 BC to 50 BC**

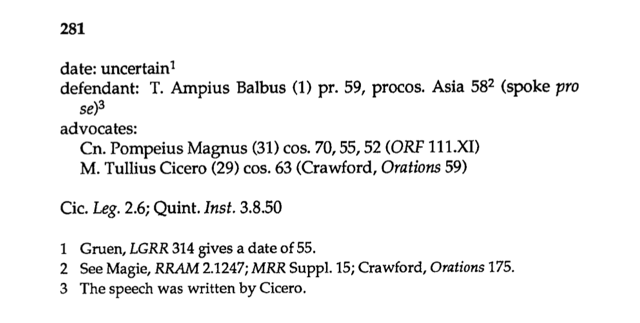
Sperberg-McQueen, M.

Trials in the Late Roman Republic: 149 BC to 50 BC (Alexander, 1990) is a tabulation of the known legal facts pertaining to trials and possible trials, criminal and civil, during the last century of the Roman republic. The first edition was compiled by Michael C. Alexander and published in 1990; a second edition is in preparation by a team of collaborators under Alexander’s direction. This short paper reports on the technical work involved in representing the data in XML, supporting distributed editing of the database, and presenting the results on the Web. (The TLRR2 project is getting under way as this proposal is submitted and will not be completed until 2016; the paper will thus be a report on work in progress, and the project will benefit from comments on the paper.)

### Data Format and XML Representation

TLRR lists information in a consistent format for each of the 391 known trials and possible trials: date; formal charge or claim (with a description of the substantive offense, when known); defendant(s); advocates for defendant and plaintiff; prosecutor(s) or plaintiff(s); presiding magistrate; jurors; witnesses, informers, character witnesses; other individuals involved in the trial; verdict. For each trial, the ancient sources of information are listed, together with modern secondary literature relevant to the formal aspects of the trial. The material thus lends itself well to a database management system. However, it exhibits some properties that can be cumbersome to handle with conventional DBMS. The information available for each trial varies widely: each piece of information may be missing or doubtful for a given trial. Dates may be unknown but bounded by a *terminus a quo* or a *terminus ad quem*; many pieces of information require annotation explaining the inferences that led to them and referring to the relevant secondary literature for support or alternative views. And, of course, there may be one defendant or many, one plaintiff or many, and so on: the data are not naturally in third normal form.

The same properties that make the data challenging for conventional database management systems make it reasonably well suited for XML. So the first technical task of the current effort has been to develop a suitable XML vocabulary. A sample record appears thus in the first edition:



Many XML representations of this are possible. The first edition was typeset from electronic documents which, translated into XML, look something like this for this sample record:

<trial id="ZLK">

<?WScript .sr ZLK = &chapter?>

<?WScript .hi +2?>

date: uncertain

<en>Gruen, <hp1>LGRR</hp1>

314 gives a date of 55.

</en>

<br/>

<ix n="2" target="ZLK"

>Ampius (++1), T. Balbus</ix>

defendant: T. Ampius Balbus (1) pr. 59,

procos. Asia 58

<en>See Magie, <hp1>RRAM</hp1>

2.1247; <hp1>MRR</hp1> Suppl. 15;

Crawford, <hp1>Orations</hp1> 175.

</en>

(spoke

<hp1>pro se</hp1>)

<en>The speech was written by Cicero.

</en>

<br/>

<ix n="3" target="ZLK"

>Pompeius (+31), Cn. Magnus</ix>

<ix n="3" target="ZLK"

>Tullius (+29), M. Cicero</ix>

advocates:

<?WScript .in +2?>

Cn. Pompeius Magnus (31) cos. 70, 55, 52

(<hp1>ORF</hp1> 111.XI)

<br/>

M. Tullius Cicero (29) cos. 63

(Crawford, <hp1>Orations</hp1> 59)

<?WScript .in?>

<?WScript .sk 1?>

<p>

<?WScript .hi off?>

Cic.

