December 2004

A Distributed Web Service-based Infrastructure for E-Document Management

Ou Liu
City University of Hong Kong

Jian Ma
City University of Hong Kong

Ji-Ye Mao
City University of Hong Kong

Ron Kwok
City University of Hong Kong

A Distributed Web Service-based Infrastructure for E-Document Management

Ou Liu, Jian Ma, Ji-Ye Mao, Ron C.W. Kwok
Information System Department, City University of Hong Kong, Hong Kong, China
{isliuou, isjian, ismao, isron}@cityu.edu.hk

Abstract

Document management plays a critical role for the R&D project management at National Natural Science Foundation of China (NSFC). The major problem faced by NSFC in document management is how to process a large quantity of electronic documents in multiple formats within one deadline during submission peak time. This paper describes a distributed document management solution with Web services to solve the peak load problem. A document extraction process is also included to handle the multiple document formats, with XML as the intermediate of interchanged messages. The solution is applied at NSFC and positive responses have been received.

Keywords: R&D project selection, document management, Web services

1. Introduction

In this article, we examine the national Research and Development (R&D) project management in China in general, and its existing document management problems in particular. To tackle these problems, we propose an open infrastructure of distributed document management based on the Web Services (WS) technology.

The National Natural Science Foundation of China (NSFC) is the largest government-funding agency in China. Its primary aims are to promote the fundamental and applied research generally, and to support R&D projects particularly. Supported by the Chinese government, NSFC's annual budget has been dramatically increased from RMB 80 millions in 1986 to over RMB 2,000 millions in 2003. Every year, NFSC receives more than 50,000 project applications from over 1,400 universities or research institutes. The project selection process is coordinated by the top managers of NSFC and accomplished by the seven scientific departments as well as their divisions underneath. Each project application is evaluated by the external reviewers and experts invited by the departmental divisions. For this purpose, NSFC maintains an external reviewer database with more than 30,000 records, and employs more than 700 experts from 69 disciplines for panel evaluation.

Document management is a key issue for the project management (Eloranta, et al., 2001). Throughout the project application and reviewing processes at NSFC, many kinds of forms in different document formats (e.g. HTML, Microsoft Word and Kingsoft WPS which are popular in China) are to be filled in, submitted, reviewed and exchanged by various stakeholders (e.g. applicants, reviewers, and coordinators etc.) from different institutions. The most frequently document types are project proposals and progress/final reports submitted by project applicants and review forms by external reviewers. The details of these forms are to be extracted and stored into relevant databases. For example, the details of a project proposal includes project title, project coordinator, project recipient organization, methodology, deliverables and expense budget, etc. Different document formats are accepted because of the

diverse backgrounds and preferences of users all over the country, as well as different accessibility of the Internet.

All these forms are usually submitted to NSFC and to be processed within a few days before a deadline. For example, more than 30,000 application forms are handed in before March 31st each year and to be processed within 15 days so that review forms can be sent out on time. The major problem is how to process this large quantity of electronic forms in multiple formats within one deadline. This brings a heavy workload to the back-end system. This problem has not appeared in the cases of other funding agencies. For example, in National Science Foundation of USA, the proposal submission time frame is distributed over the whole year and the document quantity is much smaller.

To solve this problem, there are two challenges. The first challenge is the compatibility issue. It is a requirement that the software is able to run on different computer operating systems; Windows, Unix, Macintosh, and Linux, and that the software is able to easily interact and communicate across systems. It will increase the cost and complexity of the software development if the heterogeneity cannot be well mastered. The second challenge is the interchangeability of documents in multiple formats. It should have a well-accepted intermediate format for the interchange of multi-formatted documents in the Internet environment.

A two-tiered solution has been adopted to handle these challenges and to support the existing two-tiered management process. At the local level, project applicants submit their proposals electronically to their respective universities or research institutions first for endorsement. Each organization extracts certain data from the proposals it receives, and then forwards the applications in batch to the NSFC for central processing. Correspondingly, the e-Document management solution consists of two parts, Internet-based Research Information System (IRIS) used at the local level and Internet-based Science Information System (ISIS, http://isis.nsfc.gov.cn) at NSFC. The distributed processing gives rise to the use of Web Services as the means for integrating the two levels, and facilitating information exchange and sharing.

2. The Proposed Solution

To solve the workload problem, we propose a solution in this section, including a document process workflow and a system infrastructure which supports distributed document storage management.

