

# TREETIPS

Issue #28 • May, 2000

A Publication of Treehouse Software • [www.treehouse.com](http://www.treehouse.com)

## Your ~~XML~~ Company

**In a new Partnership with Software AG, Treehouse will bring Electronic B2B Tools to U.S. Customers**

Software AG has recently established a U.S. presence and plans to aggressively market **Tamino** and its full suite of native XML tools for B2B solutions. Therefore, Software AG USA (a wholly-owned subsidiary of Software AG, Darmstadt, Germany) and Treehouse Software, Inc. (TSI) have entered into a distribution partnership whereby TSI will market, sell and support Software AG, Inc.'s native XML e-business products in the United States. TSI will immediately begin selling **Tamino**, **Bolero**, and **EntireX**.

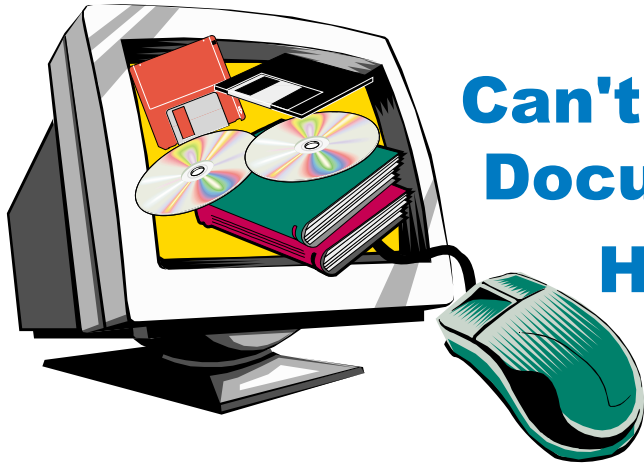
*"It is our goal to bring our XML technologies to the market through strong partnerships," Dr. Helmut Wilke, President and CEO, Software AG, Inc., said. "We are clearly pleased to partner with Treehouse to help us deliver innovative B2B solutions to our customers."*



## Inside This Issue

Partnership with Software AG .....	1	A Look at Bolero .....	4
Editor's Sproutings/Frontline .....	2	A Look at EntireX .....	5
A Look at Tamino .....	3	Introduction to XML .....	6

# Editor's Sproutings by Joseph Brady



## Can't Find those Documents?

## How about using Frontline?

**We have a special offer on Frontline for all current TSI customers. Call Bob Liptak at TSI for more information.**


There is a new product available from TSI called the **Frontline Team Server™** which can help businesses of any size coordinate enterprise-wide information through a Web-based work site that serves as a virtual library.

When the sales representatives from **Inmedius**, the developers of the product visited our offices to sell us **Frontline**, we were so impressed that we decided not only to become a **Frontline** customer, but also a distributor. In fact, **Frontline** is at the top of the list of the many products that we have been asked to represent, because we see the universal appeal of this product and how easily it can be put to use.

Whether your employees are in different offices, away on business, or working at home, they can contribute, share, and retrieve information in the **Frontline** document repository from any computer with a Web browser.

And since **Frontline** is so easy to use, even non-technical staff can contribute to the maintenance of the library through an ordinary web browser, such as Netscape Navigator or Microsoft Internet Explorer.

With **Frontline**, team members can automatically collect, store, and distribute information. Everyone within the enterprise is empowered to contribute to a common information source, with team leaders and subject matter experts managing their own content. This distributed responsibility ensures immediate results.

Contact TSI to learn more about **Frontline**. 

## TREETIPS

Editing, Writing,  
Design, and Layout  
Joseph Brady

### Contributors

Marni Harbutz, Kara Phillips,  
Dan Vimont, Ed Wolfe

### Production and Distribution

Terri Hammerschmitt  
and Carolyn Henning

Back issues available upon request.  
Documentation for all products is  
available in hard copy or  
on CD-ROM.

