XML Enables a High-Volume, Near Real-Time Information Analyst Support System (IASS)

Presented at IntelCon 2006 by:

William Wolf
Assistant VP/Division Manager

Science Applications International Corporation (SAIC)
1997 Annapolis Exchange Parkway, Suite 200
Annapolis, MD 21401

Presented at IntelCon 2006 by:

William Wolf
Assistant VP/Division Manager

Science Applications International Corporation (SAIC)
1997 Annapolis Exchange Parkway, Suite 200
Annapolis, MD 21401

Biography

Mr. Wolf is an assistant vice president at SAIC and is the deputy division manager of SAIC's Intelware Solutions Division. Mr. Wolf is an experienced IT professional and manager and has been involved with all aspects of the software development life cycle. Mr. Wolf has provided IT consulting and management for a variety of government clients including defense and intelligence agencies.

Mr. Wolf's recent and current projects include:

- Technology lead in an information technology modernization effort at the FBI. In addition to system design and architecture, Mr. Wolf has been working on document management, use of natural language processing technology to do automated information extraction from text, link analysis and data visualization.
- Responsible for design and implementation of specialized databases to support the efficient storage and retrieval of more than 1.5 billion XML documents of various types. As part of this same effort Mr. Wolf contributed to many related application development and data flow processing activities. Many of these documents are transformed using feature extraction and other natural language processing technology to enhance the usability of the information before storage. These specialized text retrieval databases are one portion of a large data warehouse. Mr. Wolf was a principal designer of the overall system which uses a message oriented middleware broker and many small specialized components ("agents") to make data across the entire warehouse seamlessly available to client applications and end users.
- Principal investigator of an effort to educate other software engineers about a new enterprise data model defined using a formal XML document type definition (DTD) and a companion data element dictionary.
- Principal investigator of a project tasked to convert legacy data in many different proprietary formats (including binary formats) to a single standard XML representation. Mr. Wolf built a high-performance system to convert legacy data to XML using state-of-the-art XSLT technology. He wrapped existing legacy parsers with a Simple API for XML (SAX) layer to make them compatible with other XML-enabled COTS products.

Mr. Wolf has a Master of Science degree in computer science from the Johns Hopkins University, a Bachelor of Science in computer science from Seattle University, and a Bachelor of Arts in mathematics from Seattle University.

Abstract

In the war on terrorism, the people are represented by two quite intertwined and critically important groups: the information analysts who draw conclusions and provide those to decision-makers, and the information m anagement developers, who use XML to assist the analysts with correlation, transformation, assimilation and delivery of that information.

The key challenge is managing and monitoring the flow of information that might alert an information analyst to a high-threat event. The information that must be indexed and stored for immediate and term analysis comes in a multitude of formats. The information may include, for example, eye-witness accounts, transportation and shipping records, records of purchases of controlled chemicals, public announcements and even blogs. Success demands the ability to fuse data, including meaning and context, from disparate sources into a coherent whole. New records arrive at the rate of thousands per second, and overall data storage is in the terabytes. Fast load-to-index times are required, as are full-text search and retrieval capabilities. Scalability and storage efficiency are a must.

We have developed and deployed multiple systems to meet this challenge. The IASS described here implements an architecture that satisfies all these requirements and is extremely scalable, flexible, and fault-tolerant. The IASS fuses structured and unstructured information from across the enterprise and provides analysts with full search capabilities across billions of records. XML is the enabling technology for IASS and in conjunction with XSLT, provides a common language for configuration, data interchange, data access and presentation.

IASS's data sources include relational databases, text and XML repositories, and analytic applications. XML and text data records comprise about half of the over 4 billion records stored in a variety of languages and structures. The IASS strategy for managing large volumes of diverse data is to handle each with the most appropriate DBMS for that particular data type. The use of the text-centric system for XML data overcomes performance and efficiency issues associated with using an RDBMS with text or XML extensions. The text DBS also allows creation of customized text parsers and indexing algorithms, providing unique search features. Full support of XML, including XPath®, provides the ability to easily load multi-language and hierarchical XML documents. The text database copes with high data ingest volumes: the millions of new records that are added to IASS every day dictate that approximately 1,000 new XML records per second are indexed.