The document process workflow includes the following steps:

1. A registered applicant downloads the application form in Microsoft Word, or Kingsoft Word Processing System (WPS) format, which is the main rivalry word processor of Word in China, and fills in the form off-line.
2. The local IRIS then extracts data from the submitted form, and generates PDF document for reviewing and record keeping.
3. After gathering all submitted application forms, the organization manager of a fund recipient organization submits the processed application files (i.e., a data file containing extracted data for project management and the application documents) from IRIS to the ISIS through Web Services;
4. Once ISIS receives the submitted files from registered IRIS's, it then merges the data files into its central database, and saves application documents into designated directories of a document repository.

The proposed system infrastructure is shown in Figure 1. There are two levels of back-end systems: IRISs (Internet Research Information Systems) or mini-IRISs for different affiliated organizations (e.g. universities or institutes), and ISIS (Internet Science Information System) for NSFC. IRISs are built for the reception and extraction of the documents (such as proposals) submitted by the responsible persons of the affiliated organizations (see line (1) in Figure 1), as well as the local storage and management of the documents and project data. For those affiliated organizations that do not want a full version of IRIS, mini-IRISs are provided for document extraction and exchange only (see line (2)). After the submission deadline, the IRIS/mini-IRIS then compresses and packs up the extracted data collected in the affiliated organization and sends them to the ISIS of NSFC (see line (3)). Applications from affiliated organization without IRIS can be submitted to ISIS directly (see line (4)). In particular, IRIS/mini-IRIS and ISIS adopt XML as the intermediate format of exchanged documents and Web Services (WS) as the interaction technology. The proposed distributed infrastructure effectively solves the workload problem.

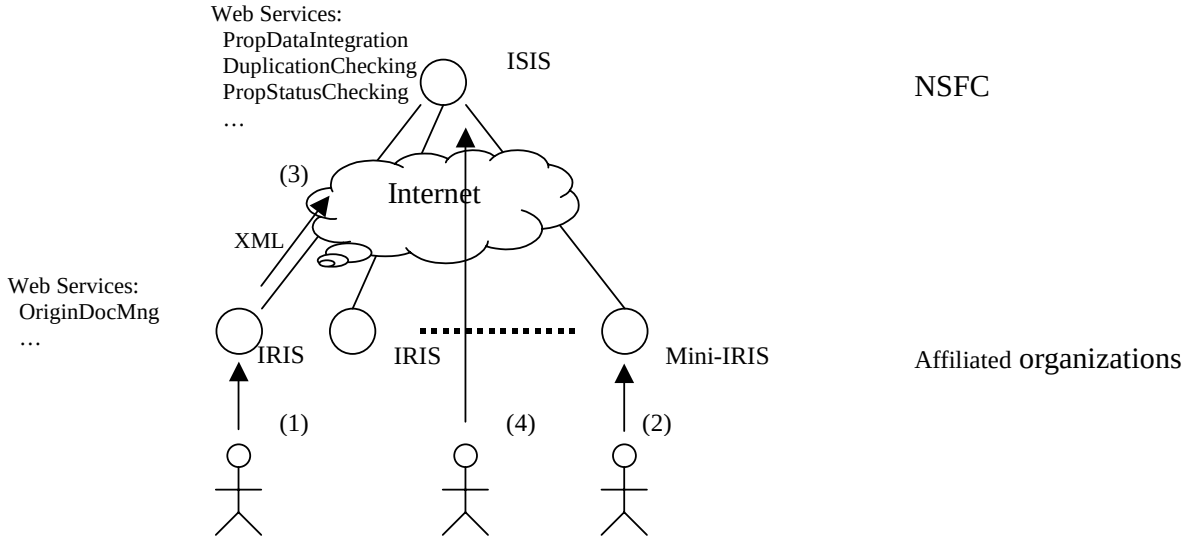

Figure 1. Web Service-based Distributed System Infrastructure

Corresponding to the two faced challenges mentioned before, we use relevant methods to resolve them. Firstly, the ISIS and IRISs all provides services (Figure 1) in the form of WS for better collaboration needed by distributed document management. The cross-language, cross-platform interoperability of WS well solves the compatibility challenge. Secondly, the process of document extraction and management is design for multiple-format document exchange. The details of these issues will be presented in the following sections.

