37 A FOLDING SYNTAX FOR SEMANTIC MODELING

H. T. Goranson
Sirius-Beta, tedg@sirius-beta.com
USA

Enterprise integration is traditionally a matter of engineering to a framework. The virtual enterprise drives the engineering focus more to the semantics side of the equation. In our lab, we are dealing with a few experimental types of semantics for the virtual enterprise. Previous papers have indicated three of these: tactile semantic distance, introspective modeling and folded metaphors from film.

The need for advanced semantics plus the need for collaborative creation and management have forced us to re-examine model presentation for the virtual enterprise. In the past, if formal these have been based on graphical notations or sentential logic, often both. It is common to have annotative text associated with the formal structure.

This paper presents some ideas adapted from other communities and presents some ideas for an alternative presentation. The primary metaphor is nesting, and the motive is to conform in a natural way to XML formats. An outline presentation scheme is sketched and emerging collaborative editing methods adopted.

1. THE PROBLEMS

Our group works with modeling issues related to virtual enterprises. Our definition of what comprises a virtual enterprise is a fluid, opportunistic, presumably temporary aggregation of small businesses and individuals who are collectively addressing or seeking manufacturing opportunities. Our virtual enterprises often task partners to do things they don't ordinarily do in unfamiliar business sectors. Our enterprises are highly dynamic, and constantly evolving, meaning that optimization simulations are continuously run in parallel with operations. No central authority dictates, adjudicates or stores practices. Any process in any partner can trigger change in another process in another partner without central direction. The integration is intimate and proactive. No prestandardization occurs, nor any harmonization of business practices; indeed many of the partners never worked with (or even knew) the others.

This differs in both difficulty and promise from the notions of most other research groups. As a result we are identifying problems that may not be apparent in less ambitious virtual enterprises, like preharmonized bidding consortia.

1.1 Operational Dynamism

Enterprise modeling originated with a need to make processes explicit so that they could be comprehended and engineered. A later need, coincident with the appearance of enterprise integration, added the function of enterprise management. But still, in that current context, nearly all of the actual operational actions are taken outside of the modeling framework.

Virtual enterprises are different: there is no pre-existing operational infrastructure that can rumble on, parallel to the additional functions of modeling. That means that whatever models the virtual enterprise uses, they must do triple duty:

— they are a means for describing and revealing processes to partners or an assembly agent, and in this role often are referenced by contract documents and related metrics.

— they are a means, together with externally referenced procedures, for actually running the virtual enterprise, filling the role of operational infrastructure. In this role, the "models" become executable code.

— they are a means for re-engineering and optimizing the enterprise, perhaps by improving partners individually but also by making changes that could only be revealed via a system—level view. This might be seen as an extension of the first duty: after all, the original impetus for modeling was for process improvement. But functionally it is resourced as an extension of the second duty: enterprise management is a matter of both operating and improving the enterprise.

Those of us who do this in a practical context know that just managing the two roles is difficult enough, but to do that in the context of the third role is exceedingly challenging and unique to the virtual enterprise case. The key to this is the ability to explicitly capture the semantics of introspection. We'll return to that later.

1.2 Speculative, Federated Structure

Modelers like to change the world to be well-behaved. When they can, they'll force harmony in as many dimensions as possible: they'll use one modeling method, one presentation syntax, one ontology, one notion of business culture and ethics and they will perform all this harmonization (and settle all the models) well before actual work begins.

None of these things are true in the high payoff cases we encounter. Diversity is the rule (has been proven in fact to be a competitive advantage). As a result, many methods, ontologies and local cosmologies must be tolerated, even leveraged.

Usually work starts before many processes are well modeled or understood, so a way of discovering models by doing work is required. Agility is the name of the game, by which we mean extreme appropriate adaptability (rather than mere speed). Simulations of alternative strategies and configurations need to be hosted in parallel with real operation, using the same models.

This specific problem, also unique to the virtual enterprise case, is being addressed by the notion of "semantic distance," which we reported in an earlier Pro-VE. The basic notion is to have a semantics and related tools to represent certain aspects and assumptions about unknowns in models ranging from the misunderstood to the missing (data or process).

But how to represent this semantics?

1.3 Distributed, Low Level Comprehension

Virtual enterprises of the type we encounter are characterized by lean operation: the staffing and emphasis is on the expertise or core competency in question. Legal, enterprise engineering and managerial functions are slighted, certainly relative to the staid large organizations with which they compete.

It is also the case that decisionmaking and situational awareness is pressed to the lowest level possible, simultaneously "flattening" organizational charts.

These two reasons mean that whatever the sophistication and depth of the models, no matter how complex the introspection and distance (both being esoteric concepts that most professional modelers wouldn't comprehend), the results must be understandable by an ordinary partner. That translates into being understandable by an average human with a high school education but essentially no training in modeling, abstract mathematics or computer science.

In essence, they need a syntax as simple as a spreadsheet.

In the thirty-five years I have been doing modeling, the community has addressed every problem by ever-increasing sophistication in the theory and tools used. Real enterprise models are rarely used outside of large enterprises for the simple fact that only a large operation can afford to double-staff a process: one person who understands the process and another who understands the models and metrics associated with it.

Every year the community pays lip service to this need and every year it goes unaddressed (excepting tools that lack a formal basis). The virtual enterprise case forces the issue.

In the previous Pro-VE, we described a rather promising approach to this problem, the leveraging of sophisticated introspective abstractions from the movie world. This paper describes another approach, one more amenable to existing desktop computer tools. We will not discuss the combination of the two here for simplicity's sake.