The IASS application uses a collection of distributed, loosely-coupled components to find, collect, analyze, and synthesize information. A commercial Web services messaging system is used to bind the components together; XML-based messaging allows the components to interoperate in virtually any language, to fulfill virtually any function. The IASS components, which serve as database adapters, user interfaces, or to reflect business logic, all connect to Web services in a hub-and-spoke architecture. The loosely-coupled design provides the added benefit of fault-tolerance. In fact, this feature has been exploited to migrate components from machine to machine, during business hours, with no downtime.

XML is used ubiquitously as markup to facilitate data fusion. XSLT engines (software and hardware) are used to perform just-in-time transformations of XML information into the format requested by the client application. XSLT re-purposes data for a variety of applications and audiences, such as management and the news media. XSLT also transforms XML into intermediate forms optimized for automated analysis.

XML and XML-related standards provide the underpinnings on which the highly successful IASS application rests. These technologies allow IASS developers to focus on the problem at hand and apply the best tools to implement solutions, using XML for information encoding, transformation, assimilation and delivery.

The Information Analyst Support System

The Information Analyst Support System is used by analysts to query relevant information from a massive warehouse of diverse information pertaining to their specific area of interest and to identify specific targets of interest.

These analysts are faced with the extremely complex challenge of finding indicators of potentially hostile acts in unprecedented volumes of information. The information analyst delivers to decision-makers not raw data, but richer, deeper, supported information.

The information that comes to the attention of analysts comes from many sources, in many different formats, at rapid pace, and the interrelationships of those pieces of information are almost always indirect and sometimes disguised. Automated systems can support the analyst by converting documents and messages to a common format, taking advantage of that common structure to compare and contrast discrete elements of information, and aggregating the information into what appears to be a single storage/retrieval structure.

There is no one single Analyst Information Store, but application of the right tools and the right standards can provide the analysts with a virtual storehouse of information and with the means to assimilate and transform data, and to synthesize complex models and conclusions. The really hard part belongs to the analysts, who are judged on their judgment. Given the daunting task of scrutinizing huge volumes, selecting the "nuggets", and synthesizing a whole (from the parts), the analyst really should not be required to manage and manipulate the raw data. The IASS must do that. The key challenge is managing and monitoring the flow of information that might alert an information analyst to a high-threat event.

The system deals with extreme volumes of data … real-time data … around-the-clock. And, the diversity of the data calls for application of a number of different tools – there is no "one-size-fits-all" data management tool. The right approach, and a complex approach, is to use the right tool for the right job on the right data – but to ensure that to the analyst, all the information is understood and coherently represented. XML is *the* enabling technology for IASS and in conjunction with XSLT provides a common language for configuration, data interchange, data access and presentation.

IASS is, simply put, big and fast. Thousands of users. Terabytes of information. Tying together dozens of analytic tools, and data bases. The system provides rapid response to informed queries – simple ones return answers, searching against billions of documents, in seconds; more complex operations, such as establishing linkages and relationships, can take 20 seconds.

And the system evolves and grows. Scalability is always a challenge, and in every possible dimension – more users, more data, more types of data, more languages, more complex problem sets and questions. Choosing the right tools and technologies helps to provide scalable performance – and, yes, more hardware is an important component – but, one very key component is “building the system around the analysts’ information needs.” Understanding what the analyst will do with the information is a key component in loading, indexing, and storing the information, and in defining the analysts’ interface to the system, and in data presentation.

The IASS described here implements an architecture that satisfies all these requirements and is extremely scalable, flexible, and fault-tolerant. IASS’s information technologies include a middleware messaging system, relational databases, text and XML repositories, high-performance storage and XML-conversion hardware, and analytic applications.