Hard Copy Circulation: 22,000



## TSI Reveals Newly Designed Logo

You may have noticed a new look for the TSI logo in this issue of TREETIPS. Gone is the tree with the 5.5" floppy trunk that has identified TSI for the past 10 years. We decided that with all of the recent changes, including the Software AG partnership, marketing new products, and the beginning of a new century, it is time for a change.

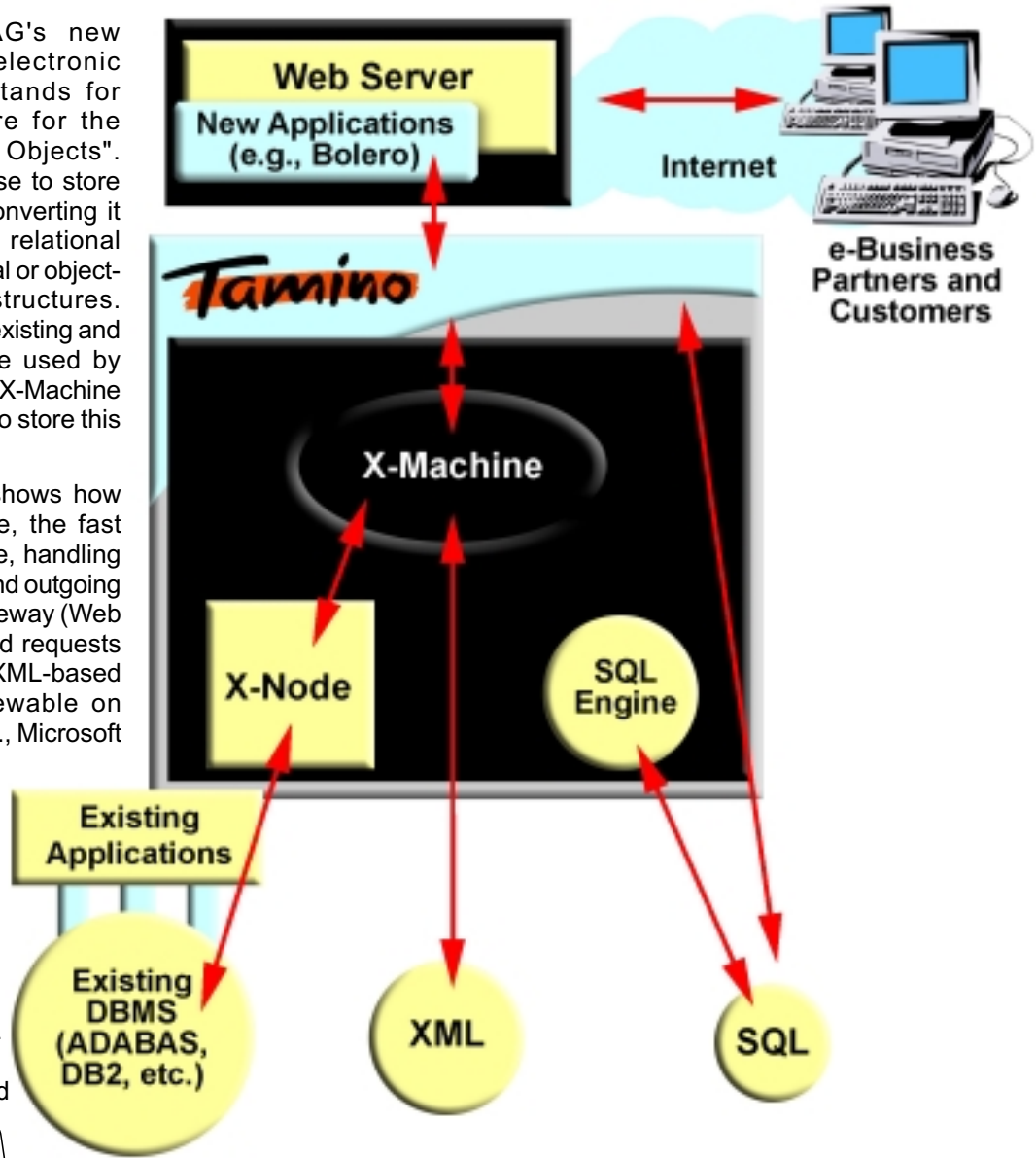
After many variations of a new design for the TSI logo, it was decided that the "flying red circle" design has the look we desire to carry us into the next century.

