ISQ

INFORMATION STANDARDS QUARTERLY

SPECIAL EDITION: YEAR IN REVIEW AND STATE OF THE STANDARDS

SUSHI IMPLEMENTATION: THE CLIENT AND SERVER EXPERIENCES

DEDICATED TO STANDARDS

EXTENDING AND PROMOTING THE USE OF OPENURL

ESTABLISHING SUGGESTED PRACTICES REGARDING SINGLE SIGN-ON

NISO

How the information world CONNECTS

IP
[IN PRACTICE]

SUSHI
IMPLEMENTATION EXPERIENCES

CLIENT
(PAGE 18)

VS.

SERVER
(PAGE 20)

I n 2007, NISO published the *Standardized Usage Statistics Harvesting Initiative (SUSHI) Protocol* (ANSI/NISO Z39.93), which defines an automated request and response model for the harvesting of electronic resource usage data utilizing a web services framework. It was developed to replace the time-consuming, user-mediated collection of usage data reports. COUNTER reports are the main type of usage data that is being harvested with SUSHI. Use of COUNTER with SUSHI requires that the reports be in XML format, which further enables the automation of importing this data into an electronic resource management (ERM) system.

In 2008, COUNTER issued Release 3 of the *Code of Practice for Journals and Databases*. New to the release was the requirement to support SUSHI in order to be considered compliant. This further added to the impetus of implementing SUSHI. Currently 34 publishers or aggregators are listed on the SUSHI Server Registry. The registry is one of many implementation aids that have been developed by the SUSHI Standing Committee and the SUSHI Developer community and posted to the SUSHI website.

Two implementers of SUSHI, one for the client side and one for the server side, have shared their experiences in implementing SUSHI in their organizations in the following articles. Neither developer had any previous experience in working with web services, but with the help of the SUSHI Developer community they both created successful implementations with only a few headaches along the way.

Brinda
Shah

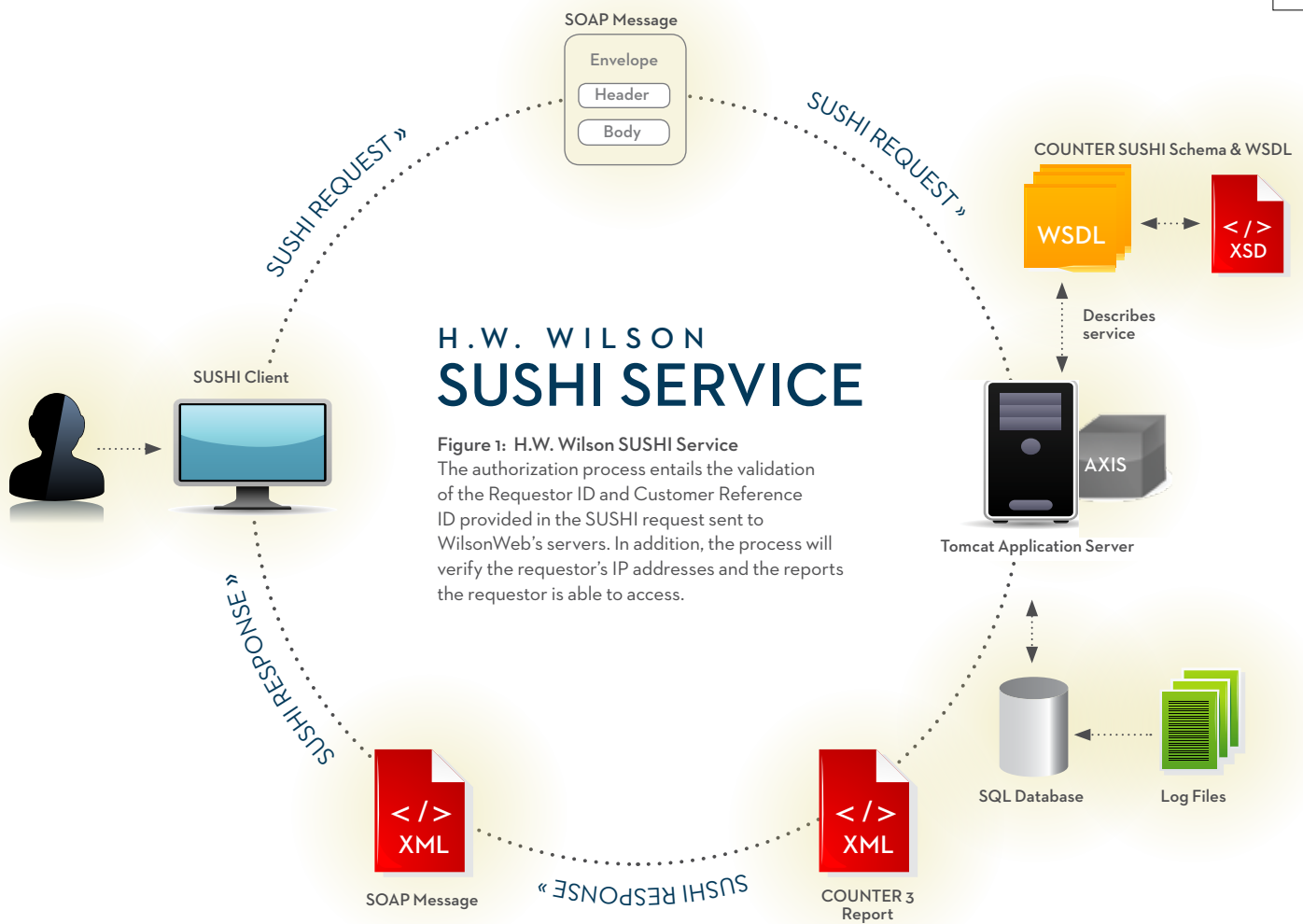## SUSHI Implementation:
THE SERVER SIDE EXPERIENCE
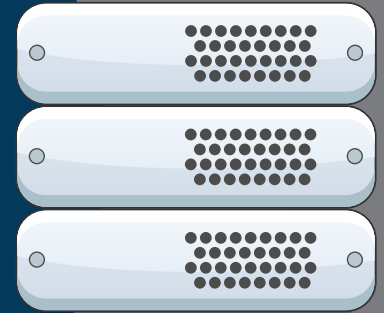
BRINDA SHAH

