JATS: An Unexpected Journey

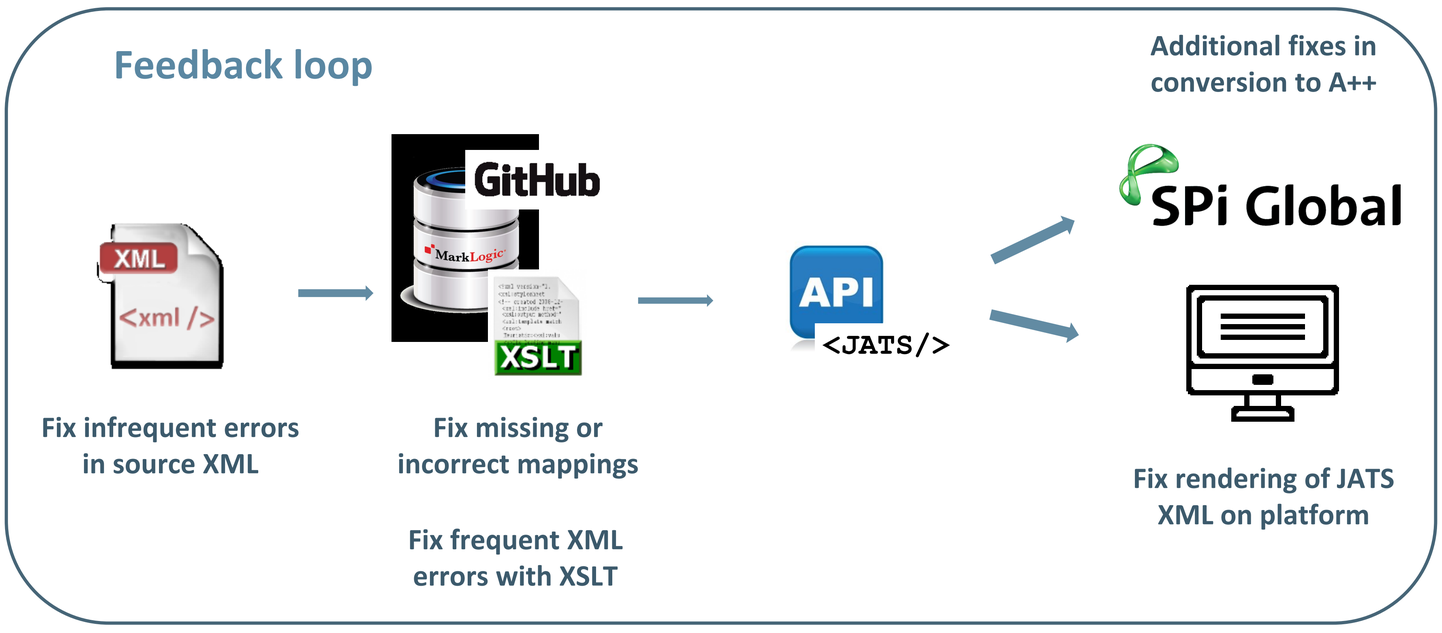
Jenny Sherman

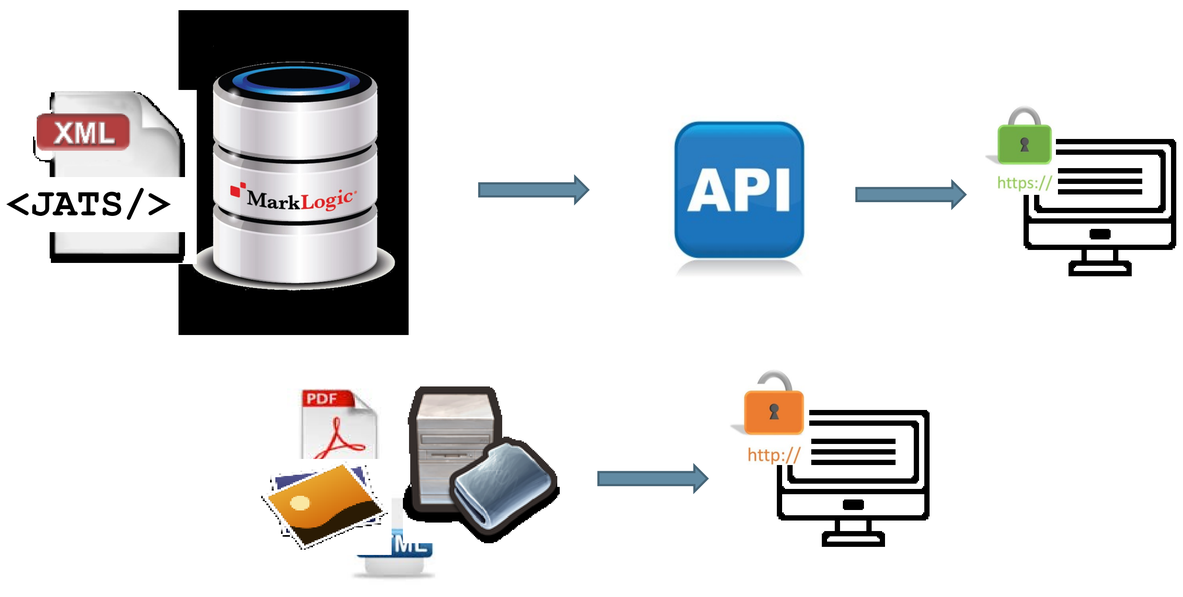
Senior XML Developer, Springer Nature

Abstract

In 2015, we presented our paper[1] on introducing JATS for new articles within the Macmillan publishing group. A few months later we had merged with Springer to become Springer Nature. The Springer side of the business uses a proprietary DTD (A++) and all the highly-automated production systems are based on that DTD. To utilise those same production systems for Macmillan content, we also switched to using A++. However, that's not the end of the story for using JATS within the company. This poster will show how we continued to use JATS in two different ways:

- **as an exchange medium for conversion of the Macmillan archive DTDs to A++, and**