# A Look at Tamino:

## The World's First Native XML Database

**Tamino** is Software AG's new information server for electronic business. The name stands for "Transaction Architecture for the Management of Internet Objects". **Tamino** is the only database to store XML information without converting it to other formats, such as relational tables. It integrates relational or object-oriented data into XML structures. **Tamino** can access data in existing and remote databases that are used by existing applications. Using X-Machine technology, **Tamino** can also store this data as XML objects.

The diagram to the right shows how **Tamino** utilizes X-Machine, the fast storage and retrieval engine, handling large volumes of incoming and outgoing data via an active HTTP gateway (Web server). Valid URL-encoded requests are echoed by appropriate XML-based response documents, viewable on XML-capable browsers (e.g., Microsoft Internet Explorer 5.0 and Netscape 6). The X-Node module opens the enterprise up for XML technology and the Internet. It allows the use of data from existing databases that are left at their original location. In addition, an internal SQL database and an SQL interface are included, enabling **Tamino** to store and retrieve relational data.



### Native XML Storage

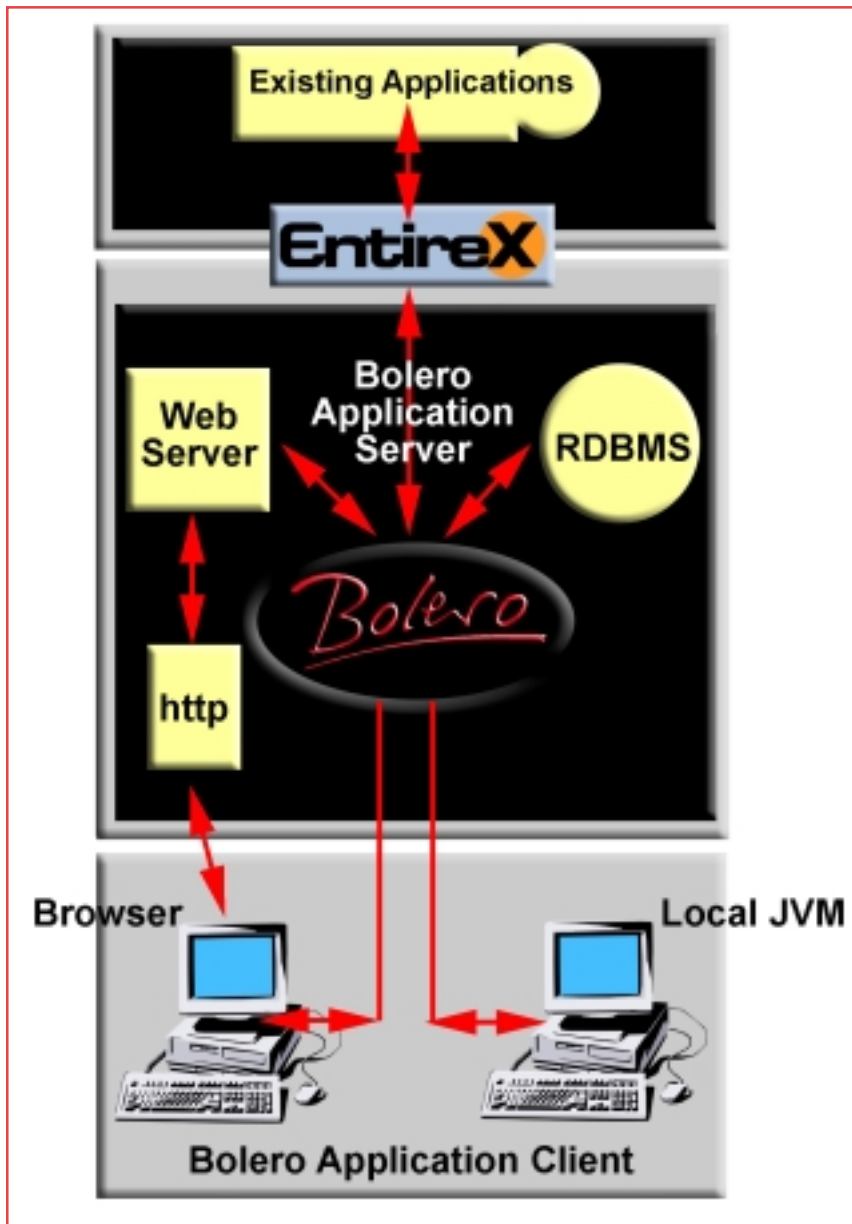
Relational database management systems (RDBMS) may appear to be a possible choice to facilitate the exchange of XML objects. Unfortunately, the table-based data model of RDBMSs does not suit the hierarchical and interconnected nature of XML objects. An RDBMS would need to break an XML document into a multitude of independent tables. A query against this database would result in many relational retrieval and join operations, requiring high processing power to overcome a considerable degradation of performance. In addition, RDBMSs as well as more advanced DBMSs, such as multi-dimensional relational databases or object-oriented databases, cannot handle data of dynamic structure, which is the key to XML's extensibility. An XML database must