<hp1>Leg.</hp1>

2.6; Quint. <hp1>Inst.</hp1> 3.8.50

<?WScript .sk?>

</p></trial>

An XML representation less oriented toward typography can dispense with the forced line breaks and with the indexing instructions (since indices can be built from the data):

<trial tlrr1="281" sortdate="55">

<date>uncertain<fn>

<p><bibref bibl="Gruen-LGRR"

detail="314"

>Gruen, LGRR 314</bibref>

gives a date of 55.</p>

</fn></date>

<defendants>

<defendant>

<personref person="T-Ampius-Balbus--1"

>T. Ampius Balbus (1)

pr. 59,

procos. Asia 58</personref><fn>

<p>See

<bibref bibl="Magie-RRAM"

detail="2.1247"

>Magie, RRAM 2.1247</bibref>;

<bibref bibl="MRR-Suppl-15"

>MRR Suppl. 15</bibref>;

<bibref bibl="Crawford-Orations"

detail="175">Crawford,

Orations 175</bibref>.</p>

</fn>

(spoke pro se)<fn>

<p>

The speech was written by Cicero.

</p>

</fn></defendant>

</defendants>

<advocates>

<advocate>

<personref

person="Cn-Pompeius-Magnus--31"

>Cn. Pompeius Magnus (31)

cos. 70, 55, 52

(<srcref src="ORF"

detail="111.XI"

>ORF 111.XI</srcref>)</personref>

</advocate>

<advocate>

<personref

person="M-Tullius-Cicero--29"

>M. Tullius Cicero (29) cos. 63

(<bibref

bibl="Crawford-Orations"

detail="59"

>Crawford, Orations 59</bibref>

</personref>

)</advocate>

</advocates>

<sources>

<ancient>

<srcref src="Cic.Leg." detail="2.6"

>Cic. Leg. 2.6</srcref>;

<srcref src="Quint.Inst."

detail="3.8.50"

>Quint. Inst. 3.8.50</srcref>

</ancient>

</sources>

</trial>

Three problems interact here: the design of an XML vocabulary to provide good search and retrieval interfaces, the creation of an XSLT stylesheet to translate from the existing format into the new target format, and the creation of an editing interface that can allow the editors to update the data correctly and successfully. The first two need no further discussion here; the third is elaborated below.

### Implementation Strategies for Distributed XML Editing

The major practical challenge for the technical support of the TLRR2 project is that the project team is geographically distributed. The obvious solution is to provide a web interface to the database, to allow the co-authors to work independently of each other in their Web browsers.

The TLRR project involves work with a small number of distinct entity types: trials, persons, laws or crimes, ancient sources, modern sources. A standard implementation approach in such cases is to create distinct electronic objects for each entity (here: XML documents), and use inter-entity pointers (hyperlinks) to express relations among entities (Richardson and Ruby, 2007). An implementation must then provide mechanisms for creating, retrieving, updating, and deleting the XML documents representing the different entities.

The choice of implementation platform for this distributed editing interface proved challenging enough that a comparative evaluation of different platforms has become a subsidiary project in its own right. Instead of choosing an implementation strategy as best we can, and then living with it, we plan to implement our distributed editing system several times, with different tools, in order to be able to report realistically on comparative advantages and disadvantages of each. Among the approaches to be considered are

• A shoestring approach using an Apache server with a Subversion back end to accept PUT requests from XForms running in the client browser (using XSLTForms as the XForms implementation). The absence of an XQuery server will make some aspects of this approach very clumsy, but it can be implemented on a very simple, low-cost hosting provider. Providing a good user interface for editing free prose (e.g., in footnotes) will be a major challenge in this approach.

• Implementation using the tools of any one of several open-source and/or commercial XQuery or XForms servers: Orbeon, eXist, BaseX, MarkLogic. (28msec has re-branded itself as a JSON engine, but its infrastructure is reported still to support XML and XQuery, so it is also a possibility here.)