- **for on-the-fly rendering of archive content on our new secure platform.**





The Starting Point

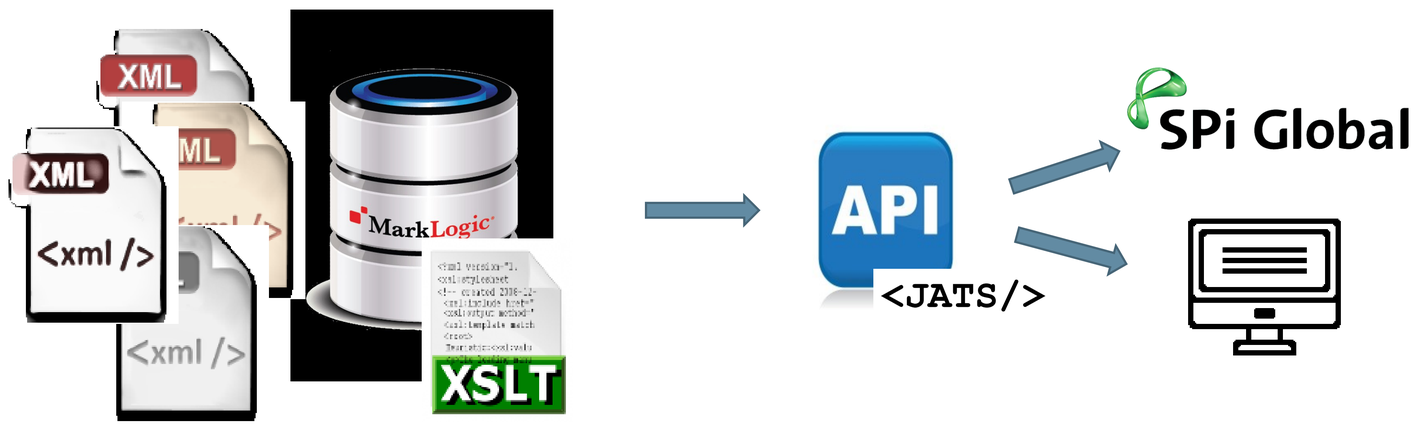
- **New articles already being supplied as JATS XML.**
- **MarkLogic NoSQL Database, where article XML in all four DTDs is stored.**
- **"Asset" API which returned JATS XML from MarkLogic.**
- **"Content Hub" API to return standardized metadata about all articles, regardless of DTD.**
- **Legacy http platform displaying static HTML article pages generated from in-house DTD article XML.**
- **Secure https platform rendering JATS XML articles from asset API and using Content Hub metadata.**
- **JIRA project management system.**
- **Mappings and XSLT developed to convert in-house DTDs to JATS.**
- **GitHub for XSLT version control.**

