The ZapLetter

Monthly Research & Analysis on XML

XML Adventures in Userland

While small in terms of business size, Userland has made big waves in its adoption and advancement of XML standards for content management. In particular, David Winer has made himself well known for evangelizing and developing the XML-RPC standard that was a precursor of the SOAP format for communications between applications running on heterogeneous systems. Userland has pioneered a variety of technologies that have all one thing in common – making it easier for users to create, manage, and share content in a Web environment.

Their goal is embodied in Userland's three main applications: Frontier, Radio, and Manila They are key indicators of Userland's desire to integrate various different XML technologies and standards to meet the goal of rapid content development and integration.

The Frontier application is a development environment whose primary use is for developing web applications. While this is a fairly broad description, individuals have used Frontier in many ways that require user interaction through web browsers, including such applications as automated software testing. The application has an integrated scripting system, multi-threaded runtime object database, outline editing, and even a built-in web server. Many individuals have implemented Frontier in different ways, but its primary usage has been for content management. Manila, which is described later in this article, is a content management system that is based on Frontier, and as such, many of the Frontier purchasers are really looking for Manila functionality, but they are also buying the system for its built-in support for a number of major industry standards such as HTTP, XML-RPC, SOAP, RSS, and third-party support for WebDAV. Having support for these standards makes it a powerful tool for creating integrated web applications.

As a web development and content management system, Frontier competes with everyone from Microsoft to Broadvision and Vignette. However, Frontier's main selling point is that it is making content management available to people who can't afford the more expensive and support-intensive Vignette or Broadvision approach. These applications are not practical for more small to medium sized organizations. Frontier is licensed on subscription-based, annual basis at $900. The system is focused at being developer friendly, but many people are involved in Frontier development without prior development experience. The scripting language within the product is called UserTalk, and has been under development for 13 years. It is very similar to the "C" programming language, but is more loosely typed and is compiled at run-time.

The final results of Frontier processing can be either rendered dynamically or result in a set of static pages for optimal efficiency. Frontier is often used in a multi-author website capacity, serving as a central point for content generation, management, and approval. However, don't let its size and cost fool you. It has been used to support many large, high-traffic sites including The Industry Standard's main intranet page, and Scripting News main web site.

While Frontier serves as the basis for web site content management, Manila is the actual application that performs most of the functionality. It started out primarily as a browser-based content management system, and builds upon an object database that stores the various components of a web site. Originally people would edit the web sites in the database itself, but now users can edit their Manila-based websites using either the web browser interface or the XML-RPC and SOAP based editor, Radio. Described later in this article, Radio Userland provides a mechanism for remote content develop-

XML Adventures in Userland (cont.)

(Continued from page 1)

ment and delivery as well as containing technologies such as an outliner.

In addition to basic content management functionality, Manila contains major features in the area of membership and discussion groups. Manila can support complex discussion and membership requirements and allows for the discussion groups to be used as a "back-end editorial room" for situations where there are multiple writers and editors. This allows the various portions of a content development organization to have a round table and give feedback about the various parts of a web site. Manila comes with source code specific to the Frontier system for additional developer-friendliness.

The membership and discussion is implemented in a middle tier between Manila and Frontier at the "Main Responder" level. Main Responder is a set of application functionality built on top of the Frontier website framework that provides an HTML rendering engine. All functionality and elements that is built on Manila or Responder share templates as a common element. A designer creates a HTML template, which allows a writer to focus on the content rather than the layout. In this manner, the website framework, and in turn Manila and Main Responder, separates design from content.

The final part of the three-legged stool is Radio Userland. It has been in existence for about a year, and another release is scheduled in the next few months. The main idea is "Content Management for the masses". Growing out of Frontier, it produces a mechanism to create and update content on a periodic basis. The main idea is based on the content of a web log. An average news site has a bunch of stories, and each will have its own page, with some calendaring requirements for scheduling the articles and their appearance on the web site. Radio Userland provides a mechanism to feed this data while supporting a calendar and time-based editing. Radio Userland can read news from RSS feeds as well as individual news submissions. These various news feeds are supplied to the website with editor approval and categorization. The content is then placed on the final web server for publication. Radio Userland systems can also produce its own news feeds for further news distribution.

An outgrowth of Radio Userland is the XML storage system which provides a persistent means for storing XML data. The motivation for this development is the fact that some users don't have a full-time internet connection, but still wish to publish news feeds. The XML storage system provides a store for users to save their XML data, while an XML-RPC or SOAP interface provides a means to publish the data to the end system. This system also allows external data feeds such as RSS news feeds to be written as static feeds locally. This allows users to have a persistent data store of news information. However, this store is also smart about its data. As a news feed is received, the system can figure out all the users and systems that are subscribed to the feed so that the system can automatically update the correct portions of the various sites. The system uses a content-based notification scheme, which depending on notification method can notify a user of an update.

