PubMed Central

XML-based archive of life sciences literature at the NLM

Jeff **Beck**

Abstract

PubMed Central is the National Library of Medicine's XML-based archive of free full-text journal literature. Started in 2000 to increase public access to journal literature, the project now receives content submitted by publishers, back issue content that is scanned with XML headers, and articles funded by the National Institutes of Health submitted through the NIH Public Access Policy.

Table of Contents

1. Early PMC 3
2. The History of PMC 4
3. PMC as an Archive 5
4. Issues in Content Conversion 6
    4.1. Evaluation 7
    4.2. PMC QA 7
    4.3. Regression Testing of Converters 7
5. Special Characters 7
Bibliography 8

PubMed Central (PMC; http://www.pubmedcentral.gov) is the National Library of Medicine's digital archive of free full-text journal literature. Traditionally, journals deposit material in PMC on a voluntary basis. Articles may be retrieved either by browsing a table of contents for a specific journal or by searching the database. The content in PMC is always free, although there may be a time lag of a few weeks to a year or more between publication of a journal issue and when it is available in PMC.

To increase the functionality of the database, a variety of links are added to the articles in PMC: between an article correction and the original article; from an article to other articles in PMC that cite it; from a citation in the references section to the corresponding abstract in PubMed and to its full text in PMC; and from an article to related records in other Entrez databases such as Reference Sequences, OMIM, PubChem, and Books

As of September, 2005, PubMed Central includes over 600,000 articles from more than 300 Journals. Participating journals range from small new journals like Evidence-based Complementary and Alternative Medicine (http://web.pubmedcentral.nih.gov/tocrender.fcgi?action=archive&journal=241) to standards like Proceedings of the National Academy of Sciences of the USA (PNAS; http://web.pubmedcentral.nih.gov/tocrender.fcgi?action=archive&journal=2), the Journal of Clinical Investigation and the journals of the American Society for Microbiology.

PubMed Central was started in 2000 as a project for the National Center for Biotechnology Information (NCBI), a center in the National Library of Medicine (NLM). NCBI was established in 1988 as a national resource for molecular biology information, NCBI creates public databases, conducts research in computational biology, develops software tools for analyzing genome data, and disseminates biomedical information for the better understanding of molecular processes affecting human health and disease.

By 2000, PubMed (http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=PubMed), a database of abstracts that includes over 15 million citations from MEDLINE and other life science journals for biomedical articles back to the 1950s, was already well established. PMC was created to allow and encourage free access to the full-text of articles from life sciences journals.

Because the early goal of PubMed Central was free access to the journal literature – to as large an audience possible worldwide, there were two presentation guidelines that have a strong effect on the operation of the PMC site:

1. PMC must run on slow, old machines with slow internet connections.

2. PMC must run without plugins.

We'll see the effects of these below.

1. Early PMC

There were a handful of early participants: The American Society for Cell Biology with Molecular Biology of the Cell (MBC), The National Academy of Sciences with PNAS, The BMJ with bmj.com, Nucleic Acids Research, and BioMed Central with several of their early journals (Breast Cancer Research, Genome Biology, and Arthritis Research).

The content for all of these early participants (except for Nucleic Acids Research) came in two forms:

1. SGML in the keton.dtd (MBC, PNAS, BMJ).

2. XML in the article.dtd from BioMed Central.

The articles were loaded into a PMC Database in their native form (see Figure 1, "Early PMC Workflow"). When an article (or collection of articles, like a Table of Contents) was requested by a user, the SGML or XML was retrieved from the database and converted to HTML for display on the PMC Website.

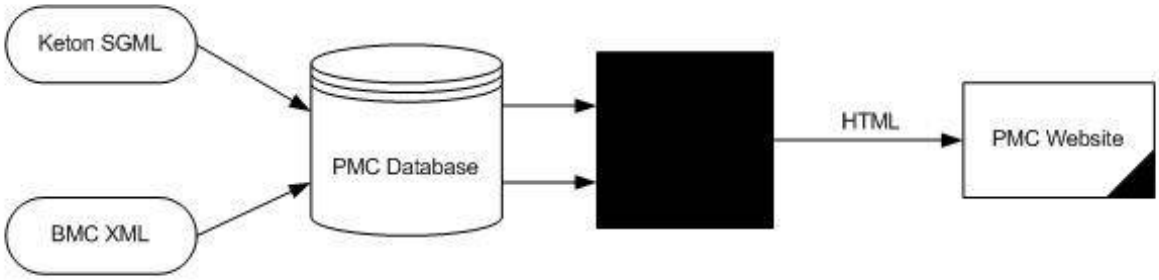

Figure 1. Early PMC Workflow

In an ideal world, with a system like this you will need a piece of converter code for each source schema. In reality these rendering converters needed to be written for each title – or even for ranges of issues within a journal.

