The Next BIG Thing (really!) — Computable Semantics



A MODULANT WHITE PAPER
JEFFREY T. POLLOCK
AUGUST 2001

MODULANT™
The Interoperability Company

TABLE OF CONTENTS

AFTER THE INTERNET — THE NEXT KILLER APP 2

WHERE IS THE INFORMATION IN INFORMATION TECHNOLOGY? 3

INFORMATION ALCHEMY (ABSTRACTION, METADATA, & ONTOLOGIES OH MY!) 4

THE SEMANTICALLY CONNECTED WORLD IN OUR LIFETIME 5

BUILDING HYPERBOLE WORLDWIDE 6

This document is protected by copyright and may not, in whole or in part, be copied, reproduced, translated, or reduced to any electronic medium or machine-readable form without prior written authorization of Modulant.

Modulant™, the Modulant logo, and Curtana™ are trademarks of Modulant Solutions, Inc. Other trademarks, service marks, trade names, and company logos referenced are the property of their respective owners.

RESTRICTED RIGHTS: Use, duplication, or disclosure by the U.S. Government is subject to restrictions of FAR 52.227-14(g) (6/87) and FAR 52.227-19 (6/87), or DFAR 252.227 7015(b) (6/95) and DFAR 227.7202-3(a).

Information in this document is subject to change without notice and does not represent a commitment on the part of Modulant. This document is provided “as is” and all express or implied conditions, representations, and warranties, including any implied warranty of merchantability, fitness for a particular purpose, or non-infringement, are disclaimed, except to the extent that such disclaimers are held to be legally invalid.

AFTER THE INTERNET — THE NEXT KILLER APP

The next big thing! Humph! We've all heard that before, right? What if I were to tell you that a semantic tsunami (say that 5 times fast!) was forming right now? Well, Tim Berners-Lee, father of the World-Wide-Web, would believe me! He thinks that the semantic web will become the next killer app. In the May 2001 issue of Scientific American he posits the possibility that the semantic web will become as big and important as the Internet itself. Yes, believe it or not, a giant tidal wave of computers and semantics is headed our way.

Berners-Lee is not alone in this idea either.

In fact, the World-Wide-Web Consortium (W3C), under the direction of Berners-Lee, has adopted the research and work of leading computer scientists from around the world. From the halls of MIT to the labs in universities at Stanford, Brussels, and Crete — the ideas and techniques for making semantics computable are falling into place each day.

The first waves of software companies to embrace computable semantics are emerging as well. San Francisco based Modulant Solutions, the German company Ontoprise, and Manchester based Network Inference are just some of the companies focused on delivering commercial software based on the principles of semantic interoperability.

Even venture capital has started to identify the huge growth potential in this new space. A director from one of the leading Menlo Park VC's, Draper Fisher Jurvetson, has gone to Red Herring with an article that proclaims a new type of software tagged 'metaware' — which is really a word to succinctly describe software that makes use of abstractions to collaborate more effectively — the semantic web.

Interestingly, the latest rage among the software giants — web services — is a complementary idea to the semantic web. But, it will not, by itself, solve the overarching problems of information sharing that the semantic web is tackling — because it is fundamentally about solving the physical connectivity and transportation issues.

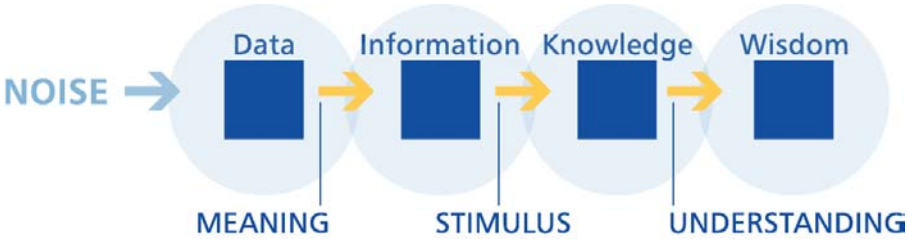
The idea of semantic interoperability is so important and valuable because all the data, in all the machines, all over the world, has meaning to somebody — but not necessarily to other machines or other people. The semantics of the data is locked away in where the software is, how an application uses it, and how human users interpret it. The overriding goal of semantic interoperability is to attach computable meaning to the data and turn otherwise meaningless data into sharable information.

WHERE IS THE INFORMATION IN INFORMATION TECHNOLOGY?