3. Web Services and Implementation

Web Services (WS) provided by the ISIS and IRISs well support the collaboration among them. Each WS includes some relevant functions/operations to fulfill a specific task.

The ISIS usually provides some extra services for those tasks that require high authority or that cannot be fulfilled in IRIS. Examples include:

- "PropDataIntegration". It is responsible for the integration of the proposal data sent from different IRISs. An IRIS packages the extracted data in the relevant university and sends it in XML format to ISIS. The WS receives and classifies the packages, and fills in the data into ISIS database.

- "DuplicationChecking". There is a regulation in NSFC that each person cannot take too many projects: a normal researcher can only lead one project and attend up to three ones; and a senior researcher can only lead or attend up to two projects. The WS is used to check if the regulation is followed in ISIS. Meanwhile, there is the same requirement in IRIS before it submits proposal data to ISIS. However, the IRIS doesn't have enough information in its own database because a project or a proposal may involve applicants from many universities/organizations. So it needs to call "DuplicationChecking" in ISIS from remote.
- "PropStatusChecking". It enables IRISs or even software agent in extranet to check the current approval status of a specific proposal.

As for IRISs, they also provide some services for calling from remote. For example, "OriginDocMng" provides functions for remote retrieval and management of original document copies.

The above services are implemented as WS for better communication between ISIS and IRISs. WS provide a standard means of communication among different software applications (Kreger, 2003). We can use any programming language to implement them upon any platform as long as following the specifications of WS. WS's platform- and language-independence feature makes the proposed framework easily extensible to support more document standards. Supporting another document standard is simply a task of adding another WS, because of the separation of service interfaces from implementations. They can be requested/called through well-accepted Internet standards and protocols. Also, without the WS mechanism, the remote access to many services in ISIS such as "DuplicationChecking" will be very troublesome; while it is very easy to call the services from remote in the proposed Web service-based infrastructure.

There are three layers for the implementation of WS (Figure 2). HTTP is used for the transportation of enveloped messages. SOAP (Simple Object Access Protocol), which is represented in XML syntax and restricted by XML Schema, is a standard messaging protocol across Web services in the Internet. The transportation layer and message layer form the execution foundation of Web services. At the service definition layer, WSDL (Web Services Description Language) is to describe functional interface of Web services. Issues across these layers include security, management, et al. We adopt Microsoft .Net framework to support the implementation.

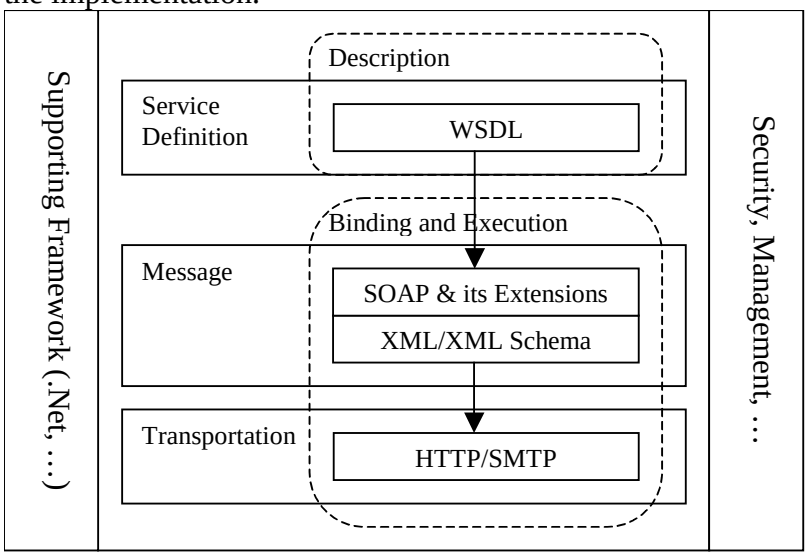

Figure 2. Implementation of the Web Service Infrastructure

To request a Web service, a user or a software agent firstly retrieves the WSDL description the wanted service complies with, and learns how to interact with it using SOAP messaging and SOAP RPC calls. Then it can send SOAP/XML messages by HTTP packages to interact with the Web service.

