Programs at the Center are possible due to the support of its Sponsors:
Amdocs • Cable & Wireless • Deutsche Telekom • Fujitsu • IBM • SAP • Sun Microsystems

SUMMARY

On March 19, 2002, the second meeting of the ongoing Working Group on Dynamic Information Integration reconvened at the Center. The session was entitled "Models and Taxonomies" and examined how data is mapped across different platforms and integrated into and between applications and systems. Participating in this session were Attensity; Dejima, Inc.; iSpheres, Inc.; KnowMadic, Lake Forest Venture Management; Rod Heisterberg Associates; SAP; Strategies, Concepts & Solutions Labs; and WhizBang! Labs.

"It was a great discussion," said co-host Kaj van de Loo of SAP Markets. "The most important insight was that we cannot look at data models in isolation. An overall model has to take events, rules and processes into account -- the influences operating on the data."

The meeting opened with an introductory presentation by van de Loo showing how data is presently gathered, stored in and spread out across heterogeneous systems. He also posed the question of what models might help in integrating the data into business processes so that it can be used by workers across the enterprise as well as between enterprises. "How do we get to integrated data?" asked van de Loo. Currently data comes into an enterprise from departments such human resources, sales force, finance and strategic partnerships via various applications and programs. The data may be captured in purchase orders, job applications, service level agreements and many other sources in low-level systems. The challenge in today's enterprise is to integrate that data and turn it into information that becomes part of the knowledge store of the company and helps individuals to make better, more well-informed business decisions.

Mapping the Taxonomies
Participants discussed mapping between various versions of XML, from the back end to ERP software and from the back end to various translation software such as Rosetta Net[1] and possibly recruiting the aid of agent technology to assist in the chore.

Currently mapping of the back end, whether to translation software such as Rosetta Net[2] or to ERP software such as SAP, is done manually and can be very expensive for SAP's clients. Agents might be developed to look at the meta data and find common vocabularies, thereby reducing some of the complexity. One option is to set up the model, and require everyone to map to that particular model. However, new ways of looking at data and non-traditional ways of making decisions with non-traditional data are emerging. Called "decision re-engineering," these

---

[1]Rosetta Net is industry consortium to standardize "a master dictionary to define properties for products, partners, and business transactions." Have a look at http://xml.coverpages.org/rosettaNet.html.

[2] Ibid.

processes should be monitored and re-integrated into the ERP architecture. Rigid models will not allow for change and growth. Information integration must be part of the process. The challenge is to figure out and map the entire lexicon of experiences.
3

Participants said that the tools are not yet evolved well enough to bring the data into the models. "The big problem is still how to manage the models," said one participant. Could agents solve the problem by going through the meta data model to figure out the mapping? The difficulty is the semantics. However, it was suggested that if the process model is defined in great detail, the data will begin to sort itself out in the context of that process. Highly-detailed context can inform the semantics of the language. This process leads to rapid information integration into that particular model.

Another tool for looking at complexity and multiple taxonomies is through the use of more hardware to support more schemas. "Hardware is cheap compared to people," said one participant. Hardware can be used to maintain a variety of taxonomies and learn how they are used. This allows users to grow their own end-user applications. They define the needs and goals and are much more satisfied with the results. The group agreed that users are not happy with schemas that are too general, but it takes time for them to input the parameters and data for them to be satisfied with the results. And it can take lots of hardware.

Standards
"Standards don't solve everything, but they can reduce complexity."

"Good standards should not try to solve all problems at once, but should focus on well defined problems and be designed to be used as building blocks."

The group decided that there is a need for a common middle layer that would facilitate various XML-based applications talking with one another and allowing information from one department, for instance, to be used by another. Although multiple taxonomies are a reality, one way to begin to untangle the knot of many different sources trying to talk to one another is to map between various XML-based applications and create common terms. For instance, "First Name" or "Street Address" could be exactly the same in hrXML (Human Resources) as in sfXML (Sales Force) and also in financialXML. Another way of accomplishing the translation is to create an intermediate lingua to which everything translates, and then translate that back to a department language. For instance, hrXML could be translated to a common lingua and then translated to sfXML. But participants agreed that a great deal is lost in the translation to the common lingua, and that the most efficient and effective way to connect vocabularies would be through an XML standard taxonomy common to all XML applications. However, as one participant commented, "You cannot always wait for standards to be developed. What we need is a temporary, intermediate tool that allows us to share the work we have done." In this way, developers could agree on a certain vocabulary according to the work they have accomplished to date and move forward in development by sharing this tool.

"Standards are good and helpful and actually do solve problems," said van de Loo, "but it takes time for standards to evolve. In the meantime, early adopters need to work without standards. Also, a standard can never be complete, so additions and extensions are necessary. Because of this, you cannot rely on standards to solve all problems, but you need tools that can help you

solve the problems while benefiting from standards. Good standards should not try to solve all problems at once, but focus on well defined problems and be designed as building blocks."

Assisting Agents
Alerts and Unintended Consequences
The group discussed how software agents might be used to reduce the complexity of data and assist humans through alerts, visualizations and simulations. If the system has some idea of where the user is going and what they want to accomplish, software agents can assist the user in making decisions. One case study that was put forth involved financial portfolio management. Agents were used to alert the manager when the portfolios were out of balance, and then pointed out the unintended consequences of mitigating actions. For instance, if the portfolio contained Company A and Company B stock, and Company A bought Company B, the software alerted the manager that the portfolio was consequently out of balance with too much Company A stock. The option to sell part of the stock might bring up an alert that a stock sale would result in a tax penalty, while another option might trigger an alert of a stock-rule violation. Once the original goals of the human being are articulated, the software can assist. One participant noted that it is important to understand the original goal of the query. "If my current model is not rich enough yet, if the mapping is not done yet, it is important to recognize that these things are changing. If we write the goal in expressive enough language, it is almost like writing down the law so that when you come back later and look at it, you have the context for what you are doing, where you are going."