• Using the Author Mode applet of Oxygen, the widely used XML editor and interactive development environment (IDE) (Bina, 2013).

• Deploying any of several browser-based XML editors, e.g., FontoXML, XOpus (on which see O’Connor et al., 2013, or others).

• Using an SQL database for the data. As noted above, there are aspects of the data that make this likely to be cumbersome, but since SQL support is much more widespread among commercial web hosting providers than support for XQuery servers, it may be worth seeing just how high a price the use of non-XML-aware SQL would exact from an implementation.

Elements of these different approaches can, of course, be combined to create still more implementation approaches.

The implementation experiences of the TLRR2 project should be relevant to the general problems of providing suitable user interfaces for the editing of XML resources by non-XML specialists, and to the general problem of using computers for historical data without allowing information technology to push us into oversimplifying our data.

### Presentation and Search Interface, Links to Other Data

Both during and after the process of revision that will lead to TLRR2, it will be necessary to support useful search and retrieval operations on the data. Once the project is complete, some search and retrieval techniques may be feasible on the static data that would not have been feasible on the updatable data. It is not expected that the development of styles to display and search interfaces to search the data will present particularly difficult challenges.

Any historical project like TLRR may benefit from linking to, and being linked to from, other projects with overlapping scopes. Specialists in Roman legal history may feel no pressing need to have references to ancient sources like ‘Quint. *Inst.* 3.8.50’ hyperlinked to networked versions of Quintilian's text; they already know where to find the text when they need it. But others may find such hyperlinks helpful. And providing standardized interfaces to the information content of a project like TLRR2 can allow the information to be regarded in other contexts. For that reason, the TLRR website will both use interfaces like the Canonical Text Services (CTS) interface (Smith, 2009; Berti et al., 2014) to link out to other resources, and make its information accessible to other sites and projects using standard methods of exposing prosopographic and other data, e.g., as described by the SNAP project (Bodard et al., n.d.; Cayless, 2014).

### References

**Alexander, M. C.** (1990). Trials in the Late Roman Republic, 149 BC to 50 BC. Phoenix. Supplementary Volume XXVI / Tome supplementaire XXVI. University of Toronto Press, Toronto.

**Berti, M., et al.** (2014). A New Implementation for Canonical Text Services. In Proceedings of the 8th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH) @ EACL 2014, Gothenburg, Sweden, 26 April 2014, pp. 1–8.

**Bina, G.** (2013). Customizing a General Purpose XML Editor: oXygen’s Authoring Environment. In Proceedings of the International Symposium on Native XML User Interfaces, Montréal, Canada, 5 August 2013. Balisage Series on Markup Technologies, vol. 11, doi:10.4242/BalisageVol11.Bina01.

**Bodard, G., et al.** (n.d.). Standards for Networking Ancient Prosopographies: The SNAP Project. http://snapdrgn.net/wp-content/uploads/2014/02/SNAPall.pdf.

**Cayless, H.** (2014). SNAP Persons, 10 October, http://snapdrgn.net/archives/302.

**O’Connor, C., Gnanapiragasam, A. and Hepp, M.** (2013). ProofExpress: An Online, Browser-Based XML Article Proofing System for STM Journals. In Proceedings of the International Symposium on Native XML User Interfaces, Montréal, Canada, 5 August 2013. Balisage Series on Markup Technologies, vol. 11, doi:10.4242/BalisageVol11.OConnor01.

**Richardson, L. and Ruby, S.** (2007). RESTful Web Services. O’Reilly, Sebastopol.

**Smith, D. N.** (2009). Citation in Classical Studies. Digital Humanities Quarterly, **3**(1).

**W3C** (World Wide Web Consortium). (2008/1998). In Bray, T., et al. (eds), Extensible Markup Language (XML) 1.0, 5th ed., W3C Recommendation, 26 November 2008. Cambridge: World Wide Web Consortium.