The IASS application uses a collection of distributed, loosely-coupled components to find, collect, analyze, and synthesize information. A commercial Web services messaging system is used to bind the components together; XML-based messaging allows the components to interoperate in virtually any language, to fulfill virtually any function. The IASS middleware components, which serve as database adapters, user interfaces, or to reflect business logic, all connect to Web services in a hub-and-spoke architecture (see diagram). The loosely-coupled design provides the added benefit of fault-tolerance. In fact, this feature has been exploited to migrate components from machine to machine, during business hours, with no downtime.

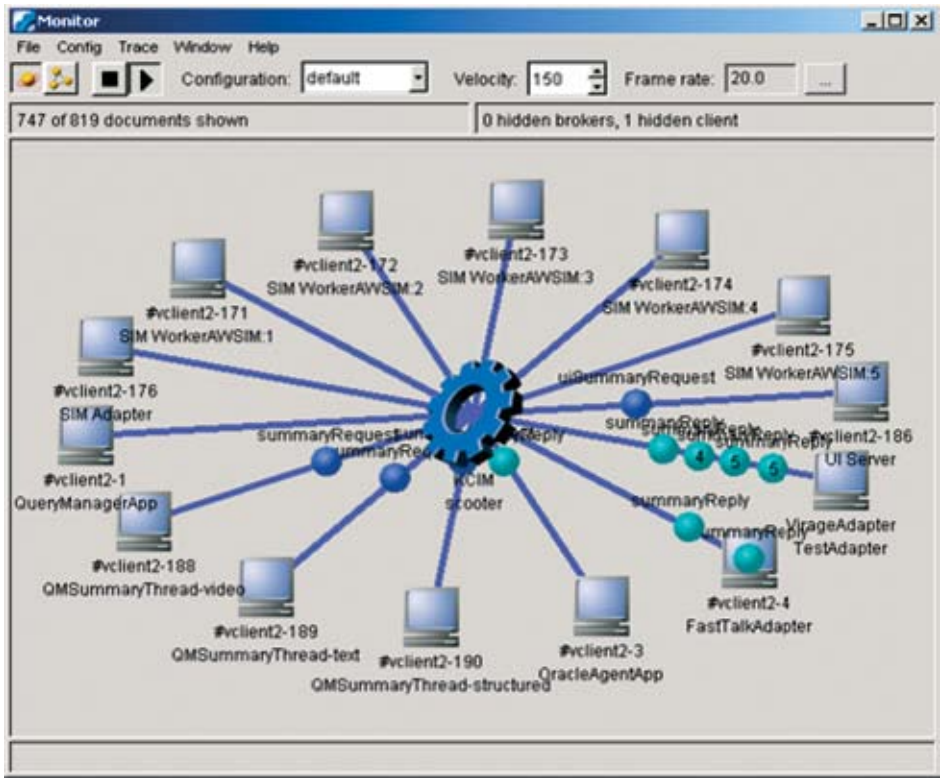

Figure 1

The centralized messaging system provides a common lingua franca for applications integration and gives the IASS:

- Single **API**
- Single query language
- Single results format
- A security gateway, and
- Ease of extension and expansion

The use of the text-centric TeraText® system for XML data overcomes performance and efficiency issues associated with using an RDBMS with text or XML extensions. The text DBS also allows creation of customized text parsers and indexing algorithms, providing extremely valuable search features. The text database copes with high data ingest volumes: the millions of new records that are added to IASS every day dictate that approximately 1,000 new XML records *per second* are indexed. Support of XML, including XPath® provides the ability to easily load multi-language and hierarchical XML documents.

Because the TeraText DBS was designed around the Z39.50 international standard for distributed search and retrieval, it offers an exceptional level of scalability that is exploited to advantage by the IASS – multiple servers, each running a separate instance of a TeraText database. The payoff is not only in scalability/performance, but also in fault tolerance and compartmentalization of data by date, source, or other criteria.

