

10.5 USING JAVA, XML AND XSLT TO CREATE SECURE INTERNAL ACCESS TO LOCAL NOAA RESEARCH SCIENCE PROJECT AND BUDGET INFORMATION

Jason E. Fabritz and Nancy N. Soreide*
University of Washington/JISAO, NOAA/PMEL, Seattle Washington

1. INTRODUCTION

The OAR Financial Data Management System (OAR FDMS) has been developed to meet the needs of the eighteen laboratories and program offices in NOAA Research. Implemented in Microsoft Access, this internal system is hosted on a server located at PMEL in Seattle, and is used by the administrative staff in the NOAA Research laboratories and program offices throughout the country with CITRIX Metaframe. Administrative staff input budget and administrative information using Microsoft (MS) Access forms and queries and web forms, and can view status of all current data while on-line.

We have created a secure, internal "PI Reports" software system to provide project leaders or Principal Investigators (PIs) within a single OAR Operating Unit with access to their own budget information in the FDMS databases. Administrative staff in the Operating Unit can establish appropriate access permissions so that each project leader is able to view only their own project information. Access permissions can be established to allow managers and supervisors access to the project information for every project leader under their management. In this paper, we describe the technologies utilized in building a PI Reports software system for the FDMS system.

2. DESIGN GOALS

There are many options available today for creating dynamic websites the present real-time data stored in databases. Each of these options has different strengths and weaknesses. The principal design goals guiding the selection of technology and design are:

* *Corresponding authors addresses:* Jason E. Fabritz, c/o JISAO, Box 354235, University of Washington, Seattle, WA 98195; e-mail: jfabritz@u.washington.edu. Nancy N. Soreide, NOAA/PMEL/OCRD, 7600 Sand Point Way NE, Seattle, WA 98115; e-mail: nns@pmel.noaa.gov.

Integration with existing infrastructure.

The website must be able to integrate seamlessly with the existing FDMS as it currently exists. It must not interfere with the system or impose additional operational constraints.

Insulation from changes. The website must be flexible enough to gracefully accept changes to the underlying FDMS without the need of massive alterations to website related source code, including the potential of FDMS switching to a different database implementation.

Administration. The system must provide a means to manage user's rights and accessibility to different types of information in the system.

Performance. The website must provide an adequate level of performance.

Scalability. It is desirable to produce a system that will be able to grow and fulfill future needs that are not visible at this time.

Cost. It is desirable to use technology that leverages existing software licenses, technology and developer experience.

3. ARCHITECTURE

In fulfillment of the design goal to create a system that was flexible enough to gracefully adapt to potentially major changes in the core FDMS system, the web access system was separated into three major pieces. The first piece is a core application server that integrates with the FDMS Access Database tables and provides a means of data access through a customized Application Programming Interface (API). This module is called the FdmsXML server. The second component is the intranet webserver itself. This component communicates with the FdmsXML server to obtain information stored in the core FDMS. The third module is the Administration module, an application that provides a graphical user interface for the purpose of displaying and modifying the security and configuration settings of the FdmsXML server.

3.1 FdmsXML Server

The FdmsXML Server hides the details of the implementation of the core FDMS database tables and provides an alternative API into the FDMS

system. The API provided by the XML Server consists of requests and responses written in XML and transmitted over the HTTP protocol. The XML Server listens for HTTP POST requests on a given port. When a request is received, it processes the request, queries the FDMS in real-time and returns the results through HTTP formatted as XML. With this module acting as an intermediary between client applications and database tables, changes in the underlying database tables only need to be reconciled with the FdmsXML server, saving a complete re-deployment of the entire system for minor changes.

The FdmsXML Server utilizes technology already present and compatible with the existing core FDMS. It runs inside of Microsoft's Internet Information Server (IIS) Active Server Pages (ASP) environment. Most of the business logic resides in a Dynamic Linked Library (DLL) written in Visual Basic (VB). It communicates with the Access Database Tables through Microsoft's Active Data Objects (ADO). It utilizes libraries provided IIS to facilitate the exchange of XML data through HTTP.

3.2 Administration Module

The FdmsXML Administration module provides a convenient graphical user interface for administering user accounts and security privileges for the FdmsXML Server. Administrative staff in the Operating Unit can establish appropriate access permissions so that each project leader is able to access only their own project information. Additionally, access permissions can be established to allow a managers and supervisors access to the project information for every project leader under their management.

The module was written in Java because of the excellent support for XML in that environment and for its cross-platform capabilities that are important in a heterogeneous computing environment like the one that can be found at PMEL.

3.3 Website Module

The FDMS Principle Investigator (PI) Web Reports site is the component of the system that most individuals will interact with. It translates requests from client browsers into the appropriate XML requests and queries the FdmsXML Server. In turn it interprets the results and sends back HTML to the client's browser. Users are required to log in with a password. Only information that is accessible to the client is sent from the FdmsXML

server to the web server. This helps prevent unauthorized access to sensitive information.

Like the FdmsXML Server, the PI Web Reports server runs inside of IIS. Most of the business logic resides in a compiled DLL written in Visual Basic. Another major technical aspect of the server is that it relies heavily upon Extensible Sylesheet Language Transformations (XSLT) transformations to quickly and efficiently generate HTML from the XML response retrieved from the FdmsXML Server.

4. SUMMARY

The internal OAR FDMS PI Reports system provides project leaders within a NOAA Research laboratory or program office with secure, near-realtime, web-based access to their own project budget information. This system has served as a basis and prototype for the SuperFDMS, which will provide headquarters management with access to financial information consolidated across the eighteen NOAA Research laboratories and program offices throughout the country (Burger, et. al., 2002).

5. ACKNOWLEDGMENT

This publication is partially funded by the Joint Institute for the Study of the Atmosphere and Ocean (JISAO) under NOAA Cooperative Agreement No. NA17RJ1232, Contribution #878. PMEL Contribution no. 2419. The views expressed herein are those of the authors and do not necessarily reflect the views of NOAA or any of its sub agencies.

6. REFERENCES

Burger, E.F, J. E. Fabritz and N. N. Soreide, 2002: NOAA Research Scientific Project and Budget Reporting Over the Internet. *18th International Conference on Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology AMS*, 13-17 January 2002, Orlando, FL (submitted)