

EXPANDING YOUR XML & ARBORTEXT DEVELOPMENT TOOLKIT

PTC/USER WORLD EVENT 2009

James Sulak
Developer, Jones McClure Publishing
jsulak@jonesmcclure.com
Blog: www.wordsinboxes.com
Twitter: jsulak

INTRODUCTION

The purpose of this presentation is to introduce Arbortext developers to several emerging XML development technologies – XSLT 2.0, XSpec, and XProc – that I’ve found useful in my day to day work. I will cover each in turn, giving (1) a brief overview of its history and purpose, and (2) a step-through of simple example code. The goal is for you to:

- Understand the purpose of XSLT 2.0, XSpec, and XProc.
- Know where to find additional information about all three technologies.
- Be able to take the first steps towards leveraging these technologies in your own Arbortext development.

Good tools help you get things done. Great tools are a joy. Even if these particular technologies are not a fit for you, I hope this presentation will inspire you to critically examine your development technologies.

ABOUT THE SPEAKER

James Sulak is a developer at Jones McClure Publishing. He converts legacy content to XML; writes and maintains Arbortext customizations; creates new, XML-based editorial processes; and develops stylesheets for multiple publishing platforms.

EXAMPLES

All examples in this presentation (along with this document and the slides) are available from the PTC/User 2009 online portal.

XSLT 2.0

XSLT 2.0 is the successor to XSLT 1.0 and a current W3C Recommendation. It is a more polished and conceptually complete language than XSLT 1.0 that addresses many of that language's shortcomings. Specific advantages include built-in grouping instructions, the ability to define custom functions, regular expressions, new XPath operators and functions, and nodesets that are first-class data types. All of this means that developers can write more concise code that's easier to read and maintain.

MORE INFORMATION

Dave Pawson's XSLT 2.0 FAQ

<http://www.dpawson.co.uk/xsl/rev2/rev2.html>

XSLT 2.0 and XPath 2.0, by Michael Kay

XSPEC

XSpec is a new open-source XSLT unit-testing framework under active development by Jeni Tennison. Testing code with XSpec is faster and more reliable than the usual development cycle of manually and repeatedly examining the output from a transform. In XSpec, test suites are XML documents that contain a series of scenarios describing stylesheet behavior as input / expected output pairs. Running a test suite produces an HTML report that shows passed tests shaded green and failed tests shaded red.

Although XSpec is intended to be used in test-first development, I've found that it is still helpful in a more traditional development cycle. For example, I sometimes create an XSpec test suite after the fact as a "repository of discovered bugs."

Advantages of XSpec (and unit testing in general) include:

- **Faster development.** XSpec tests can be run quickly from the command line or from your XML editor. Constructing tests from client- or employer-provided sample documents is a small initial time investment, but one that pays off, since you don't need to repeatedly hunt through a result document for the specific aspect you're currently focused on.
- **Precision.** Writing tests forces you to be concrete when thinking about the problem.
- **More maintainable code.** An XSpec test suite is a human- and computer-readable document specifying exactly what your XSLT is supposed to do. A complete test suite can help to bring other developers up to speed or refamiliarize you with code after a long absence.
- **Confidence.** If you need to completely refactor your code, you can be confident that you haven't broken anything.

MORE INFORMATION

XSpec Project Page

<http://code.google.com/p/xspec/>

Getting Started with XSpec

<http://www.wordsinboxes.com/2008/12/getting-started-with-xspec.html>

Test XSLT with XSpec by Jeni Tennison

<http://river-valley.tv/test-xslt-with-xspec/>

XPROC

XProc is a new XML pipeline language that is currently a W3C Candidate Recommendation. XProc pipelines define a sequence of actions to be performed on XML documents – for example, XSL transforms, validation, and filtering. The idea of XProc is to make XML manipulation simpler and more reliable by hiding many of the complications that other languages impose. The most complete implementation of XProc is XML Calabash, which is open source and created by Norman Walsh, who is best known for his work on Docbook.

Advantages of XProc over other ways of manipulating XML include:

- **Focus on *what*, not on *how*.** XProc's declarative nature makes simple things simple. You state what you want to happen and let the XProc processor take care of the details.
- **Less complexity = less code, less bugs, and easier maintenance.** In XProc, you work at a high level of abstraction. In contrast, processing XML with most programming languages involves worrying about countless small details – parsing, data marshalling, temporary-file wrangling, etc.
- **Modular development & simpler transforms.** When transforms are so easy to chain together, you can think of them like you do functions in a programming language. Instead of one sprawling transform, you can write multiple smaller, more focused ones, which is easier to debug and maintain.
- **Emerging standard.** Even at this early stage, there are already several implementations available. It has the backing of many major XML players, including MarkLogic, EMC, and the open-source eXist database.

Since XProc and its implementations are so new, you may not want to deploy it in a production environment, although we at Jones McClure have done so successfully. But XProc is coming up quickly and has the potential to become a bedrock XML technology.

MORE INFORMATION

XProc Tutorial

<http://www.xfront.com/xproc/>

Xproc: Step by Step

<https://community.emc.com/docs/DOC-3337>

Getting Started with XProc

<http://www.wordsinboxes.com/search/label/xproc>

An Introduction to XProc: An XML Pipeline Language by Dave Pawson

<http://www.dpawson.co.uk/xproc/>

XProc Specification

<http://www.w3.org/TR/xproc/>

XML Calabash Project Page

<http://xmlcalabash.com/>

XProc-Dev Mailing List Archives

<http://xproc.markmail.org>