be able to store and retrieve any well-formed XML document, even if the DTD (Document Type Description) of the document is not available. An RDBMS, however, requires schema definitions for each table, so a document with an unknown tag would require a change request for a new schema definition to be built and approved before it can be put into production. Likewise, in an object-oriented database a new class definition would be necessary - too time consuming for Internet applications.

Native XML storage is the essential method to avoid these performance limitations that are a crucial factor in the evolving high-speed world of electronic business via the Internet.

# A Look at Bolero:

## The Application Platform for e-Business



**Bolero** is an object-oriented platform for implementing and deploying mission-critical commercial applications in multi-client and multi-tier environments such as the Internet and the World Wide Web. **Bolero** is fully interoperable with Java, and **Bolero** object code is 100% pure Java bytecode. The same **Bolero** component will run on any platform for which a certified Java Virtual Machine is available and can cooperate with other distributed **Bolero** and non-**Bolero** components.

### Component-based Development and Integration

**Bolero** is designed for the implementation of component-based commercial applications. The Component Studio provides seamless integration with the Java and DCOM worlds, so you can import Java classes and COM objects and at the flick of a switch generate JavaBeans and COM objects out of **Bolero** classes. CORBA is supported through the Java API.

### Web Integration

**Bolero** integrates with Web operations through its ability to create applets to run in Web browser clients. This also allows **Bolero** components to be called from browser scripting languages like JavaScript and VBScript. In addition, **Bolero** components can interface with popular Web servers to allow backend processing.

### Key features in Bolero include:

- Business-oriented programming
- Persistent object interface to relational databases
- Transaction control for business objects
- Integration with other applications through DCOM, JavaBeans, and CORBA
- Modeling of business processes using Long Transactions
- Rapid Application Development (RAD) using Design Patterns
- Repository for **Bolero** components with project team support
- Powerful tools for building graphical user interfaces and reports
- Seamless Java integration