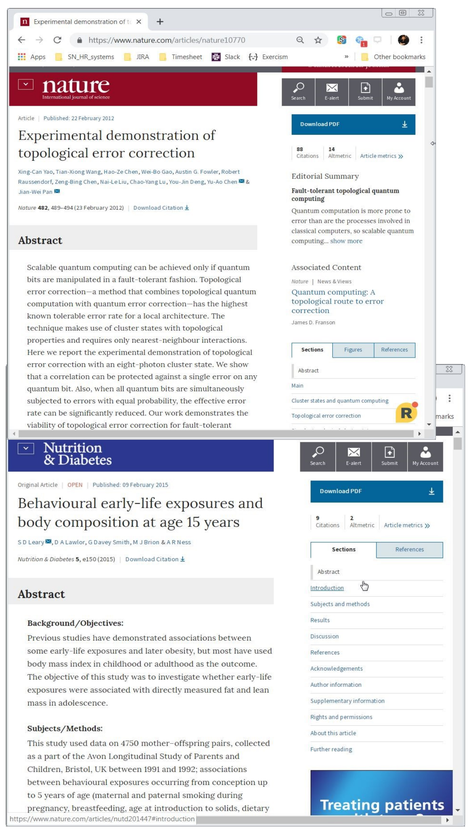
Long-term conversion of archive to A++

- **Completed XSLT to convert three in-house DTDs to JATS.**
- **Continuous integration pipeline created to deploy XSLT to MarkLogic.**
- **Asset API updated to return XML in original format or as JATS equivalent:**

  http://hub.nature.com/assets/v1/xml-assets/nature10770.xml?accept=application/xml

  http://hub.nature.com/assets/v1/xml-assets/nature10770.xml?accept=application/xml**;type=jats**
- **Worked with vendor (SPi) on mapping from JATS to A++.**
- **Feedback from SPi on errors in JATS or A++.**
- **Fix made to XML, XSLT, or JATS to A++ mapping, as appropriate.**





Three months to switch all content to https platform

- **Regular meetings between Product Owner, XML, Content Hub and platform teams to discuss problems and prioritise work.**
- **Platform updated to reference API for all articles, not just native JATS.**
- **Web Production and stakeholders spot-check articles from previous five years and feedback issues in JIRA tickets. Work is done by comparing html version on legacy http platform with JATS version on https platform.**
- **Analysis of "error" to decide at which point of the workflow it is caused – source XML, XSLT, platform rendering.**
- **Errors in source XML – bulk errors fixed with script or, if necessary, within the XSLT; small number of errors fixed manually.**
- **Incorrect or missing mapping – XSLT amended and redeployed to MarkLogic.**
- **Platform rendering – code amended and redeployed.**
- **Verification that fix has worked and JIRA ticket signed off.**

Advantages of the approach

- **No physical conversion to JATS, therefore no reconversion necessary after XML fix or XSLT update – the asset API will automatically return the new version of the article.**
- **Quick feedback on whether a fix has worked in the expected way.**
- **Short project timeframe forced all involved to make pragmatic decisions, which were documented for future reference.**

References

1. The Long Road to JATS (2015)
https://www.ncbi.nlm.nih.gov/books/NBK279831/

Contact

jenny.sherman@springernature.com

SPRINGER NATURE