The Data Center

An Update for the M Language

David L. Brock, Robert A. Bryant, Brian Jacokes, Hyoung-Gon (Ken) Lee, A.J. Lisy, Peden P. Nichols, Hristo S. Paskov, Edmund W. Schuster

The Data Center, Massachusetts Institute of Technology, Building 35, Room 212, Cambridge, MA 02139-4307, USA

ABSTRACT

Formal research on building an interoperable data and modeling network began in March 2003. By the fall of the same year, a comprehensive framework for the M Language was put forth in a paper that became the base for another paper submitted to the 2004 Educators Conference sponsored by the Council of Supply Chain Management Professionals. This work won the Plowman Award given each year for best paper. In December 2004, SmartWorld was sponsored by the MIT Industrial Liaison Program to discuss the M Language. Several hundred people attended the event.

Since 2004, the M Language has undergone a number of different iterations. A version of M is at mlanguage.mit.edu. Though this is not an alpha version, it does represent an effort to demonstrate the ideas associated with interoperable data and modeling networks. There remains a great deal of work remaining to make the M Language a tool for business.

This report documents the efforts of several undergraduate computer science students to develop a system based on the principles of the M Language.

1.0 INTRODUCTION

At Smart World 2004,* Sunil Gupta of SAP paraphrased Samuel Taylor Coleridge by saying "data, data everywhere, but not a byte to use."

Each year, the amount of data grows by as much as 40 – 60% for many organizations. In 2004 alone, shipments of data storage devices equaled four times the space needed to store every word ever spoken during the entire course of human history.

Amidst the ever-increasing amounts of data, "…companies are struggling to figure out how to turn all those bits and bytes from a liability into a competitive advantage." Emerging technologies such as the EPCglobal Network and RFID technology, along with sensor networks and the use of "loyalty cards" are certain to generate even more data.

By one estimate, there will be 300 million RFID readers active in supply chains within ten years. In another prediction, "the number of deployed sensors will dwarf the number of personal computers by a thousand fold" in 2010." Both of these technologies will boost the amount of raw data available for supply chain management.

Dealing with the increasing volumes of structured and unstructured data will require new standards and information architectures to improve integration and communication between hardware, software, and business entities. This becomes important as companies seek to overcome the barriers that limit the seamless transfer of data, internal and external to the firm.

With this goal in mind, the MIT Data Center Program, situated within the MIT Laboratory for Manufacturing and Productivity, has spent almost five years developing an open system capable of integrating data and mathematical models into a computer network. Designed to work with existing standards such as XML, the MIT Data Center Program has developed a new computer language called "M" designed to address a number of semantic issues that currently limit the free exchange of data and mathematical web models across the Internet.

The M Language provides the ability to label physical objects and abstractions with an exact semantic that is machine understandable. In the case of abstractions such as data or mathematical models, the machine understandable semantic unique to M provides the capability to "self identify." This is of great importance for many business systems and is the base for creating a crude form of intelligence. The M Language approach handles context, an important aspect of creating intelligence, in an innovative way that will improve the productivity of information processing and change current thinking about manufacturing systems and supply chains.

After a semester of research work accomplished by a team of MIT undergraduate computer science majors, release of a version of the M Language occurred during July 2007. Since the idea of the M Language is to create an open system, all of the components of the M Language are free to use much like the current case of Wikipedia. The initial release is at mlanguage.mit.edu.

* Smart World 2004 was sponsored by the MIT Industrial Liaison Program

Looking at the big picture, the M Language is essentially a means of overcoming the semantic and syntactic issues that inhibit the usefulness of XML as a universal format for data. For example, when a message is formulated in computer-to-computer communications, the target must be identified. Then agreement must exist concerning semantics and syntax. This is a time consuming and expensive process that is subject to error because the context of words used to describe each data element must be shared between the sender and the target. The M Language provides a way to achieve machine-to-machine communication when the target is unknown.
While enhancing the ability to make data interoperable, the M Language also establishes a means of creating a network of mathematical models. To accomplish this task, M forms the equivalent of an artificial language capable of connecting data and mathematical models together.

This paper represents a step forward in practical implementation. It also includes comments about progress to date and future needs.

The next section provides more detail concerning the underlying theory of M, in addition to how it differs for other approaches.

2.0 M COMPARED TO XML

The core strength of XML, the ability for users to define arbitrary tags within the XML framework, is also the core difficulty. Because users are not bound to any predefined data representation, they must develop their own keys and terms to describe XML documents.

The M markup language was developed as a solution to data integration. At its heart, M is similar to XML – nearly any word can be used to describe data within the overall framework. However, several features of M are different as compared to XML. These differences position M as an innovative way to combine vast quantities of information in meaningful ways.