# A Look at EntireX:

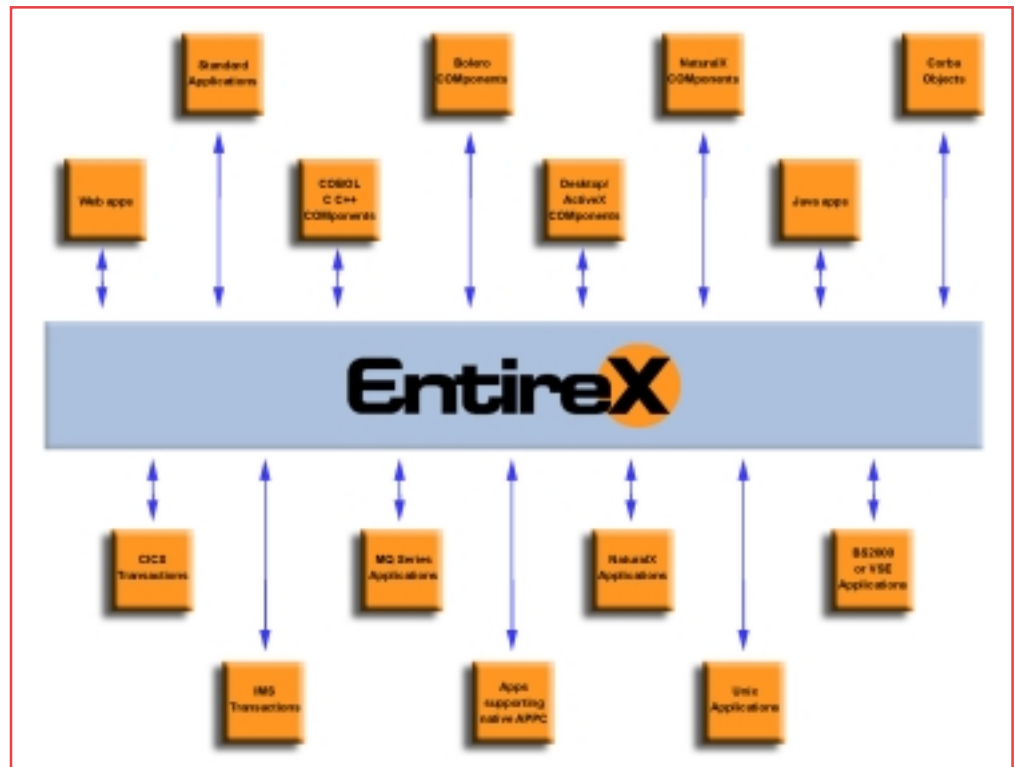
## Componentware for Enterprise Application Integration

Many organizations have a large investment in existing applications and data that simply cannot be transferred cost effectively (i.e., legacy applications), but must be integrated with, and accessed by, new applications residing on new, remote platforms (e.g., Web access). These organizations will want to deploy application systems and data across different platforms in the most efficient manner.

**EntireX** is a highly flexible and easy-to-use component-based solution for integrating enterprise applications across heterogeneous operating systems. **EntireX** offers application integration at high speed – both in terms of implementation time and performance. Strong DCOM (Microsoft's Distributed Component Model) capabilities make **EntireX** the product of choice when connecting NT to mainframe applications. By connecting previously disparate worlds, **EntireX** enables enterprises to lay the foundation for a seamless implementation of electronic business applications.

**EntireX** works something like a hardware bus, adding new functions to a system by means of extender cards. Every application that is "plugged into" **EntireX** represents a functional expansion. The main purpose of a bus system is to provide a common means of communication between the individual components. This is exactly what **EntireX** does, forming a bridge between applications. It doesn't matter how old or how cutting-edge an application is, what hardware it runs on, or whether it was originally designed to communicate with other applications.

Any application connected to the **EntireX** bus can communicate with any other application connected to **EntireX**. A rich set of interfaces and tools is provided to allow applications running on many IT platforms, written in a broad variety of programming languages, such as COBOL, NATURAL, C/C++, Visual Basic, Java, etc., to "plug into" the EntireX bus.



### The Components of EntireX

**EntireX Broker** is a very flexible, high-performance message server, the centerpiece of the **EntireX** bus.

**EntireX Broker Services** features MQSeries Adapter, APPC Adapter, CICS 3270 Bridge Adapter, and Attach Service.

**EntireX Security** provides protection for applications utilizing Broker and Broker Services.

**EntireX Developers Kit** comprises a rich set of interfaces and tools to attach applications written in a variety of programming languages to the **EntireX** bus.

