THE GILBANE REPORT

White Paper

XML Databases: An Idea Whose Time has Finally Come

The convergence of market forces, enterprise technology & computing standards has created a sustainable demand for XML-centric data stores

Sebastian Holst
June 1, 2003

Abstract

The demand for an efficient means to manage XML content is increasing proportionately with our increased dependence on XML-based applications, messaging and distributed computing. This white paper examines current market forces, available technology and the emerging XML landscape to gauge the value and, by extension, the demand for an XML-centric database that can sit alongside the many flavors of databases and repositories that are already standard issue in today's enterprise. While existing databases and repositories are adding various degrees of XML support, specific functional and resource constraints make it unlikely that any one of these data stores will be able to meet all of the needs of an XML *saturated* enterprise. This paper proposes a working definition of an XML-centric database and suggests specific applications where its value should be greatest.

SOFTWARE AG
THE XML COMPANY

XML Saturation

XML may be everywhere, but it is not doing everything it can

XML is the underlying language of the World Wide Web; there are hundreds of industry and special interest standards based upon XML; and there are thousands of software products that claim various levels of XML support. XML is truly everywhere. However, the full impact of XML has not been felt because, while XML may be everywhere, XML is not being used in all of the ways that it can and will.

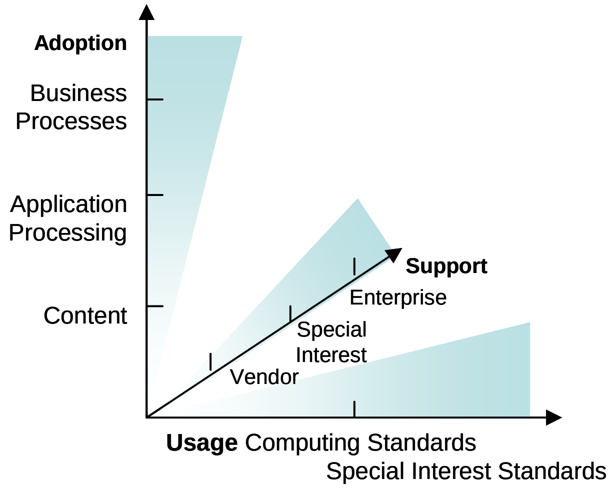


Figure 1: The three dimensions of XML saturation[1]

As XML usage approaches a saturation point within the enterprise and across the Internet, its true nature as a game changing catalyst is undeniable and unavoidable. How can XML saturation be measured? It can be measured in terms of usage, support and adoption.

Usage is a gauge of the number of XML standards and specifications being created and adopted. For example, how pervasive are XML standards in a particular software stack? Solutions architecture is continuously being transformed as developers monitor the ideas of groups like the W3C, which is transforming the architecture of the initial web with recommendations for its future development.

Support can be examined across three levels: vendor-specific support for XML, special interest support of XML, and enterprise support of XML. For example, what level of XML support has an enterprise achieved? Does the enterprise rely upon applications that support XML internally or only as import and export formats? Does B2B processing leverage XML?

Adoption refers to items like: what facets of an organization's business are modeled in XML? Have they moved beyond the traditional semi-structured content to include processing information and/or macro business process management? Are best practices, process management and other proprietary knowledge encoded within their XML content?

[1] Please refer to the Appendix for a more thorough discussion of XML saturation.

Enterprise infrastructure and B2B computing impact on XML saturation

XML saturation is being driven by numerous factors. Three of the most important factors today are the rise of the corporate portal, the emergence of web services as a cornerstone of the enterprise infrastructure stack, and the specialization and virtualization of enterprise content management.

If an organization is committing to and investing in any of these factors, the rate and extent of XML saturation will increase proportionately.

The enterprise portal

An enterprise portal assembles applications and collaborative workspaces within an enterprise web, indexes and organizes content, and rationalizes security and user information, all from multiple, distributed systems.

What is clear is that a portal has to interact with broad cross-sections of communities and applications. XML is the obvious catalyst to manage the many layers of interaction and content. As such, the portal is a major accelerating force in XML saturation. Figure 2 maps the immediate impact of a portal solution on an enterprise and indicates the likely ongoing impact as the portal is itself more fully assimilated into the day to day operations of an enterprise.