A sample WS description is partly shown in Figure 3 for the WS “PropStatusChecking”. It firstly defines the input and output messages in SOAP format, and the operation interface of the WS; then, it makes a binding and links to Internet address of the real WS implementation (the address in Figure 3 is just for illustration). With this definition, client applications can correctly connect with the WS by sending and receiving SOAP messages in the right formats.

```
<message name="PropStatusSoupIn">
        <part name="parameters" element="s0:PropID" />
</message>

<message name="PropStatusSoupOut">
        <part name="parameters" element="s0:StringResponse" />
</message>

<portType name=" PropStatusSoup">
        <operation name=" PropStatusGet">
                <input message="s0: PropStatusSoupIn" />
                <output message="s0: PropStatusSoupOut" />
        </operation>
</portType>

<binding name=" PropStatusSoup" type="s0: PropStatusSoup">
        ...
        <operation name="PropStatusGet">
                ...
        </operation>
        ...
</binding>

<service name="PropStatusChecking">
        <port name="PropStatusSoup" binding="s0: PropStatusSoup">
                <soap:address location="http://localhost/ PropStatusChecking.asmx" />
        </port>
</service>
```

Figure 3. WSDL fragments defining of a sample WS

4. E-document Extraction and Management

The system architecture for document extraction and management in the ISIS and each IRIS are very similar. It can be shown in Figure 4 with ISIS as example. It includes two process steps: 1) at the front-end, system users including applicants and external reviewers, etc., download the document templates and submit the completed ones in different formats and by different ways; 2) at the back-end, software agents extract the incoming messages, then store the documents and extracted data into document repository and databases respectively. We will introduce the two parts in details respectively as follows.

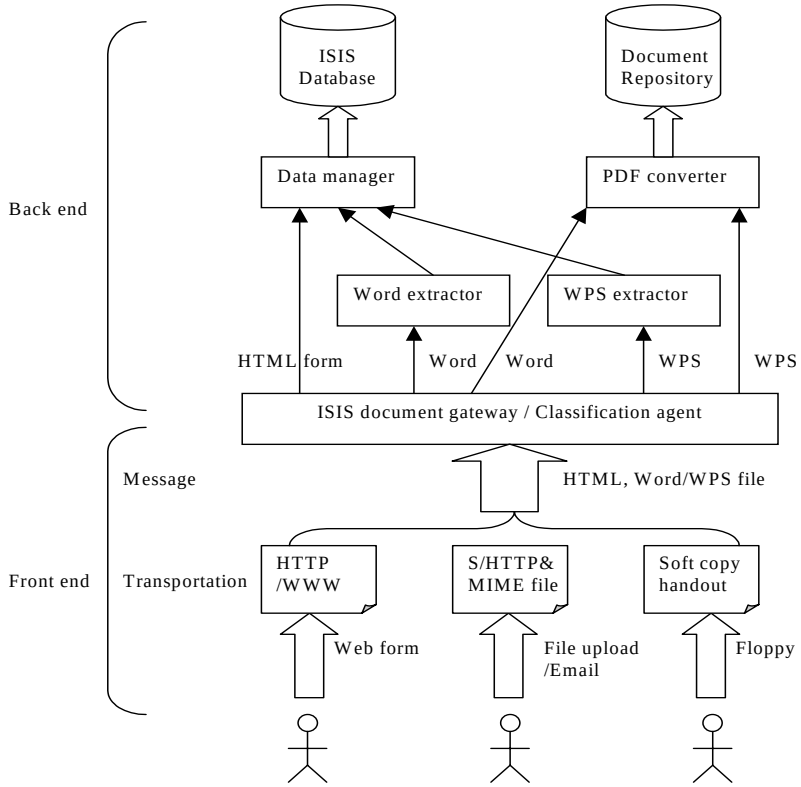

Figure 4. e-Document extraction and management architecture

Front-end:
Front-end users interact with the systems using standard Web browsers, electronic mail applications, or even floppy disks. There are two layers, i.e., transportation layer for enveloping and carrying messages through the Internet, and message layer for representation of the messages.

The transportation layer includes many different transportation protocols depending on the mode of submission. Simple Mail Transfer Protocol (SMTP), Secure Multi Purpose Internet Mail Extension (S/MIME) and Secure HyperText Transfer Protocol (S/HTTP) are three major protocols being used in the transportation layer. Security infrastructure has been applied on the usage of these three protocols. The system requires users to provide their certified key when doing the transfer. The keys are released by NSFC along with the user registration and authentication process.