Under this scheme, the rendering converters were being written and maintained by the programmers who had developed the software. Variations in the source were rejected as “Bad Data”, sometimes even when they were normal expected variations in publishing.

For example, the software wanted to generate the symbols on author names (to affiliations and/or author-related footnotes) from a standard list. One of the early selling points of PMC was that journals could maintain much of their style. This meant that they wanted to see the author symbols that they were used to seeing in their print product. It wound up that these symbols were encoded in the ID/RID for the footnote and affiliation elements and references, but the PMC rendering software was not able to accommodate these.

PMC went live in 2000 with a handful of issues, but it soon became obvious that this system was not nearly flexible enough or scaleable to be able to handle any real growth.

2. The History of PMC

In late 2000, there was a decision to move to the workflow detailed in Figure 2, “Modified PMC Workflow”. All incoming content would be transformed to a common format before it was loaded to the PMC Database. Once there, when it was requested, it would be rendered to HTML through one rendering converter that expects only one format of XML.

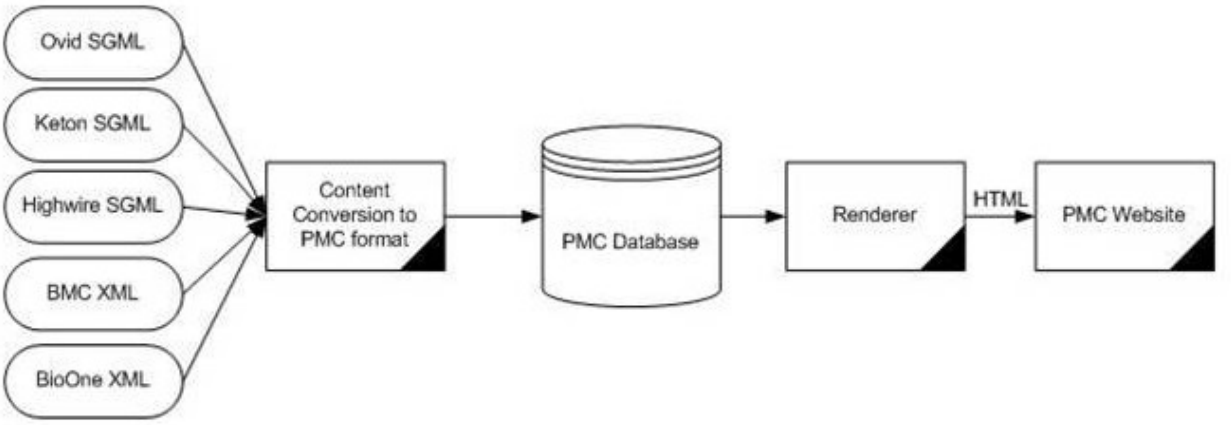

Figure 2. Modified PMC Workflow

We created the pmc-1.dtd – still with the main goal of online access. It was based on the keton.dtd and the BMC article.dtd.

The pmc-1.dtd is still available online (http://www.pubmedcentral.nih.gov/pmcdoc/dtd/pmc-1.dtd), but it is no longer being used for any content.

This change addressed a number of challenges from the first PMC system. First, loading a single format into the database could simplify the database and the software used to load content into it. Now it only had to be aware of one DTD.

The rendering software could also be simplified greatly from this shift. There would still be differences in the XML and the output style between journals, but these differences were minor compared with the conversion and rendering that this step was expected to accomplish in early PMC.

It may seem like the hard work has just been pushed upstream (from post-DB to pre-DB), but there was another significant change at this point.

We started building the mappings from the keton,dtd and BMC article.dtd to the new pmc-1.dtd, and it was decided that it would be easier for the staff with editorial and content background to write and run the content conversions.

Essentially it was easier for the editorial staff to learn XSLT than for the programming staff to need to make decisions based on the article content (as they had done in the first PMC).

This shift was critical to the transition. Although it took a while to learn XSLT, once there was a basic understanding, the converters could be created and updated by one person. There was no need to translate the editorial rules and communicate them to a programmer who was not familiar with the possible idiosyncrasies of journal article content.

3. PMC as an Archive

In 2001, The National Library of Medicine named PubMed Central as its “free digital archive of biomedical and life sciences journal literature.” This was indicative of a shift in PMC from a primary focus on Access to a primary focus on preserving the content.