Even though the products developed by Userland are in a very competitive landscape, the idea is not merely to provide applications, but to push a philosophy embodied in each of these applications. This philosophy is that users own their data. Simple content management and community applications such as eGroups don't provide a way for users to get their content back out once it has been submitted to the application. That information would disappear if the service was to go away. So, Userland has provided applications and a means for users to get their data out in an XML format. The idea is to build applications that leverage information in a standard manner, giving users the freedom to manipulate their data as they please.

> *The idea is to build applications that leverage information in a standard manner, giving users the freedom to manipulate their data as they please*

This philosophy is embodied in Userland's commitment to open standards, and the ease by which its applications can be integrated with others. This is further enhanced by Userland's support for protocols such as XML-RPC and SOAP. With over 30,000 Manila web sites currently in operation, XML-RPC is the most widely used XML-based inter-process communications protocol, beating out SOAP almost two to one. While Userland doesn't see SOAP and XML-RPC necessarily as competing standards, SOAP has been extended for support of things like document support and intermediaries which XML-RPC has shied away from. Initial deployments of SOAP will be vendor-specific, and some implementations of specific portions of the spec may not be published. However, in support of its goal of open standards and ease of integration, the Userland set of products has been extended to support SOAP as an interoperability protocol.

With its continued development and support of XML-based standards, and its philosophy of data ownership for all, there is no doubt that Userland will continue to be a major player in the XML industry. It is hoped that some of its approaches and technologies achieve widespread usage and success, and help further the ability for XML to be used as a content management and information distribution technology.

Contact Information:
www.userland.com
John Robb jrobb@userland.com
Jake Savin jake@userland.com

Featured Companies and Products in this Issue

Autonomy
www.autonomy.com
David Appelbaum dma@us.autonomy.com
Linda Findley lfindley@west.schwartz-pr.com

Business Genetics
xBML
www.businessgenetics.net
Cynthia Storer - cstorer@csapublicrelations.com
Cedric Tyler - cedric@businessgenetics.net
Steve Baker - steve@businessgenetics.net

Cape Clear
CapeStudio
CapeConnect
www.capeclear.com
Tom Murphy Tom.Murphy@capeclear.com
Annrai O'Toole annrai.otoole@capeclear.com

COMTEX
NewsML
www.COMTEXnews.net
Heather Henderson hhenderson@comtexnews.net
Slawek Ligier sligier@COMTEXnews.net

Datapower
XA35
DGXT
XSLTMark
www.datapower.com
Rick Lifsitz rick@datapower.com
Eugene Kuznetzov eugene@datapower.com

Infonyte
Infonyte DB
www.inofnyte.com
Thomas Tesch tesch@infonyte.com

LOGML
www.cs.rpi.edu/~puninj/
John Punin puninj@cs.rpi.edu

Mozquito
Forms Markup Language (FML)
XHTML-FML
www.mozquito.com
Sebastian Schnitzenbaumer schnitz@mozquito.com
Leonard Saffir lenpr@bellsouth.net
Mike Hightower mhightower@tilsonpr.com

Open Applications Group, Inc. (OAGI)
www.openapplications.org
Tony Blazej ablazej@openapplications.org
David Connelly dconnelly@openapplications.org

Open Buying on the Internet (OBI), Part of CommerceNet
OBI
www.openbuy.org

www.commerce.net
Patti Kozlovsky Koz2@ix.netcom.com
Kevin Kienast kevin@commerce.net

Outrigger Hotels & Resorts
OTA
www.opentravel.org
www.outrigger.com
Joe Durocher joe.durocher@outrigger.com

Scientio
XMLMiner
XMLRule
www.metadatamining.com
Andy Edmonds andy@metadatamining.com

Userland
Frontier
Radio
Manila
XML-RPC
www.userland.com
John Robb jrobb@userland.com
Jake Savin jake@userland.com

Viata
OTA
www.opentravel.org
www.viata.com
Laura Parker lparker@viata.com
Harold Williams hwilliams@viata.com
Gail Slawson gslawson@viata.com

ZAPTHINK, LLC

681 Main Street
Suite 2-11
Waltham, MA 02451

Phone: 781-207-8534
Fax: 786-524-3186
Email: info@zapthink.com

Continuous Innovation
through Continous Thought

Subscription Form

Fee: *ZapLetter* is $595 for 12 monthly issues. ZapLetter is distributed electronically. Please add $95 for print version. Fill out and fax or mail this application to ZapThink. Order online at http://www.zapthink.com. Please call or email info@zapthink.com for reprint or back order information.

Name
_____

Company / Organization
_____

Address
_____

_____

_____

Phone

Method of Payment

☐ Check

☐ Visa   Number of Issues

☐ MasterCard

☐ American Express

_____   _____
Credit Card #   Exp. date

_____
Signature

ZAPTHINK, LLC

681 Main Street
Suite 2-11
Waltham, MA 02451

Phone: 781-207-8534
Fax: 786-524-3186
Email: info@zapthink.
com