In the message layer, HyperText Markup Language standard and third party document standards (Microsoft Office, Kingsoft WPS, etc.) are used as the representation vehicles. Applicants can submit proposal data using HTML forms, Word or WPS file attachments.

When choosing HTML form format, users need to fill into HTML form all proposal data which need to be filled into the ISIS database. Then, ISIS system receives the data and generates a semi-filled document (with the submitted data already filled) in Word or WPS format by option. The applicant downloads it and fills in other required content.

When choosing Word or WPS format, users download an empty document templates and need to fill in all required data. Before submission, the Word or WPS file will be checked automatically, and macros will run and generate XML meta-information for the data needed

to be filled into the ISIS database, which is imbedded in the document file and can be extracted by components in back-end systems.

Back-end:
In ISIS, the submitted documents are sent to corresponding extraction agents after the classification is done by the classification agent of ISIS. The various extractors (in Figure 4) extract certain information from different kinds of document formats and generate XML format data for a component called “Data manager”, which parses the XML data and exports them into ISIS database. XML provides a simple, standard, self-describing way of storing and exchanging text and data in the Internet (Harold, 1999). This makes the solution more independent from the implementation platform. Even when some components, such as the DBMS, are to be changed, only minor modification is needed for the system. Finally, a component called “PDF converter” converts the whole proposal documents into PDF format and keeps a copy in the document repository.

As for the implementation of extractor agents, various techniques have been applied. When using HTML document standard, a simple web server can extract the data from the form POST/GET data packet. The extraction of other document standards, e.g., Microsoft Office, and Kingsoft WPS, are modeled based on the reconstruction of document using data manipulation functions supported and deployed by third party in different industrial component standards. For example, Microsoft Office supports the Distributed Component Object Model (DCOM). Hence, data extraction can be achieved through the same process. Since some XML meta-information has been generated by macro and imbedded in submitted files and XML is used as the intermediate data format in our solution, the work load of those extractors is reduced and the processes are standardized, which make it easy to support more third party document format choices.

5. Implementation of different components for e-document management

The implementation of different components in the proposed e-document management architecture (Figure 4) is introduced below.

Implementation of Word extractor
Table 1 presents the supporting software components for data interchanged using Microsoft Office document standard.

Table 1: Supporting Components for Data Interchange in Word Format

Component	Description	Correspondent Technique
Fill-in Software	Software for fill proposal data into Word document templates	Microsoft Word Application
Extraction Software	Software for extracting proposal data from submitted Word documents	Microsoft Word components with server transaction container service (Microsoft Transaction Server, for example)

To implement the extraction agent, a new type library is used to glue both the database manipulation components and Word components to provide both database operation and document extraction function. In this case, the ActiveX standard is used as the basic protocol to communicate with Word component. Moreover, Microsoft Transaction server (MTS) is

used to support enhanced component objects with distributed database transaction support, to integrate both the extraction and database operation processes.

Implementation of WPS extractor
Kingsoft *Word Processing System* (WPS) file format is one of the most widely used document standards in China. As one of the key software vendors in China, Kingsoft have released its WPS Office series to compete with its powerful US-based archrival Microsoft's Office XP. For its wide adoption in China, WPS file format is selected as one of the document standards in ISIS. WPS Office series also provide API for the manipulation of document content. Similar to the implementation mechanism of Word file extraction, imbedded macro extracts data and the WPS extractor can get the data out at the server side by calling WPS Office supporting components.

Table 2 presents the supporting software components for data interchanged using WPS document standard.

Table 2: Supporting Components for Data Interchange in WPS Format

Component	Description	Correspondent Technique
Fill-in Software	Software for fill proposal data into WPS document templates	WPS Office series
Extraction Software	Software for extracting proposal data from submitted WPS documents	WPS Office extensions

Common to the above two kinds of extractors is that they all present the extraction results in XML format, which is targeted to provide a method for Web interchangeable data representation. XML is adopted as the intermediate representation format for its wide acceptance and global standardization. In the Word/WPS file templates downloaded from ISIS, macro codes have in fact been imbedded to generate XML format information so as to reduce the workload of the serve side and increase the efficiency.

Implementation of Data Manager
After returned by the extractor agents as the extraction result, the XML messages are sent to the data manager for filling into the database. The data manager calls the standard package for XML extraction, which helps to rebuild the Document Object Model (DOM) by translating the serialized XML text. The document object is accessible via document object model.