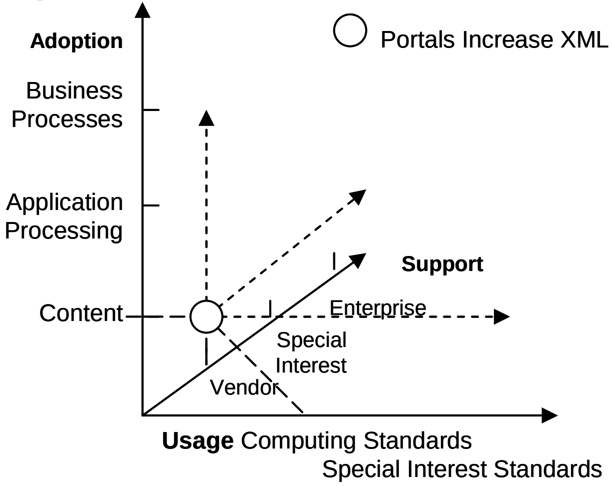


Figure 2: Portals are mapped into the three dimensions of XML saturation.

Portals provide immediate content-level *adoption*, with every vendor offering some level of XML *support using* a subset of the W3C recommendations. Further, portals have the potential to encapsulate business processes, provide enterprise level support and provide operational support for any number of special interest standards. If portals are a part of your strategy, you should also be planning for an increasingly XML saturated environment.

Every portal offering has some level of XML support with some vendors offering their entire portal solution as a bundle of web services.

Web Services and Distributed Computing

A Web service is a software system whose public interfaces are defined and described using XML. Other systems interact with Web services as prescribed by their definitions using XML based messages. In short, web services offer Internet protocol-based distributed computing. The web has transformed the user's experience and it will ultimately do the same for applications. As such, web services are a significant agent of change and are driving XML saturation. Like earlier distributed computing technologies that have preceded web services such as DCE/RPC and CORBA, web services imply a significant shift in development methodology, environments and skill sets[2]. As organizations assimilate web services into their computing infrastructure, they will see their saturation-level increase as illustrated in Figure 3.

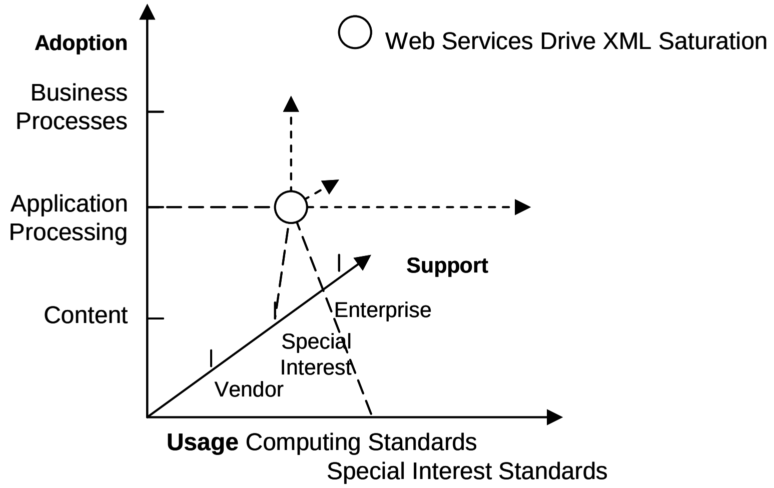


Figure 3: Web services are mapped into the three dimensions of XML saturation.

XML services provide application processing *adoption*, for special interest distributed XML *support using* W3C recommendations. Further, web services have the potential to encapsulate business processes, provide enterprise level support and form the basis for any number of special interest standards. In summary, if web services are a part of your deployment strategy, you should also be planning for an XML saturated environment.

Specialization and virtualization of content management

Today's enterprise is being overwhelmed by an exponential growth in business content. Growth is measured in volume, diversity and complexity. Today's business information is comprised of multiple data types managed across disparate repositories and databases and mixed with increasingly sophisticated forms of metadata.

It is only natural that content management and database management vendors have become equally diverse and specialized. However, this has created a whole new set of issues including;

- Sharing and transforming content across these systems
- Managing the complexity of a distributed, heterogeneous content management environment.