Also at this time, it was becoming more and more difficult to convert new content from different source SGML and XML DTDs into the access-optimized pmc-1.dtd. We created the NLM Archiving and Interchange XML Tagset and associated DTDs[R1] based in part on recommendations that came out of the Harvard’s E-Journal Archive DTD Feasibility Study [R2]

But there is more than SGML or XML to making up the PMC archive. Figure 3, “PMC Archiving Workflow” shows the workflow for different types of content coming in to PMC.

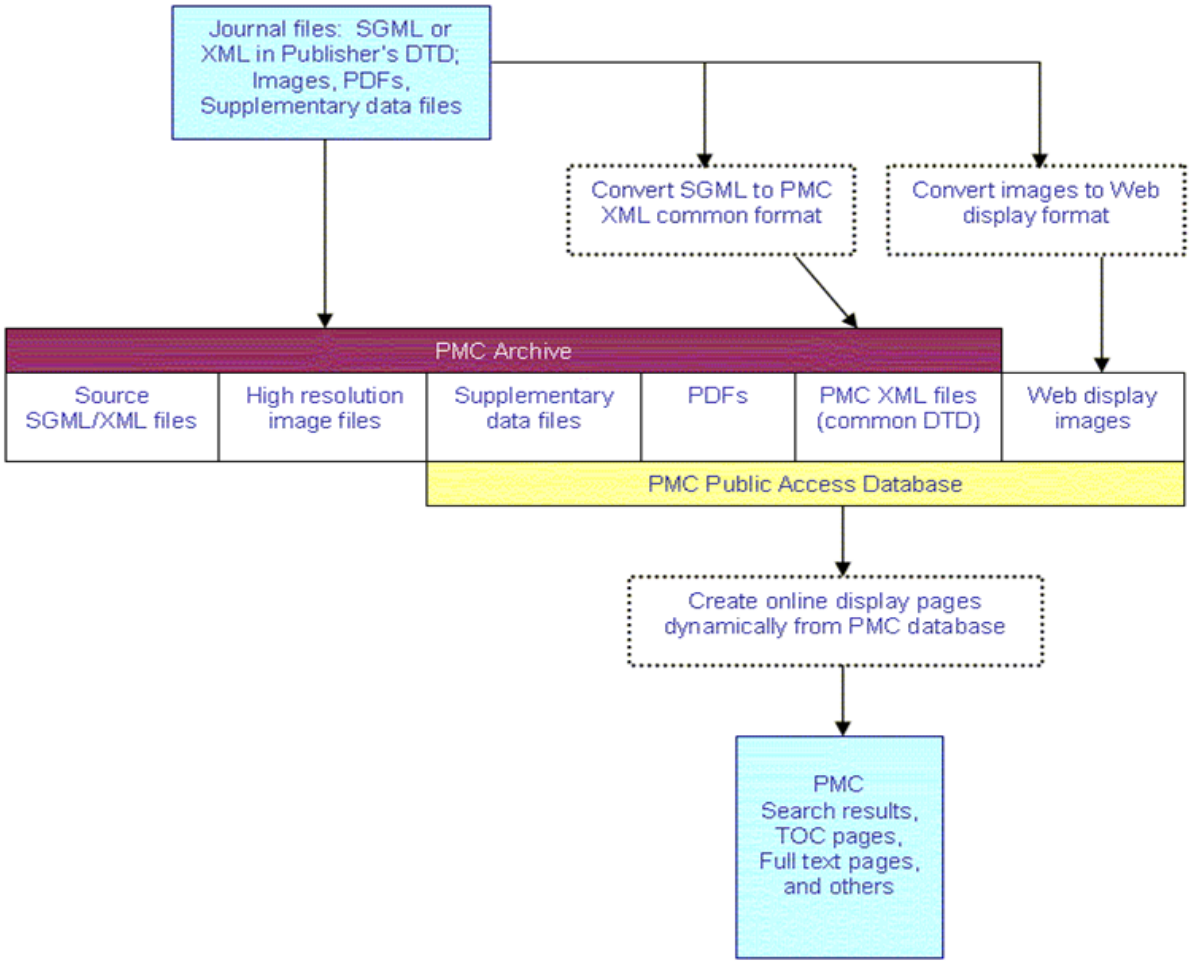

Figure 3. PMC Archiving Workflow

The red bar represents PMC’s archiving function. The text, in SGML or XML is saved in the archive. The version of the article that has been converted to the NLM Archiving DTD is also saved. The source high resolution images, any supplied PDF files, and supplementary data files are also stored.

The yellow bar represents the public side of PMC. The only type of file available to the public that is not stored in the archive are the web-optimized images.

