

Curating a nine year old journal FAIR data table.

Henry Rzepa 

Published May 29, 2017

Citation

Rzepa, H. (2017). Curating a nine year old journal FAIR data table. *Henry Rzepa's Blog*. <https://doi.org/10.59350/z9g5j-r2p69>

Keywords

Chemical IT, Applet, Compression Algorithm, Computing, Cross-platform Software

**Certificate has been revoked.
The application will not be executed.**



Name: jmolApplet0

Location: <http://www.rsc.org>

OK

More Information...

Copyright

Copyright © Henry Rzepa 2017. Distributed under the terms of the [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Henry Rzepa's Blog

As the Internet and its Web-components age, so early pages start to decay as technology moves on. [A few posts ago](#), I talked about the maintenance of a relatively simple page first hosted some 21 years ago. In my notes on the curation, I wrote the phrase “*Less successful was the attempt to include buttons which could be used to annotate the structures with highlights. These buttons no longer work and will have to be entirely replaced in the future at some stage.*” Well, that time has now come, for a rather more crucial page associated with a journal article published more recently in 2009.[cite]10.1039/b810301a[/cite]

The story started a few days ago when I was contacted by the learned society publisher of that article, noting they were “*just checking our updated HTML view and wanted to test some of our old exceptions*”. I should perhaps explain what this refers to. The standard journal production procedures involve receiving a Word document from authors and turning that into XML markup for the internal production processes. For some years now, I have found such passive (*i.e.* printable only) Word content unsatisfactory for expressing what is now called FAIR (Findable, accessible, inter-operable and re-usable) data. Instead, I would create another XML expression (using HTML), which I described as **Interactive Tables** and then ask the publisher to host it and add that as a further link to the final published article. I have found that learned society publishers have not been unwilling to create an “exception” to their standard production workflows (the purely commercial publishers rather less so!). That exceptional link is <http://www.rsc.org/suppdata/cp/b8/b810301a/Table/Table1.html> but it has now “*fallen foul of the java deprecation*”.

Back in 2008 when the table was first created, I used the Java-based Jmol program to add the interactive component. That page, when loaded, now responds with the message:

**Certificate has been revoked.
The application will not be executed.**



Name: jmolApplet0

Location: <http://www.rsc.org>

OK

More Information...

This I must emphasise is nothing to do with the publisher, it is the Jmol certificate that has been revoked. That of itself requires explanation. Java is a powerful language which needs to be “sandboxed” to ensure system safety. But commands can be created which can access local file stores and write files out there (including potentially dangerous ones). So it started to become the practise to sign the Java code with the developer certificate to ensure provenance for the code. These certificates are time-expired and around 2015 the time came to renew it. Normally, when such a certificate is renewed, the old one is allowed to continue operation. On this occasion the agency renewing the certificate did not do this but revoked the old one instead


Henry Rzepa's Blog

(Certificate has been revoked, reason: CESSATION_OF_OPERATION, revocation date: Thu Oct 15 23:11:18 BST 2015). So all instances of Jmol with the old certificate now give the above error message.

The solution in this case is easy; the old Jmol code (as **JmolAppletSigned.jar**) is simply replaced with the new version for which the certificate is again valid. But simply doing that alone would merely have postponed the problem; Java is now indeed deprecated for many publishers, which is a warning that it will be prohibited at some stage in the future. ^‡ ^So time to bite the bullet and remove the dependency on Java-Jmol, replacing it with JSmol which uses only JavaScript.

Changing published content is in general not allowed; one instead must publish a corrigendum. But in this instance, it is not the content that needs changing but the style of its presentation (following the principle of the Web of a clear-cut separation of style and content). So I set out to update the style of presentation, but I was keen to document the procedures used. I did this by commenting out non-functional parts of the style components of my original HTML document (as <!-- comment -->) and adding new ones. I describe the changes I made below.

1. The old HTML contained the following initialisation code: `jmolInitialize(".", "JmolAppletSigned.jar"); jmolSetLogLevel('0');` which was commented out.
2. New scripts to initialize instead JSmol were added, such as:
`<script src="JSmol.min.js" type="text/javascript"> </script>`
3. I added further scripts to set up controls to add interactivity.



3D: [Show](#)◆[Hide](#). Spin Vibrate anti-alias [Print](#)◆[Help](#)

4. The now deprecated buttons had been invoked using a Jmol instance: `jmolButton('load "7-c2-h-020.jvxl"; isosurface "" opaque; zoom 120;', "rho(r) H")`
5. which was replaced by the JSmol equivalent, but this time to produce a hyperlink rather than a button (to allow the greek ρ to appear, which it could not on a button): ` $\rho(r)$ `,
6. Some more changes were made to another component of the table, the links to the data repository. Originally, these quoted a form of persistent identifier known as a Handle; [10042/to-800](#). Since the data was deposited in 2008, the data repository has licensed further functionality to add DataCite DOIs to each entry. For this entry, [10.14469/ch/775](#). Why? Well, the original Handle registration had very little (chemically) useful registered metadata, whereas DataCite allows far richer content. So an extra column was added to the table to indicate these alternate identifiers for the data.
7. We are now at the stage of preparing to replace the Java applet at the publishers site with the Javascript version, along with the amended HTML file. The above link, as I write this post, still invokes the old Java, but hopefully it will shortly change to function again as a fully interactive table.

Henry Rzepa's Blog

8. I should say that the whole process, including finding a solution and implementing it took 3-4 hours work, of which the major part was the analysis rather than its implementation.

It might be interesting to speculate how long the curated table will last before it too needs further curation. There are some specifics in the files which might be a cause for worry, namely the so-called JVXL isosurfaces which are displayed. These are currently only supported by Jmol/JSmol. They were originally deployed because iso-surfaces tend to be quite large datafiles and JVXL used a remarkably efficient compression algorithm ("marching cubes") which reduces the cube size one hundred-fold or more. Should JSmol itself become non-operational at some time in the (hopefully) far future (which we take to be ~10 years!) then a replacement for the display of JVXL will need to be found. But the chances are that the table itself will decay "gracefully", with the HTML components likely to outlive most of the other features. The data repository quoted above has itself now been available for ~12 years and it too is expected to survive in some form for perhaps another 10. Beyond that period, no-one really knows what will still remain.

You may well ask why the traditional journal model of using paper to print articles and which has survived some 350 years now, is being replaced by one which struggles to survive 10 years without expensive curation. Obviously, a 3D interactive display is not possible on paper. [cite]10.6084/m9.figshare.797481[/cite] But one also hears that publishers are increasingly dropping printed versions entirely. One presumes that the XML content will be assiduously preserved, but re-working (transforming, as in XSLT) any particular flavour of XML into another publishers systems is also likely to be expensive. Perhaps in the future the preservation of 100% of all currently published journals will indeed become too expensive and we might see some of the less important ones vanishing for ever?†

‡Nowadays it is necessary to configure your system or Web browser to allow even signed valid Java applets to operate. Thus in the Safari browser (which still allows Java to operate, other popular browsers such as Chrome and Firefox have recently removed this ability), one has to go to **preferences/security/plugin-settings/Java**, enter the URL of the site hosting the applet and set it to either "ask" (when a prompt will always appear asking if you want to accept the applet) or "on" when it will always do so. How much longer this option will remain in this browser is uncertain.

†In the area of chemistry, an early pioneer was the *Internet Journal of Chemistry*, where the presentation of the content took full advantage of Web-technologies and was on-line only. It no longer operates and the articles it hosted are gone.

<https://orcid.org/0000-0002-8635-8390>