The TeraText® DBS also offers a very rich suite of searching capabilities, including fuzzy matching, proximity queries at the word, sentence, and paragraph levels, "Boolean" operators, relevance ranking, sorting, field-based queries, wildcards (truncation), stemming, term highlighting, and saving of result sets. Other product features include data compression, full Unicode support, API's in C, Java™, and Ace (TeraText's own object-oriented scripting language), field and record level security, and index scanning for data discovery. The software runs on Microsoft® before Windows®, Solaris® and Linux® operating systems.

XSLT engines (software and hardware) are used to perform just-in-time transformations of XML information into the format requested by the client application. XSLT re-purposes data for a variety of applications and audiences, such as management and the news media. XSLT also transforms XML into intermediate forms optimized for automated analysis.

In order to deal with thousands of user requests and queries, and to provide real-time performance, IASS makes use of hardware that transforms and formats documents. IASS draws on the high-speed performance of Data Power hardware in the sorting of every result set and the presentation of responses to every user request. The DataPower® XA-35 provides 10-50x increased performance in XSLT transformations, and integrates very well with industry-standard load-balancing software and hardware, delivering the scale required for enterprise systems. The XA-35 supports all W3C standards related to XML processing and has proven rugged and reliable. IASS uses the DataPower XA-35 in both proxy and co-processor modes.

And, taking advantage of the speed and accuracy of the DBS, and the XML standard format we have evolved the system to take advantage of this automation in a very important way. Working closely with analysts, we defined many of the *next steps* in the analysis process, and built into our middleware the sequential queries that ferret out the enriching data that informs the analyst. The analyst may ask for information on an individual during a particular date range and will get a meaningful response with summaries and easily "clickable" supporting data. Good as far as it goes.

Query:
assassinated Iraqi leader, May 2004-09-10

Results:
Suicide Bomb Kills Top Iraqi Official
Suicide Bomb Kills Top Iraqi Official Abdel-Zahraa Othman Was The Current Head Of The Iraq Governing Council May 17, 2004 7:12 am

Head of Governing Council Killed in Car Bombing
... A US soldier secures the site where a car bomb exploded in Baghdad and killed Abdel-Zahraa Othman. By Ramzi Haidar, AFP. May 17, 2004

Suicide Bomb Kills Iraqi Council Chief
... Abdel-Zahraa Othman, commonly known as Izzadine Saleem, was the second member of the US-appointed council assassinated so far. He ...
May 17, 2004

Figure 2

But, if that data proves to have value, then the analyst is almost certain to want to move in some well-defined directions asking for more ID information, associations, locations, etc.

- **Who** is Abdel-Zahraa Othman?
- Who are his known **associates**?
- What other **analysts** are tracking Abdel-Zahraa Othman?
- What **locations** are associated with Abdel-Zahraa Othman?
- What **reports** have been issued recently concerning Abdel-Zahraa Othman?
- etc.

Figure 3

We call this application the Fact Sheet. It institutionalizes, in the middleware and applications business logic, the steps that a senior analyst takes to assimilate and transform the data he or she finds in query results. In many ways, it is a recipe from the collection of "great chefs of information analysis."

Conclusion

The IASS performance requirements were mission-driven and have expanded dramatically in size, scope and speed. That drove us to find alternative solutions and a scalable, robust architecture. The system demands a rich query language and multilingual support. So, clearly, XML served as the best choice to structure, store, share and deliver information, while performance and flexibility were provided by

Hardware accelerators

Network Attached Storage, and

TeraText® DBS

The IASS has scaled up by two orders of magnitude over the last four years, without a hitch. Today, the IASS deals with increased volumes and uses, extremely heterogeneous data, providing analysts the answers they need – but we have gone **beyond** that to provide answers **before they ask!**

The technologies mentioned here – using XML for information encoding, transformation, assimilation and delivery – allow IASS developers to implement solutions that can help the analysts support decision-makers everywhere from the White House and Pentagon to the cockpit and foxhole.

Acknowledgements

This document was partially formatted for XML2004 using the DocBook service from **SchemaSoft.**