Information is that curious evolution of data as it begins to gain meaning. Of course, for most software applications the information is what is most useful — and the hardest to get. When you use SAP, PeopleSoft, or Microsoft Excel for that matter, the data is readily accessible, but the information often requires a step of interpretation. The data by itself doesn't tell you everything you need to know. Similarly, when you use web pages on the Internet the original meaning of the information is never explicitly defined in a computable way — and that's why web searches have such a low signal to noise ratio. Current technologies that attempt to share and query information fail because they only move around and evaluate the data, rather than getting at its meaning.

This may seem like a subtle distinction, and it is, but it is so important because of its impact on costs, quality, and utility of that information which you would want to share. When you move from data to information you are adding meaning, or semantics, to the data.

FIGURE 1



Currently the way we search for information on the Internet is by matching the key words (data) that you type into a text box. If you misspell a word, you get no hits — the engine does not know what you mean. We are lucky to have, and love, the better search engines that at least attempted to give us more relevant results based on what other users have historically selected, guessing at the meaning behind our data.

When companies attempt to share information between business systems they typically implement a technology that uses XML, programming objects, or databases — but the same problems exits. A computer cannot intelligently distinguish two XML tags <paragraph> and <idea> unless a programmer has told the application how to interpret the pattern of characters that make up the word 'i-d-e-a' (and every other tag that can appear). The same holds true for programming objects like CORBA or EJBs where the programmer has to tell every application what the object of type 'PurchaseOrder' means in advance. Fundamentally all of the current data sharing techniques operate on the syntax of what they share — not the intended meaning. In other words the focus is on how things are said, and not what is said.

XML, CORBA and other standards have received so much early hype because it was thought that they alone would facilitate meaningful communication between systems. Instead we have only seen the proliferation of incompatible DTDs and one-off CORBA implementations with no real decrease in the effort we have to expend to build simple intelligence into the software we write. We have simply distributed the same problem to new technologies.

It seems that the data that software operates on can only maintain its meaning in the particular context that it was intended to be used in. Web pages or business systems alike, we have to build very sophisticated point-to-point solutions to share information in a meaningful way.

But of course, you know that I'm going to tell you that there is a better way. Semantic interoperability of information between different sources is exactly the goal of the semantic web. The approach for solving these problems can be attacked in multiple ways, but they all share some common characteristics.

INFORMATION ALCHEMY (ABSTRACTION, METADATA, & ONTOLOGIES OH MY!)

In a word, the way that you build in semantics across different machines is "abstraction" Okay, okay, so what does that really mean? Fundamentally, making semantics computable depends on giving your application a reference point to understand what another application is trying to tell it.

The way that researcher's across the world are attacking this problem is through the use of ontologies. An ontology is simply a way of making an abstraction explicit. These have been used in philosophy for thousands of years as a way to talk about 'what we know.' But for computers to do anything useful with an ontology it must be made machine understandable. The easiest way to do this is to model one in XML Schema, UML or EXPRESS (a modeling language from the ISO community) so that a parsing algorithm can use it. You see, the form of the ontology is not nearly as important as its content.

Once you have a machine interpretable ontology that covers a given domain (telecommunications, product, health care, etc.) — by the way, this is not at all simple — you can start to map in how your local software applications 'fit' into this ontology. Then the local applications only have to be aware of the ontology when they collaborate, not the other systems that could be using their information. So, the business systems will interoperate, but not integrate — got it?

With unstructured data on the Internet, search engines could give you a hundred percent accurate responses if HTML programmers linked their

data to a universal ontology. Then, when you used Yahoo!, it would really know what you mean. You know what I mean?

All of this sounds kind of spooky right? The Internet knows what you mean, business systems interoperating but not integrating — didn't artificial intelligence die back in the 80's? Well, AI is really that dirty little secret that underlies all this newfangled technology driving the semantic web, but not everybody will admit that.

As early as 1975, Marvin Minsky and other AI experts were focused in on discovering new ways to represent knowledge in digital form. Well they failed back then at making the technology widespread, but twenty-five years later we've got the computing horsepower to finally deploy semantic-based toolsets to a very wide audience.

The other way that people are attacking this problem is through the idea of meta-metadata. What? Okay, so it's not really meta-metadata (data about data about data?) — it's more like using environmental metadata: interesting data points that tell you things about the software application, what processes it is used in, what business domain it operates in, etc. Sometimes you will hear this referred to as "context." When you think about it, it makes sense. What do you mean when you say to somebody, "you took my quote out of context?" Well, it's the same thing that happens to software when a message is received with no context.