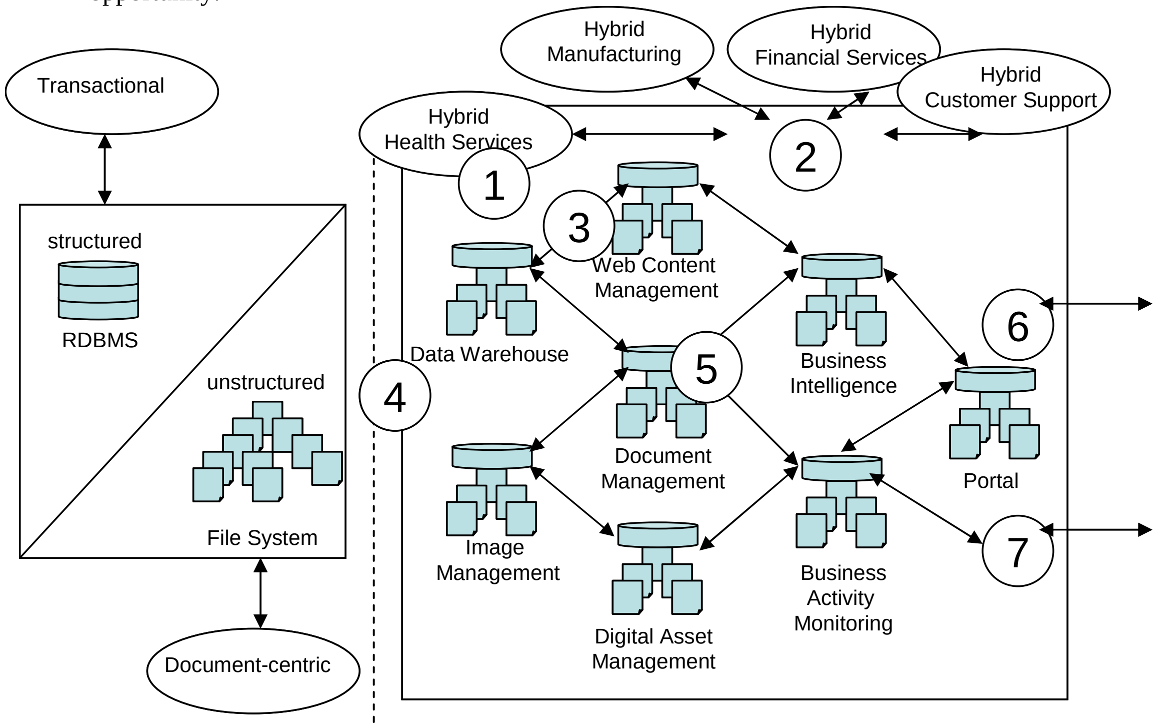
The pressure to create a virtual repository that integrates the diverse and distributed set of data stores, content, applications and users is extremely high and shows no sign of abating. Each content management supplier has begun jockeying to position themselves at the center of this increasingly distributed and heterogeneous environment, and enhanced XML support is the horse they ride in on.

[2] For a more complete discussion of the challenges and opportunities presented by web services, see The Gilbane Report, *Vol. 9, No. 8 October, 2001*, "Understanding Web Services".

How much XML support in a data store is enough?

Clearly XML has many roles to play and the incorporation of XML-based components within existing data stores, transport mechanisms and applications supports this premise. However, this also begs the question; is enhanced XML support inside non-XML-centric data stores enough?

Figure 4 illustrates the transformation from a simpler time, when content and applications were reasonably segregated into transactional and document-centric silos, into today's reality, where business, technology and content convergence has created as much confusion as it has opportunity.



Yesterday and Today: Database and Content Management

Figure 4: Diversification and virtualization of content and database management

The rectangle on the left side of the figure represents a reasonably segregated environment where database (mostly relational) managed structured business data and file systems were used to manage unstructured content (mostly business documents).

The various flavors of content management that have emerged are typically hybrid systems that use a DBMS to manage metadata, state information and file location and a managed file system to store media and unstructured content.

The specific hybrid applications at the top of the figure represent the most common application areas that have been developed on top of these hybrid content management systems. The numbered circles across Figure 4 indicate areas where XML data management is required.

1. **Persistent storage for applications whose processes rely upon sophisticated business documents and whose duration can last from minutes to months.**

Applications that build or process XML (forms, business documents, *etc.*) as a core part of their operations should consider an XML-centric data store as a runtime DBMS. It would be reasonable to expect improved performance, simplified code, a smaller footprint and increased functionality when compared to a non-XML-centric DBMS (relational or otherwise).