**EntireX Manager** provides functionality for monitoring and managing the **EntireX** environment.

**EntireX DCOM** uniquely provides DCOM for all leading UNIX platforms and the OS/390 mainframe.

# XML: Animal, Vegetable, or Mineral?

by Dan Vimont

We'll just keep this explanation short and sweet (and just disregard the fact that some XML purists might regard the following explanations as *too simplistic*).

**XML** stands for “**eXtensible Markup Language**”, and it actually is *not* a language in itself, but a relatively simple (yes, *simple*, believe it or not) set of *rules* for creating a specific breed of *new* markup languages, intended to bring a lot more structure and manageability to all the documents of the Internet. This set of rules has been declared by the grand poobahs of the Internet, the **World Wide Web Consortium** (commonly referred to as “the **W3C**” – the same good folks who brought us the standards upon which much of the Internet is based), and thus is a truly universal and open standard.

**Okay, so what's a “markup language”?**  
**In fact, what is “markup”?**

Well, any document that adheres to the rules of a “markup language” is made up of two big components: “Markup” and “Content”. “Content” is rather self-explanatory – the content of a document is the raw information that is to be communicated. “Markup” is simply everything else.

For an example, here's a perfectly well-formed XML document, complete with markup and content:

```
<PhoneNbr>
<AreaCode>412</AreaCode>
<LocalNbr>741-1677</LocalNbr>
</PhoneNbr>
```

As you can see here, there's a lot more markup than content. The two pieces of content (“412” and “741-1677”) are surrounded by a bunch of markup. Each of the pieces of markup in the above document (each piece delimited by pointy brackets) is referred to as a “tag”. The whole point of tag usage in XML is to clearly identify the content that comes between the tags.

**Tag, you're it**

Believe it or not, XML doesn't really get much more complicated than that. Well, okay, it gets a *little* more complicated. In the example above, we have implicitly invented a new (XML-based) markup language, which we might call “Phone number markup language” (or PNML). We followed the XML rules of the game and decided that a valid PNML document should consist of an element called “PhoneNbr”, and that this element will

The XML Web standard enables the pharmacist to have instant access to data from related organizations.

## Exchanging Information Between Different Systems

Because XML gives a standard format for information related to a specific subject, it can be used to simplify the exchange of information between different sources. Many kinds of applications have, or will have, standard XML coding. Chip manufacturers, and many other industries already have standard XML, and more will follow. This means that **systems can use common XML languages to exchange information with each other, regardless of their internal format.** The main applications of this will be the **exchange of data between companies in the same industry** or researchers within an academic field, although many other applications for ordinary users are imaginable.

There is already an XML standard for chemists, called CML. CML will be very useful for exchanging research results and other data between chemists and companies working with chemistry in any way. It can also be used with Java-applets in education. So, despite having data on an uncountable variety of operating systems, and located at just as many global locations, XML provides a common language that disregards these barriers. The list of possibilities goes on and on.

have inside of it two other nested elements: “AreaCode” and “LocalNbr”.

Now, how do we announce to the world that this new markup language, “PNML”, exists and what constitutes a valid “PhoneNbr” document? We simply make a “document type definition” (DTD) available to explicitly spell out the rules of our PNML game (i.e., to tell a person or a machine how *(continued on page 7)*

# XML: Animal, Vegetable, or Mineral?

(continued from page 6)

they may construct a valid "PhoneNbr" document).

Here is the DTD that we would make available to the world of phone number enthusiasts:

```
<!ELEMENT PhoneNbr (AreaCode, LocalNbr)>
<!ELEMENT AreaCode (#PCDATA)>
<!ELEMENT LocalNbr (#PCDATA)>
```

where #PCDATA means Parsed Character DATA.

That's all!! Adhering to the XML rules of the game, we've defined a new markup language. The problem is that this is a really stupid example. Nobody in the real world would come up with such a boneheaded and limited markup language.

## So, what about the real world?

Lots of folks in the real world *are* currently inventing much more complicated and useful markup languages. But, they will follow the same rules of the game that were adhered to in this example.