Some of the toolsets being developed at universities have the idea of a "context mediator." A context mediator is a software component that mediates requests coming from different contexts in order to ensure that the proper data is sent back in the reply. So context is not traditional metadata that you'd find in a database, it is a set of high-level metadata that describes software in its business context.

Okay, fine — so maybe you believe me at this point; and maybe you don't. I at least assume that you will check out the links provided at the end of this article to check out the references to semantics, ontology, and context to be sure that all this is not rubbish. Besides, it would probably take several books to do all these ideas justice! But let's assume that all of these ideas and technologies could be put together in such a way to make semantics computable — how would that change things for you and me?

THE SEMANTICALLY CONNECTED WORLD IN OUR LIFETIME

Imagine a place where (I've always wanted to write that!) the business systems talk to each other, PDAs and the Internet — where the Internet knows what you mean when you ask it questions — where the words "legacy system" are meaningless because old computer systems never get old. Imagine a place where you tell a computer program to find

something for you, then it lives for months searching across millions of computers worldwide; in different languages. Now, imagine a place where you get all of this with just a few keystrokes and no coding!

Okay, admittedly all of this is probably decades away. But realistically we're only months away from seeing parts of this utopian vision fall into place. There's rapid development taking place in three hot areas that are set to converge: intelligent agents, web services, and computable semantics. These three technologies, when used together, will change the way we think about computing.

Systems will be able to be self-configurable. Based on common reference points (ontologies) they will be able to learn (via mappings) about new information systems. Intelligent agents will traverse information schemas in multiple languages and report back results in native text to their users. Business systems will use technologies like UDDI, wsXML, and SOAP to look up services in a variety of systems — and then establish rich, meaningful communication via computable semantics.

In the short term, we'll see the first baby steps of the semantic web as RDF and XML are used together to describe content. Ontologies will become commonplace as the semantic web matures and business systems make use of "metaware" — moving complexity to a higher level of abstraction. By late 2001 more business interoperability platforms will be in place to complement and compete against the entrenched EAI, ETL, and B2Bi companies.

Companies will be attracted to this new model because it will mean huge cost savings and the ability to respond with more agility to changes in information technology. In all, the long term potential of these technologies is sky high, we'll just have to wait and see who delivers on the promise of that technology.

BUILDING HYPERBOLE WORLDWIDE

Thankfully there is quite a bit of activity to deliver on that promise. Universities all over the world are participating in coordinated, as well as uncoordinated work in the areas of computable semantics. New business models are popping up every day founded on the idea of computable semantics. No doubt we'll see and hear a lot of hyperbole surrounding the topics in trade magazines, popular journals, and even news programs. Like XML, and Java before it, it will be difficult to sort out the fact from the fiction — but in the end we're all going to benefit from it. My tact right now is to remain cautiously optimistic about the nature and timing of the wave that is forming. But, for me, the question is when, not if, it will come crashing to shore.

INTERESTING LINKS

SEMANTIC WEB
http://www.w3c.org/2001/sw/
http://www.semanticweb.org/

COMPANIES MAKING SEMANTICS COMPUTABLE
http://www.modulant.com
http://www.ontoprise.com/
http://www.networkinference.com

RESEARCH ON SEMANTICS, ONTOLOGY & CONTEXT
http://suo.ieee.org/
http://www.ontoweb.org/
http://context2.mit.edu/coin/
http://www-db.stanford.edu/Ontoagents/
http://ksl.stanford.edu/researchthem.shtml

ARTICLES ON SEMANTIC COMPUTING
http://www.scientificamerican.com/2001/0501issue/0501berners-lee.html
http://www.redherring.com/story_redirect.asp?layout=story_generic&doc_id=RH630019863
http://www.xml.com/pub/a/2001/03/21/timbl.html

Jeffrey T. Pollock is the Vice President of Technology Strategy at San Francisco-based Modulant Solutions. He is responsible for the technology vision and direction of Modulant's semantics-based middleware platform. Jeff has successfully architected, designed, and built application server and middleware solutions for dozens of Fortune 500 companies. Prior to Modulant, Jeff managed IT projects as a Principal Engineer with Modem Media and worked on leading edge technology for several years at Ernst and Young's Center for Technology Enablement. Jeff has lectured at JavaOne and San Francisco State University, and he also instructs courses at the University of California at Berkeley Extension on the topics of software engineering, software development processes, J2EE service architecture, object-oriented analysis and development, and large-scale software architecture.



425 Brannan Street, Suite 200
San Francisco, CA 94107
Tel: 415.281.4500
Fax: 415.537.3550
info@modulant.com
www.modulant.com