In February of 2010, I was tasked with developing a SUSHI (Standardized Usage Statistics Harvesting Initiative) server implementation for the H.W. Wilson Company. We had previously been compliant with Release 2 of the COUNTER *Code of Practice* for supplying resource usage statistics and wanted to continue our compliance with Release 3, which required support of the SUSHI protocol. My experience with SUSHI, which I describe in this article, went from total confusion with endless terms and technologies that I had little or no knowledge of to a pretty neat and straightforward service that turned out to be one of the most interesting projects I have worked on.

When COUNTER 3 was published and I was assigned the SUSHI project, I have to admit that I skipped through all the SUSHI headings not knowing what it was all about. Instead, I concentrated more on reviewing the code change requirements to update the COUNTER 2 reports that we serviced at the time. Seeing that there were not a lot of needed changes to the reports, I thought all that was needed was to lay out the processes and put them together.

Soon I came to learn about SUSHI and that it was an added requirement to the COUNTER Release 3 standard, aimed at making it easier to share and consolidate usage statistics through an automated process. The NISO website had a section dedicated to SUSHI that contained everything that I needed. It was well organized, detailed, and provided information on how to get started, FAQs, links to tools, and sample data to use. It was the just the guidance I needed to understand all the pieces involved.

Figure 1: H.W. Wilson SUSHI Service
The authorization process entails the validation of the Requestor ID and Customer Reference ID provided in the SUSHI request sent to WilsonWeb's servers. In addition, the process will verify the requestor's IP addresses and the reports the requestor is able to access.

Once all the necessary components were identified and a clear picture was shaping up in my head, I realized that unlike the basic client/server setup we had in place for delivering existing COUNTER reports, SUSHI involved implementing a web service framework utilizing a SOAP messaging protocol to retrieve XML-formatted COUNTER reports. After researching and reviewing the technologies available to develop web services and based on my own personal experience as a programmer, I decided to adopt the J2EE framework because of its power of cross-platform development and its easy implementation with open source components like Apache Tomcat (web application server), Axis (SOAP engine), and Eclipse (development tool).

Developing a SUSHI web service required **eight tasks** for the different workflow components I needed to put in place.

They were:

1. **Capture** usage statistics from our system for the individual customers in compliance with the COUNTER requirements.
2. **Generate** COUNTER reports in XML format using the COUNTER schema.
3. **Identify** and implement a security model.
4. **Develop** a SUSHI web service where requests can be submitted.
5. **Develop** a process to analyze the request and authorize it (when allowed) to access the relevant data.
6. **Develop** a reporting service to fetch the requested COUNTER report(s).
7. **Assemble** a SUSHI response with the fetched COUNTER report(s) inserted.
8. **Send** the SUSHI response back to the requesting client

> "I want to mention the excitement and self-satisfaction I felt when I made my first successful request and response for a COUNTER DB1 report. I had days of struggle prior to that with tearing things apart and putting the pieces back together. There were times when the request seemed to reach my service and get swallowed up somewhere in the transition even though everything was what and where I thought it should be."

Since the H.W. Wilson Company was already serving COUNTER 2 reports, and very little work was needed to make the minor changes in the usage statistics we were pulling for those reports, no additional work was needed to capture the usage data. The basic tutorial available on the Eclipse site was handy in setting up the environment needed to develop the web service using Apache Tomcat and Axis. Setting up a SUSHI server required an understanding of how these components work and connect with each other, which was a challenging experience for me and probably for anyone else who was not very familiar with web services. After some trial and error and help from the dedicated SUSHI development group, all the components finally seemed to be in place (see Figure 1).

I want to mention the excitement and self-satisfaction I felt when I made my first successful request and response for a COUNTER DB1 report. I had days of struggle prior to that with tearing things apart and putting the pieces back together. There were times when the request seemed to reach my service and get swallowed up somewhere in the transition even though everything was what and where I thought it should be. After an extensive debugging process, establishing a stable environment, and testing using some of the different available SUSHI testing clients like SOAPUI and the web client from University of Pennsylvania, the SUSHI service was ready to deploy.

I would like to conclude this article with stating my support for what SUSHI offers and the many avenues it opens for data sharing and the consolidation of data to serve the customer base in the library, publishing, and information technology communities. Implementing a SUSHI server has satisfied our customers with their requirements of getting consortia-level database and journal reports. It is the perfect bridge or gateway for customers to consolidate their usage data from the different vendors and services they use. The experience I gained in learning about web services and SOAP will definitely be useful for future projects. I would also like to extend special thanks to the help and support provided by the SUSHI Development Group as well as all the support and documentation provided by the NISO website. Further thanks to my manager as well as the VP of Information Systems at the H.W. Wilson Company for their continuous support for the project and for the encouragement to volunteer and share my experience through this article. | IP | doi: 10.3789/isqv23n1.2011.05

BRINDA SHAH <bshah@hwwilson.com> is a web programmer at H.W. Wilson <www.hwwilson.com>.

SUSHI website
www.niso.org/workrooms/sushi

SUSHI Developers E-mail List
www.niso.org/lists/sushidevelopers/

SUSHI Standard (ANSI/NISO Z39.93)
www.niso.org/standards/z39-93-2007/

RELEVANT
LINKS