2. **A database of web services, taxonomies and other unifying components that create the virtual enterprise.**
   This is a special case of the previous applications. An XML data store could serve as the host for a UDDI registry, a universal metadata repository or some other XML-based resource providing a single access point to all other systems and services.
3. **A long-term cache or intermediary between other specialized repositories.**
   Distribution, replication and incremental update across repositories can be fortified by a high speed, lightweight XML data store that can be used to apply multiple transformations and store current network state information.
4. **A work in progress repository for XML authoring environments.**
   Authoring documents is typically an iterative process (as opposed to transactional) and increasingly, documents are being authored in native XML. Examples include web-based application development, electronic forms development, technical documentation and other highly regulated financial and health science documents. An XML-centric repository provides improved validation, search and reuse utilities, simpler administration, improved version control and difference detection.
5. **A repository managing business documents.**
   There are an increasing number of XML-based document types that are being used to drive a variety of supply chains and complex financial services. MISMO (mortgage), RixML (financial research), and papiNET (book manufacturers supply chain) are just a few examples of integrated services that will be driven by – and create – xml-based documents. A repository that was optimized to parse, transform and manage XML would likely increase stability, simplify development and improve performance.
6. **A repository for all portal-centric content.**
   An enterprise portal assembles applications, indexes and organizes content, and rationalizes security and user information, all from multiple, distributed systems. Staging content prior to distribution through the portal and storing taxonomies, templates, user profiles and other core XML-based information can be elegantly handled by an XML-centric data store.
7. **Staging repository.**
   There are other use cases beyond the portal where the aggregation and staging of information in native XML can be extremely valuable. Catalogue assembly, distribution to print, wireless and other specialized communication and reporting channels are all typically best served by an intermediate staging and transformation process.

Taken together, these use cases suggest a set of high-level XML data store requirements:

- Deep support for XML schema, query, transformation and hypermedia standards
- A lightweight footprint and performance profile that can be supported at each of the many touch points between applications, data stores and across enterprise boundaries
- Solid DBMS functions including concurrency, query processing, multilevel security, data validation and Atomicity, Consistency, Isolation and Durability (ACID)
- Optimized access into back office RDBMSs and other legacy databases of record.

The ubiquity of XML-based content and processing makes it unlikely that any non-XML-centric products have the flexibility and extensibility to support all of the important use cases. It clearly makes sense to consider the value of an XML-centric database as both an optimized data store for XML content and as an infrastructure appliance that acts as a stabilizing and integrating force across the enterprise.

XML Databases: A Practical Definition

Defining an XML-centric DBMS in terms of the requirements it must meet establishes criteria for selection and success

The Essential Characteristics of an XML-centric DBMS

Given the tremendous early success of XML-based applications, it is clearly not an absolute requirement that an XML-centric DBMS be at the heart of every solution. Having said that, are there specific use cases or scenarios where an XML-centric DBMS makes a material difference in cost, functionality and robustness? The following characteristics offer a working definition of an XML database and highlight the benefits of being XML-centric.

The XML-centric DBMS is a true DBMS

A DBMS is a collection of programs that provide a logical view of a collection of information that is independent of its physical storage. A DBMS provides update, access, search, security, concurrency, integrity, high availability and centralized administration to other programs and users entirely through its logical view.

The XML document must be the underlying organizing structure

An XML document includes very specific notions of elements, their order (organization), embedded content and descriptive attributes. Examples of information models that use an XML document as it underlying structure include the XPath data model, the XML Infoset, and the models implied by the DOM and the events in SAX 1.0. If the underlying data model does not have inherent support for these constructs or has additional support for unrelated constructs, there will either be an impedance mismatch (an incompatibility) requiring additional programming or excessive underlying logic that increases the footprint and the DBMSs stability.

All database functionality must be organized around the XML document

If a DBMS provides update, access, search, security, *etc.*, then these functions should be able to take full advantage of XML constructs. Examples include XML schema, namespaces (schema), XQuery, XPath (query language), direct .NET XML classes, JDOM, SOAP, XML:DB, SAX (programming interfaces) and specialized indexing and access control that accounts for the special organization of collections of XML documents.

Performance and resource requirements must be optimized to support XML-centric functionality

Support for additional data definition languages, data models or any other extraneous functionality will force compromises in performance, footprint, quality and/or cost.

Architecture and programmatic access must provide simplified integration with transformation and transport utilities

The transformation and transport requirements are significant enough in their own right that one can expect to see specialized XML buses/pipes that leverage XSLT and other technologies to move and transform XML documents at high speeds between repositories, DBMSs and delivery channels. Ideally, the XML-centric DBMS will come pre-integrated with such utilities.

Architecture and API's must support trusted access into RDBMS and other resident databases and repositories

In order for an XML-centric database to fulfill its promise as an "infrastructure appliance that acts as a stabilizing and integrating force across the enterprise," it must have built in read and write access into all major database management systems and repositories as well as an API that supports custom and site specific integration.

Packaging and bindings must support re-branding, runtime images and wrappers that can effectively submerge the DBMS inside visible applications and services

XML-centric database systems are not end-user applications or complete business solutions – they are best conceived of as critical components or appliances that reduce custom development, improve scalability and increase performance. As such, these appliances need to be packaged and distributed in such a way as to be invisible to the end user. Whether embedded inside a commercial product, or integrated into enterprise infrastructure, an XML-centric database must be "*felt but not seen*. "

Considering the functional capabilities and packaging requirements outlined here, who are the ideal consumers?