Incidentally, at the same time that we have the comprehension problem, we have the need for models and their associated instances to be reportable in a number of formats for humans and machines. Practically, this means we need XML structure. What we propose here takes advantage of this structure visually.

2. THE OUTLINE VIEW

The modeling syntaxes we use today — graphic and sentential — originated in a print environment. There are good reasons for this. We need training and documentation material which is usually printed, and the sketching, exchange and archiving of models is often paper-based in ordinary or large enterprises. The virtual enterprise shifts the case from paper-based to desktop computer-based as the models and assembly of models is developed collaboratively, and the "use" of them is in a shared, networked environment.

Outlining, as it turns out, is possibly the first and probably the most mature paradigm for a display technique that is not tied to paper. Actually, there is a paper-based origin, but the conventions we mean here (collapsing, disclosure triangles,

indented children, cloning, hypertext, transclusion and folding) exist only on a screen. Outlining is a way of allowing a focus on a single element, like a process, but also expanding the whole collection so that the entire system can be viewed, each element in a larger context.

The outlining paradigm goes back to coding tools at the MIT AI lab in the 1970's, but outlining came to the desktop on the Macintosh when it appeared in 1984. Since then, all the advances in this area have been on the Mac and the most robust user and developer communities are Mac-based.

Outlining has been added to the 100-year old Gantt chart format in a manner that has been adopted in essentially every project management software package.

Our expectation is that the outlining paradigm can be adapted as the basis for a basic virtual enterprise modeling tool that:

— can be widely supported on the cheapest hardware by a large number of software vendors

— will be accessible to someone in text mode who has essentially no special training in modeling

— but at the same time have the ability to be strictly formally based and have conventional graphic syntaxes

— be integratable into existing small business project management packages.

3. OUR ENGAGEMENT IN THE COMMUNITY

We started this project eighteen months ago. Our strategy was to engage a critical mass of the user and developer community, and having done that develop a set of principles and standards that can be leveraged by the developers. Our goal is to be as universally influential as fast as possible.

Our primary vehicle for this a monthly on-line column our project writes. It is part of an on-line "magazine" that focuses on Macintosh software. The periodical is free and can be read either on-line in html format or downloaded in screen or print-optimized PDF formats. Previous issues are well-indexed and persist on-line.

Our column on outlining (Goranson, 2005) gets a couple hundred thousand readers each month and about fifty readers a day thereafter. Though all the examples are Mac-based, it has quickly become the center of activity and theorizing about that specific interface convention for users of all desktop platforms. The columns have provided a definitive history of the paradigm and set an agenda for development and use. That agenda has been profoundly influential in the developer community by creating a market pull of educated users. Significant capability suggested by the column has appeared in more than a dozen widely varying outliner-based products in recent months.

Outlining is all about structure, so is almost universally employed in XML editors. We intend to open the collaborative modeling paradigm to our identified ontological needs for advanced virtual enterprise modeling. As it happens, outlining has historically been at the center of web-centric XML standards: XML-RPC, RSS, SOAP and OPML all came from the outlining world first, then found wider applications (especially SOAP and RSS) in the larger internet world.

OPML is an outline-specific XML format for the exchange of outline content (Winer, 2001).

We have initiated an effort to develop the replacement for OPML; because of the critical mass of developers involved we expect to be able to deliver the results in a very short time, together with compliant applications. We expect to have robust commercial and free products available to virtual enterprise users from this community in a short a time as a year.

4. THE SEMANTIC ISSUES

Some key elements to the new standard are typed links, semantic distance and attribute ontologies.

We keep the common notion that the primary hierarchy is used to describe the work breakdown according to existing conventions in the work breakdown and workflow communities. These breakdowns have legal legacies, associated contract law and accepted performance metrics. Secondary hierarchies are associated with human organizations and other resources, risks, costs and liabilities as also commonly used in the industry. We extend these as required for the virtual enterprise case, especially to cover transitional states where control over processes is distributed among several partners with different liabilities and rewards.

We also keep and extend the prevailing ontological infrastructures in the various enterprise functions. As is consistent in enterprise integration, we tag these to process ontologies. In the case of outlines, these appear as attributes, which express as XML types.

What is new is the notion of typed links. We type links among outline elements two ways. The first is a relationship link, for example a process or resource dependency for a process is linked to that process (and back) with the appropriate type link. These display as hyperlinks of different colors.

The second notion of type relates to the ontology employed in the outline element. We find all enterprises employ multiple ontologies, and we want to capture the "semantic distance" (Goranson, 2004) between the semantics of elements. In a typical enterprise, this might be the mismatch between how production and financial managers evaluate the quality of a given process. In a virtual enterprise, we find the ontological mismatches are amplified by different ontologies applied by similar actors over similar elements.

An international workshop was held in 2003 to identify the types of distance in a virtual enterprise, and these are the notions captured in the linkage types as alternatively a scalar or a geometric fit. The unsophisticated user usually chooses to see the semantic distance as a literal distance among elements in a map view of the outline elements.

5. CONCLUSION

The virtual enterprise has amplified needs for collaborative enterprise modeling. Such enterprises also have different, difficult needs. One of these needs is for many simple desktop modeling, project management and information collection tools. The basic structure of the outlining paradigm, existing tools, our role in the community and the possibility of a virtual enterprise-centric OPML 2 (or whatever it is called) may be the best way to address these needs.

6. REFERENCES

1. Goranson, HT, About This Particular Outliner series <http://www.atpm.com/Back/atpo.shtml>, 2003-2005.
2. Goranson, HT, "Semantic Distance," Proceedings of ICEIMT 04, to appear. 2004.
3. Winer, Dave, OPML Specification <http://www.opml.org/>, 2001.