Why are industry consortiums and other groups getting on the XML bandwagon and establishing their own industry-standard DTDs? Initially, it seems that two big needs are most immediately addressable by adoption of appropriate XML technologies: (1) complex document management and (2) business-to-business data exchange.


We are not trying to educate the world about XML, but only introduce the topic. We are trying to tell the world that the voluminous XML data they're soon to experience had better be stored efficiently and accessed quickly. That's where **Tamino** comes in.

Software AG has the lead on this type of database technology with **Tamino**. The "wrapper effect" posed by other database vendors is their cosmetic solution which they'll soon realize won't cut it. We hope our readers will get in front of the XML wave and call us about **Tamino** (see page 3).

## Find out more about XML

The TSI Web site ([www.treehouse.com](http://www.treehouse.com)) has full information on **Tamino** and a link to Software AG's XML resource page where you can find more in-depth information on XML.

Several interesting books on XML are readily available at most book stores, such as **XML for Dummies**, **XML a Primer**, **Applied XML**, and **XML a Manager's Guide**.

Another great Web resource is the **XQL FAQ**, a Frequently Asked Questions list for the XML Query Language (XQL is a query language that uses XML as a data model). The address is: <http://metalab.unc.edu/xql>. This site is managed by **Jonathan Robie**, R&D Fellow, Software AG. 

Many major industry groups have already developed, or are in the process of developing, XML standards as you can see from the following list taken from a presentation given at TSI by Software AG:

- Channel Definition Format (CDF)
- Web Collections using XML
- Meta Content Framework Using XML (MCF)
- Resource Description Framework (RDF)
- The Australia New Zealand Land Information Council (ANZLIC) - Metadata
- XML Metadata Interchange Format (XMI)
- Educom Instructional Management Systems Project (IMS)
- Structured Graph Format (SGF)
- Web Standards Project (WSP)
- Open Software Description Format (OSD)
- Extensible Log Format (XLF)
- Wireless Markup Language Specification (WAP)
- HTTP Distribution and Replication Protocol (DRP)
- Chemical Markup Language (CML)
- Bioinformatic Sequence Markup Language (BSML)
- Virtual Hyperglossary (VHG)
- Weather Observation Definition Format (OMF)
- Open Financial Exchange (OFX)
- Open Trading Protocol (OTP)
- FIXML - A Markup Language for the FIX Application Message Layer
- Signed Document Markup Language (SDML)
- Bank Internet Payment System (BIPS)
- Real Estate DTD Design (OpenMLS)
- XML for the Automotive Industry - SAE J2008
- Mathematical Markup Language (CML)
- OpenTag Markup
- Metadata - PICS
- Synchronized Multimedia Integration Language (SMIL)
- Precision Graphics Markup Language (PGML)
- Vector Markup Language (VML)
- Web Interface Definition Language (WIDL)
- Electronic Data Interchange (XMLEDI)
- Information and Content Exchange (ICE)
- CommerceNet Industry Initiative
- eCo Framework Project and Working Group
- vCard Electronic Business Card
- Telecommunications Interchange Markup (TIM, TCIF/IPI)
- Encoded Archival Description (EAD)
- UML eXchange Format (UXF)
- Translation Memory eXchange (TMX)
- DMTF Common Information Model (CIM)
- Process Interchange Format XML (PIF-XML)
- Ontology and Conceptual Knowledge Markup Languages
- LACITO Projet Archiving de données linguistiques sonores et textuelles  
[Linguistic Data Archiving Project]
- Astronomical Markup Language (AML)
- Genealogical Data in XML (GedML)
- Newspaper Association of America (NAA) - Classified Ads Format
- Notes Flat File Format (NFF)
- Cold Fusion Markup Language (CFML)
- Document Content Description for XML (DCD)
- XSchema
- IETF "Extensions for Distributed Authoring and Versioning on the World Wide Web (WEBDAV)
- Tutorial Markup Language (TML)
- Development Markup Language (DML)
- Java Speech Markup Language (JSML)
- Virtual Reality Modeling Language (VRML)
- XML for Workflow Management [NIST]
- Simple Workflow Access Protocol (SWAP)
- Theological Markup Language (ThML)
- XML for FAX (XML-F)
- Extensible Forms Description Language (XFDL)
- Broadcast Hypertext Markup Language (BHTML)
- Open Settlement Protocol (OSP)
- Web Distributed Data Exchange (WDDX)
- Common Business Library (CBL)
- Schema for Object-oriented XML (SOX)
- Voice Markup Language (VoxML)
- XML Transfer Protocol (XMLTP.Org)