Ideal XML Database Consumers

The ideal customer for an XML database is one who is skilled at assembling components into a whole that is greater than the sum of its parts.
There are essentially three categories of customer who are most likely to benefit from an XML-centric database:

- **IT Savvy Enterprises** who are competent in architecting and building large hybrid commercial solutions
- **Application and Solution Providers** who add value higher in the solution stack but require powerful XML capabilities
- **System integrators** whose value proposition is based on providing integrated business solutions and who want to avoid building complex components that do not directly map to site-specific business needs.

A word of caution; often the ideal XML database consumer is tempted to build their own XML data store or extend another non-XML commercial product to provide XML support.. While XML databases are clearly distinct from today's RDBMS systems, they are no less sophisticated. No sane enterprise or technology provider would choose to build an RDBMS rather than license one (other than RDBMS vendors); do not build your own XML database (or support for XML into a non-XML database) unless it is actually your business to do so.

Additional Commercial Requirements

The XML family of Recommendations from the W3C, the raft of industry and special interest standards, and the best practices that they enable continue to mature and grow. As such, a credible and sustained commitment to implementing new XML-related developments is essential if the XML-centric database is to fulfill its promise as an optimized data store for XML content and as a stabilizing and integrating force across the XML saturated enterprise.

Summary

The XML-centric database will become a standard building block inside XML saturated enterprises and applications

XML is a game changing technology and portals, web services, XML forms and CMS evolution are driving that change

XML has clearly emerged as the dominant (meta) language of choice for data interchange, messaging, metadata modeling, linking and annotation. It is likely to be equally as dominant in the areas of information modeling and management and that is having a profound impact on the evolution of database and content management technology. As organizations move to portals and XML-based forms to enhance their business processes, the volume of XML and the need for specialized XML storage beyond a centralized DBMS increases dramatically. Further, the increased reliance on web services to power the enterprise infrastructure and the diversification of specialized content management systems is also heightening the sense of urgency to manage and manipulate XML content throughout the enterprise stack.

Extending XML support in existing database technology is necessary, not sufficient

RDBMS and content management vendors are all extending their XML support. Some have developed alternate views into their products that are entirely based upon XML recommendations for schema and query support. While this is certainly valuable and validates the necessity to provide true DBMS support for XML content beyond data type support, it is not sufficient in every case.

The ubiquity of XML in an XML saturated enterprise requires that the XML-centric DBMS support be available on the desktop, at the edge of the network, inside business applications and other areas where the large, all purpose enterprise data store is either overkill or, more likely, too expensive in terms of resource consumption[3].

Building and maintaining an XML-centric database is a sophisticated and expensive endeavor

The requirements for an effective XML-centric data store dictate that all of the essential capabilities of a full blown DBMS system be made available. The trimming of this XML appliance needs to be accomplished through its singular focus on XML and not by eliminating functionality such as scalable query processing, trusted security, scalability, *etc.*

As such, building an XML-centric DBMS is not for the faint of heart. It requires a significant investment of resources over a relatively lengthy period of time. Further, the evolution of the XML family of Recommendations and standards requires a long term commitment to maintain and extend the XML database.

If ever there was a case to buy versus build – this is it.

[3] If current licensing models and terms prevail, they are likely too expensive as well.

The XML-centric database must add scalability, improve reliability and free resources to focus on end-user functionality

Recall that we took the route of defining an XML-centric database in terms of the requirements rather than simply providing a technical description[4]. This was deliberate and intended to stress that it is the need to meet those requirements that are essential and "whose time has finally come." It will certainly be possible – even likely – that there will be XML databases that will meet the letter of any technical definition but will fail to meet its spirit; to serve as both an optimized data store for XML content and as an infrastructure appliance that acts as a stabilizing and integrating force across the enterprise.

When evaluating candidates for an XML-centric database, ensure that it embodies the eight essential characteristics. A viable XML-centric DBMS should:

- Be a true database management system.
- Use the XML document as the underlying organizing structure.
- Organize database functionality around the XML document.
- Optimize performance and resource allocation to support XML-centric functionality.
- Must provide simplified integration with transformation and transport utilities
- Support trusted access into RDBMS and other resident databases and repositories
- Support re-branding, runtime images and wrappers that can effectively submerge the DBMS inside visible applications and services.