The data manager includes two components, i.e. a database access component and a data integration component. The database access component provides the basic support for database communication, including connection creation, transaction coordination and query execution. The data integration component receives the extracted document data from remote IRISs and integrates them into exsiting data set. The data integration component is wrapped as a Web service so that it can be called from remote sites (IRISs) in a standard manner for data exchange.

Implementation of PDF converter

Adobe Acrobat *Portable Document Format* (PDF) file format is selected as the standard format of document base in ISIS, because it is one of the most widely used document standards for exchange of document information in electronic business applications. Besides portability and platform-independence, it provides support to DOM. The PDF converter is implemented by the programming interface of Adobe Acrobat. It calls components for converting Word to PDF or converting WPS to PDF respectively.

6. Application and Implications

The two-tiered e-Document management at NSFC has been proven effective and successful via the pilot run in 2002 involving 16 universities and research institutes. In 2003, it stood the test at the national level and sustained the annual peak load in March. The Web service infrastructure provides a standard and efficient communication and coordination way among those distributed document management systems. It has greatly improved data reliability and quality, timeliness, and consistency between NSFC and local universities/research institutes.

The e-Document solution has shortened the cycle time, and saved processing expenses. As a result, NSFC has decided to waver the RMB500 (about US$60) processing fees to be paid by the applicant of each proposal. The e-Document management solution has led to significant efficiency gains and reduction of administrative costs, by eliminating the need for multiple hardcopies of proposals, postage associated with external reviews, and data entry work. Whereas five hardcopies were still required by NSFC in 2003, only one copy with signatures and official stamps is required this year for archival and legal considerations. This will result in further efficiency and savings for all stakeholders.

Moreover, the e-Document management also benefits fund recipient organizations. Compared to the previous paper-based approach, major benefits include reduced workload, elimination of data entry/re-entry, PIs' self-management and control over their proposal preparation, and availability of managerial reports and statistics generated by the systems. A limitation of the solution is that institutions without fast or reliable Internet access cannot run IRIS. Instead, they have to use a standalone version for handled proposal submission and data extraction. Furthermore, as an Internet-based system, IRIS is subject to network infrastructure constraints and far more complicated in system installation and technical support.

A unique strength of the NSFC e-Document is that PIs can use a popular word processor to prepare their entire proposals offline, and submit them to their local administrators. This can be done without Internet access, which is a necessary and useful alternative for a national e-Document management solution for a developing country like China. Moreover, the Word/WPS-based full proposal affords several distinct advantages: (1) compared to Web-based form-filling, the Word/WPS-based proposal is easy for frequent access and modification, and easy to share among collaborating researchers; (2) consistency is maintained in the format and layout between the electronic copy and hardcopy for signatures and review; and (3) the process of proposal preparation is convenient and compatibility with users' normal way of document preparation, such as custom formatting and the use of advanced word processing functions.

A survey was conducted on a sample of 120 professors and research administrators from 16 universities in China after the pilot-run of electronic submission of proposals in 2002. The great majority of the respondents (95%) to our survey found the e-Document solution easy to use. Such high usability score proves that the offline preparation of full proposal is easy to use, and much welcome by researchers. Similarly, 95% of the users either strongly agreed or

agreed that e-Document management was convenient. Most importantly, 87% either strongly agreed or agreed that it enhanced efficiency.

7. Conclusion

In summary, the NSFC e-Document solution has effectively addressed the volume, peak-load problem, and Internet infrastructure challenges. The solution relies upon XML as the intermediate of interchanged messages, which provides a simple, standard, self-describing way of storing and exchanging data between different platform and systems. Part of the data extraction work is done prior to proposal submission, which reduces the workload at the server side. Furthermore, Web services provide the means for integrating the two tiers at the local and NSFC central levels, which makes information sharing and exchange between the two tiers consistent, secure, and reliable. The experience gained from this work is easy to be generalized to other applications on large quantity document exchange and management.

Reference

Eloranta, E., Hameri, A., and Lahti, M. "Improved project management through improved document management," *Computers in Industry*, (45: 3), 2001, pp. 231-243.

Kreger, H. "Fulfilling the Web services promise." *Communications of the ACM*, (46: 6), 2003, pp. 29-30.

Harold, E. "*XML bible*," IDG Books, NY, 1999.