4. Issues in Content Conversion

In Figure 1, “Early PMC Workflow”, there is a black box that represents that conversion/rendering process that was completely under the control of the programmers who wrote that software. When the conversion step was moved up-stream, we gained a lot of efficiency by putting the content conversion into the hands of content-minded people, but some of the same issues that we faced in early PMC now face us in the content conversion step. The major challenges we see here are:

1. Inconsistent use of source models. Two journals that are supplied in the same DTD from the same vendor may use elements differently. This leads to journal-specific tests in the converters.

2. Each journal requires analysis over time. As with journals from the same supplier that are tagged differently, the tagging of the source will change over time. Sometimes there will be an anomalous issue or two (I always figured that the regular SGML tagger was on vacation), but it could be a shift in how the supplier was applying their DTD. Changes in the print style of the journal also tend to have a great effect on the way the SGML is tagged.

3. SGML is created from print files. Most of the SGML or XML that is being created now is still be made after the print product is complete. The files that were sent to the press (usually PDF or an ASCII output from the print system) are sent to SGML taggers for conversion to SGML.

   Most of the time, this is a semi-automated process. We still see evidence of cut-and-paste in SGML files that we receive for PMC. Even without these blatant errors, the quality of the content in the SMGL files is questionable because the electronic product has not been through the same scrutiny as the print product.

4. Sometimes the content in the SGML is just wrong.

For PMC, each journal is required to supply SGML or XML either in the NLM Journal Publishing DTD or in an “approved” DTD that can be converted to our target. We have established some procedures to help work around these issues.

4.1. Evaluation

Before being accepted into PMC, new journals submit three complete issues for evaluation. The content is first validated against its DTD or Schema, and then the content is checked against the copy of record - either print or online. The content is also checked to be sure that required pieces (not necessarily elements) are supplied. These would include items like publication date(s), journal information, and article citation information.

4.2. PMC QA

All content goes through some Quality Assurance step before it is made live in PMC. Newer journals get a harder look than those with a proven track record, but certain things like Tables of Contents and Math are always checked.

Any incorrect data that is found in the QA stage is returned to the supplier to be fixed.

4.3. Regression Testing of Converters

With converters at the source DTD level, one converter can be responsible for translating the content of many different journals. To be sure that minor changes in the XSL transform for one journal don't break the XML transform of another title, we built a regression testing system based on a similar system at Inera[R3]

5. Special Characters

In PMC now, we use Unicode value-based character entities in the XML that we load into the database.

There are two areas where special characters have been challenging: characters in the submitted SGML or XML and how to render characters on the PMC website.

All named characters (beyond the usual popular ISO Charsets for publishing) must be mapped to their corresponding Unicode value(s) before any content can be converted. For the most part this is just work and not too challenging, but making the leap from the SGML world to the XML world can be tough. We use the combining accents in PMC. When we see a character like ˆ in SGML, most of the time we will want to convert it to ̂, the combining circumflex accent, but it may just mean "^" in the source.

There are, of course, characters that are named in the source DTDs that cannot be represented by Unicode value(s). These we convert to the <private-char> element in the output XML. <private-char> allows us to include a description of the glyph in the article itself so that the article will be complete.

For rendering, we supply either the Unicode value or a call to a glyph for each character depending on whether a user's browser/operating system combination is able to process the Unicode value for the given character.

Bibliography

[R1] Beck and Lapeyre (2003) New Public Domain Journal Archiving and Interchange DTDs. XML 2003. http://www.idealliance.org/papers/dx_xml03/papers/04-01-02/04-01-02.html.

[R2] Harvard University Library Office for Information Systems E-Journal Archiving Project. E-Journal Archive DTD Feasibility Study (2001) http://www.diglib.org/preserve/hadtdfs.pdf.

[R3] Rosenblum and Golfman (2004) Automated Quality Assurance for Heuristic-Based XML Creation Systems. Extreme Markup Languages 2004. http://www.mulberrytech.com/Extreme/Proceedings/html/2004/Rosenblum01/EML2004Rosenblum01.html

Biography

Jeff **Beck**
Technical Information Specialist
National Center for Biotechnology Information [http://www.ncbi.nlm.nih.gov/]
National Library of Medicine, NIH
8600 Rockville Pike
Bethesda
Maryland
20894
United States of America

Mr. Beck works on the PubMed Central and NLM BookShelf projects at the National Library of Medicine managing the flow of journal and textbook content. He has been working with XML at NLM for 3 years. He has an editorial background, and his previous (relevant) experience includes print production and online production for journals and textbooks.