It remains to be seen how these requirements will be met. While we suggest that existing database and repository products cannot simply be inserted everywhere persistent XML content management is required, that does not preclude the large and well-resourced vendors of these products from expanding their offerings with a new XML-centric DBMS. One might expect organizations that are dedicated to XML-centric product development to have the advantage of domain expertise, responsiveness and time to market. However, this will ultimately be settled account-by-account in the trenches of IT strategy and procurement. Regardless of who the commercial victors may be, we believe that the time has come for broad support and adoption of XML-centric database technology.

[4] For a good technical definition and discussion, visit http://www.xmldb.org/faqs.html#faq-1

Appendix

XML Saturation

This section introduces three dimensions of XML saturation and explores the current and emerging impact that XML saturation is having on general computing, business practices, and especially, content and database management.

XML saturation should be measured in terms of usage, support and adoption

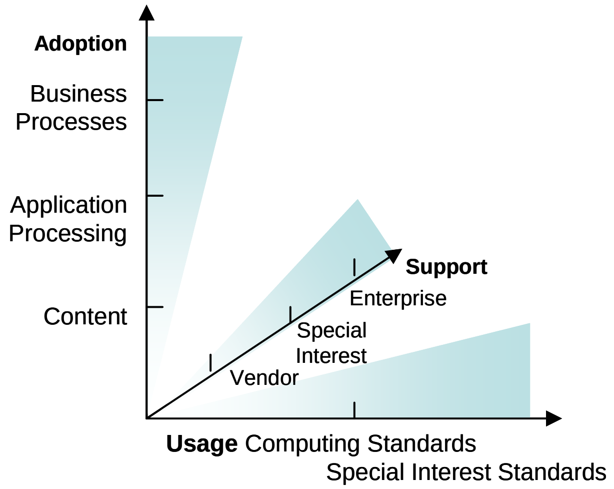


Figure 1 (repeated): The three dimensions of XML saturation

Usage: XML Specifications

There are two major classifications of public specifications: computing and special interest. General XML computing standards are, for the most part, administered by the World Wide Web Consortium (www.w3c.org). The W3C publishes “Recommendations” – not “standards.” XML is defined and published as a Recommendation by the W3C. “The W3C has published more than forty Recommendations since its inception. Each Recommendation not only builds on the previous, but is designed so that it may be integrated with future specifications as well. The W3C is transforming the architecture of the initial Web (essentially HTML, URIs, and HTTP) into the architecture of tomorrow's Web, built atop the solid foundation provided by XML.[5]”

[5] Source: About the World Wide Web Consortium (W3C) at http://www.w3.org/Consortium/#web-design