If you're involved in one or more of the industry groups above, you'll soon be affected by new XML standards. If you're not involved with any of these industry groups, please call us so that we can inform the W3C to investigate this oddity.

All product names mentioned in TREETIPS are trademarks and/or products of their respective holders. The mention of any non-TSI product, or information provided by outside sources in TREETIPS should not be considered to imply support or endorsement by Treehouse Software, Inc., its employees, or affiliates.

# TREEHOUSE



## Treehouse Software products include:

### Electronic B2B:

**Tamino** - Electronic business information server. It is the first information server to store XML information without converting it into other data structures

**Bolero** - High-level business-oriented application development environment capable of delivering distributed applications to the Java platform

**EntireX** - DCOM-based Component Bus, enables existing applications and Enterprise Resource Planning (ERP) systems to be integrated with Bolero applications

### Relational Products:

**tRelational** - ADABAS data analysis, relational modeling, and mapping tool; DPS parameter generator

**DPS** - ADABAS to RDBMS data materialization, replication, and propagation software

### Electronic Document Management:

**Frontline Team Server™** - Enterprise-wide document repository and virtual library

### UNIX Products:

**SEEDIT** - XEDIT and ISPF/PDF compatible editor for UNIX and Windows

**S/REXX** - REXX-compatible language for UNIX and Windows

**S/REXX Debugger** - Optional graphical debugger for S/REXX programs

### Software AG Related Products:

**ADAREORG** - File reorganization tool for ADABAS MVS

**ADASTRIP** - Data extraction utility for ADABAS

\* **AUDITRE** - Generalized ADABAS auditing facility

\* **AUTOLOADER** - ADABAS file automatic unload/reload/dump utility

\* **CHART for NATURAL** - NATURAL application analysis and documentation tool

**DBAUDIT** - Data integrity verification utility for ADABAS

**GENADA** - Re-engineering tool to extract, analyze, compare, and manipulate ADABAS data in databases and sequential files

**GENART for NATURAL** - Tool to extract, analyze, remedy, and document NATURAL code

**GENTEST** - Automated testing/scripting tool for ADABAS and NATURAL

**NATURAL Engineer** - Re-engineering tool for upgrading existing NATURAL applications, such as for Web-enablement

\* **N<sub>2</sub>O** - NATURAL application change management system

\* **N<sub>2</sub>O/3GL** - 3GL support within N<sub>2</sub>O for PANVALET, LIBRARIAN, ENDEVOR, and PDSs

**PEEK** - ADABAS file browsing utility

\* **PROFILER for NATURAL** - NATURAL quality assurance and testing tool

**QDUMP** - Incremental backup utility for ADABAS

**RACE** - NATURAL performance enhancer and "Redundant ADABAS Call Eliminator"

\* **SECURITRE** - ADABAS and NATURAL security interface to RACF, ACF2, and TOP SECRET

\* **TRIM** - ADABAS and NATURAL performance monitor

\* Indicates TSI Products which are marketed for TSI by international affiliates

Phone: (412) 741•1677

Fax: (412) 741•7245

E-mail: [tsi@treehouse.com](mailto:tsi@treehouse.com)

Web: <http://www.treehouse.com>

## TREEHOUSE SOFTWARE, INC.

409 Broad Street, Suite 140

Sewickley, PA 15143 USA