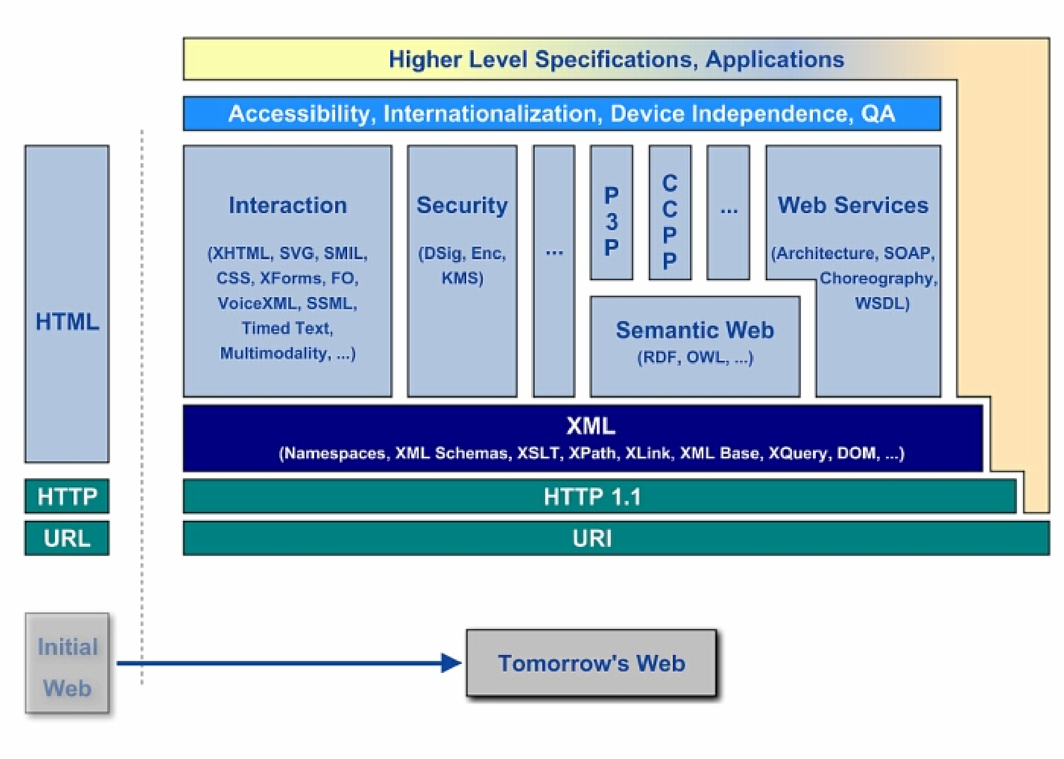

Higher Level Specifications, Applications
Accessibility, Internationalization, Device Independence, QA
HTML
Interaction
(XHTML, SVG, SMIL, CSS, XForms, FO, VoiceXML, SSML, Timed Text, Multimodality, ...)
Security
(DSig, Enc, KMS)
...
P3P
CCPP
...
Web Services
(Architecture, SOAP, Choreography, WSDL)
Semantic Web
(RDF, OWL, ...)
XML
(Namespaces, XML Schemas, XSLT, XPath, XLink, XML Base, XQuery, DOM, ...)
HTTP
HTTP 1.1
URL
URI
Initial Web
Tomorrow's Web

Figure 5: A mapping of the "Initial Web" to "Tomorrow's Web" as seen by the W3C[6]

One measure of XML saturation is the extent to which a particular software stack has incorporated and assimilated the components of W3C's evolving "Web of Tomorrow."

Special interest specifications that are also built atop the solid foundation of XML are governed by industry groups, *e.g*,. International Swaps and Derivatives Association, special interest computing associations, *e.g.*, IDEAlliance, and government agencies, *e.g.*, NIST. These standards have varying degrees of certification and are far too numerous to account for here. However, as an illustration, the Interactive Financial eXchange Forum (IFX) was formed in 1997 to create a messaging standard for financial services that would address the challenges faced with the advent of network-based computing models.

Currently IFX provides specifications for XML-based messaging in the areas of:

- Electronic Bill Presentation and Payment
- Business to Business Payments
- Business to Business Banking (such as balance and transaction reporting, remittance information)
- Automated Teller Machine communications
- Consumer to Business Payments
- Consumer to Business Banking

[6] *Ibid.*

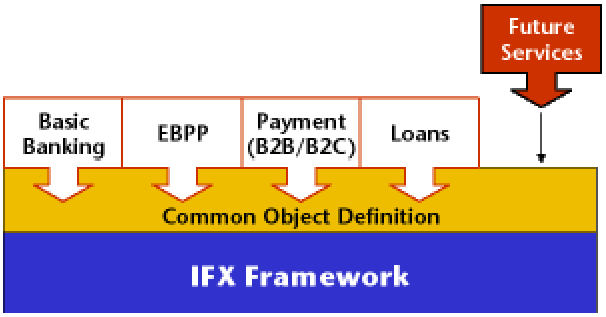


Figure 6: A representation of the Interactive Financial eXchange (IFX) architecture[7]

Another measure of XML saturation is the extent to which an enterprise and its trading partners have incorporated relevant industry standards into their operations.

Support: XML Scope

XML, like every information modeling language that has come before, has three degrees of support: vendor-specific, special interest group, and enterprise. These different levels of support have marked differences in value.

	Vendor-Specific	**Special Interest**	**Enterprise**
Definition	XML DTD/Schema(s) defined and used within a single product.	XML DTD/Schema(s) defined and used across a community with common interests	XML DTD/Schema(s) defined and used across an enterprise capturing closely held subject matter and process expertise
Examples/Participants	Microsoft Channel Definition Format (CDF) is a standard way to describe a Web site channel. Microsoft IE 4+ users can use this format.	papiNet covers transactions between parties within the Paper Supply Chain. papiNet is intended to provide value to all partners in the Paper Supply Chain.	Any enterprise that takes the time to model their content and processes in an integrated fashion using XML.
Characteristics	Simplifies interaction with a specific application	Facilitates interoperability and streamlines processes. The slowest to develop and be adopted. Can be seen as the least common denominator.	Encodes best practices. This results in increased organizational performance, improved quality of service and reduced costs.
Requirements	Vendor support	A consortium that can achieve and sustain consensus	Organizational commitment and ability to transform and "down sample" enterprise schema for external trading partner consumption
Relative value	Extremely limited	Has clear value over a relatively long period of time.	**Highest value realized in a relatively short period of time.**

Table 1: Distinctions and relative value of different uses of XML

Another measure of XML saturation is the level of XML support that an enterprise has achieved: using an application that supports XML internally, participating in B2B processing leveraging XML, and/or encoding best practices across their operations.

[7] Source: IFX Standard Overview at http://www.ifxforum.org/ifxforum.org/standards/index.cfm

Adoption: XML and semantics

XML tags have no inherent semantic definitions – what this means is that there is no predefined meaning for any XML tag. <para>, <employee> and <price> are simply human readable text strings. It is the application (or human reader) that assigns semantic values, *e.g,*. a context, implied meaning and corresponding processes. As a result, XML content can represent distinct semantic categories, each of which signifies a greater degree of XML saturation.

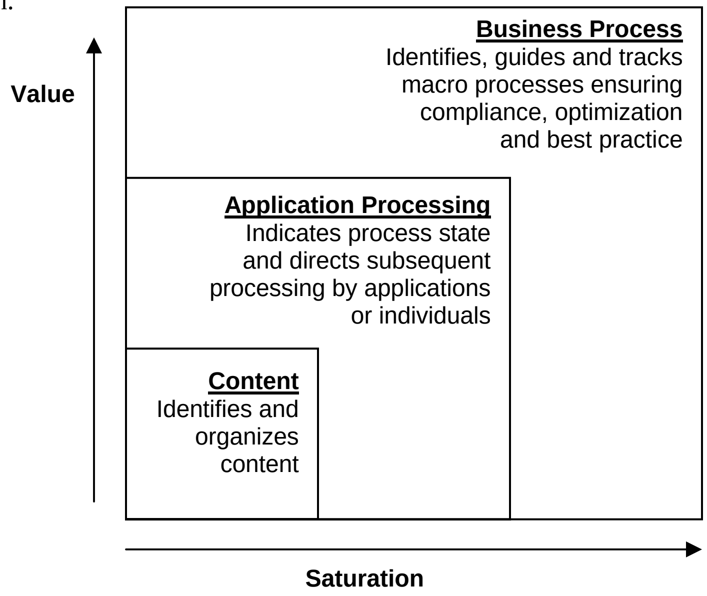

Value
Business Process
Identifies, guides and tracks macro processes ensuring compliance, optimization and best practice
Application Processing
Indicates process state and directs subsequent processing by applications or individuals
Content
Identifies and organizes content
Saturation

Figure 7: Three degrees of semantics that can be applied to XML content

Another measure of XML saturation is the nature of XML adoption. What facets of an organization's business are modeled in XML? Have they moved beyond the traditional semi-structured content to include processing information and/or macro business process management?

Software AG

Software AG is a technology company whose corporate mission and product value propositions all flow from a commitment to providing deep XML support. They have a long tradition of developing XML-centric products and are arguably one of the most successful independent software vendors in the XML solutions space.

Software AG's product portfolio features three segments: Tamino XML Server, an XML-centric database (Software AG uses the term XML Server where we use XML-centric database); EntireX, an XML transformation and transport layer; and Adabas and Natural, A hierarchical DBMS optimized for high performance transaction requirements and a 4GL programming language.

The Tamino XML Server is built to

- efficiently store XML documents natively
- expose information residing in various external XML or non-XML sources (legacy data) or applications to the outside world in XML format, and
- search effectively on the information Tamino has access to.

EntireX is integration software that integrates Web applications with packaged, custom-developed and/or legacy systems. It has two product components:

- *EntireX Communicator* provides simultaneous request and reply messages while managing interactions for XML-enabled and non-XML-enabled systems.
- *EntireX XML Mediator* is an XML document exchange tool that manages the flow and behavior of XML-based information. EntireX XML Mediator automates processing and routing of XML documents by applying rules based on the content and/or structure of the document. It supports web services standards including SOAP, UDDI and WSDL.

Tamino XML Server and EntireX are integrated XML-centric product offerings that are often deployed together in a service-oriented architecture. EntireX provides a messaging layer for XML and non-XML data to be routed, processed, and transformed. EntireX works with 250+ adapters to connect to mainframes, legacy applications, enterprise applications, and relational databases. Tamino can be used to manage, index, and store the XML data to support auditing and versioning of XML documents.

Software AG is a strong XML software and service provider and should be considered by any enterprise or software developer who is looking for XML-centric products and solutions.

Sponsoring Company:

SOFTWARE AG
THE XML COMPANY

For more information, please visit http://www.softwareag.com or contact:

Software AG Worldwide Headquarters
Uhlandstr. 12
64297 Darmstadt
Germany
Phone: +49 6151 92-0
Fax: + 49 6151 92-1191
E-mail: webinfo@softwareag.com

Software AG, Inc Headquarters
11190 Sunrise Valley Drive
Reston, VA 20191
Phone: +1 703-860-5050
Fax: + 1 703-391-6975