First, unlike the XML framework, M requires the user to select words from the existing M dictionary. XML, on the other hand, allows the freedom to use of any sequence of characters. Although this requirement imposes a slight limit on data description, it also means that each M word has a known definition. Furthermore, to prevent ambiguities among homonyms, each M word is followed by a period and then a number. For example, in M, the word 'tree' has 6 different 'M-words' – tree.1 is the plant, tree.2 represents a figure, such as an organizational tree, tree.3 corresponds to a shoetree, and so on. When a person describes data in M, they are required to specify the instance of the word they are using. Therefore, describing something simply as a 'tree' would result in an error; the user would need to choose one of the six tree M-words from the dictionary.

2.1 Semantic Associations

Using M words in XML enables a rich set of semantic associations, which greatly simplify the translation from one data schema to another. This capability does not exist in XML.

To use another example, take in-store checkout kiosks. In the future, these systems will rely on RFID tags embedded in products to specify the price, category, SKU, perhaps an age requirement, and more. Since different manufacturers are likely to use different data keys when describing their products, kiosks would be required to 'understand' many different data markup schemata. With potentially hundreds of different vendors supplying products in the average store, and more vendors releasing products every day, maintaining the database of markup tags and keeping it up-to-date is a formidable task.

However, if such a system used the semantic association properties of M, the process would be substantially easier. Rather than forcing manufacturers to create a schema that works with every type of kiosk software, they could instead use any system of markup tags they wanted as long as they described the data with M-words. This would create a form of machine understanding that product.1 name.1 is the same as product.1 brand.5. With such a system, updates would be far less frequent, and instead of requiring that a database be maintained with the complete schema for each brand, the kiosk would only need to make sure that the M-words had the proper associations, which is a much easier task.

2.2 Conversion to M

The benefits of describing data with M-words instead of standard English words in XML keys are obvious, as it enables the disambiguation and data association capabilities of M. However, companies are extremely invested in their current systems, and change, even one as seemingly trivial as adding a number behind each of the terms to specify which synonym is intended, is prohibitively difficult and expensive. If companies are to use the features of M there must be an easy process that allows them to convert their data into M.

Data conversion is another appealing facet of the M language. Since M is designed to be a central description ontology usable by virtually any field, once data from one source has been translated into M, it can be understood by a variety of users. Before M, if a company that owned many gas stations wanted to use the data from several of their suppliers, they would need to develop one-to-one converters to translate the data from the format that the refineries used into the format that the gas station company used. Assuming N gas station companies and K oil refineries, the industry would need to build (N ×K) converters to communicate.

This approach quickly becomes expensive and hinders the sharing of data to manage inventories. Since M is designed as a common description language, these companies could save significant amounts of time and money by using it as a go-between. Instead of each gas station company attempting to interface with each of the refineries, they would instead simply interface with M.

On the other side of the supply chain, the refineries could do the same, building a conduit that translated their data to M as well. This would significantly shrink the problem, from

many-to-many (N ×K) to many-to-one (N +K), thereby vastly simplifying collaboration and saving large amounts of time and money for the industry as a whole.

2.3 Web machines: A Public Interface to M

The power of M lies in bringing together different data sources and enhancing the ability for computer programs to bridge those sources. The core of this strength lies in the unique M-words and the rich semantic associations that link the M-words to each other. The use for such a system in corporations and other entities with huge data needs is clear.

However, there is another powerful facet to M – the capabilities in aggregating data from sources on the Internet. Unlike company databases, the data contained on the Internet is often not described by metadata. However, the Web 2.0 movement is gradually shifting focus to better machine-readable data sources, such as RSS and XML feeds that form the output of most web services. Unlike standard web pages, these special documents are formatted and tagged according to the XML standard and are designed to work with software that parses out the information and displays it in unique ways. Currently, many organizations are working on creative applications of these data sources. Projects such as Yahoo! Pipes have taken the first steps in combining data from multiple sources in unique and useful ways. However, such projects are currently very limited because developers aggregating web information need to make the same many-to-many associations that plague companies trying to combine business data. M represents the next generation of interactive data web machines, since it simplifies the associations and allows for easier and more "intelligent" combination of sources.

2.4 Anatomy of a Web Machine

At its core, a Web Machine is a mini-program designed within M to take inputs, process them, and return outputs. Within this classification are three types of core machines: left [input] edges, pure machines, and right [output] edges.

The left edges are designed to take a non-M data source, such as RSS, and convert it into valid M. The conversion process itself can be done through code or with the "Translator Factory," which is a tool that examines the existing tags and suggests M tags that may apply. The factory makes conversion straightforward and quick, which allows left edges to be created rapidly for nearly any data source on the Internet.

Once the data has been converted to M by a left edge, it can be operated on by any of the pure machines designed to use M as both inputs and outputs. Since the pure machines do not have to correct for different tags used by the creators of the source, they are extremely versatile. After the desired operation has been performed, the user can then use a right edge to convert back into another XML schema, such as RSS.

2.5 Leveraging Social Networks

The entire M Language system is designed to be as modular and general as possible. M forms the basic building blocks for more complex functions that users can "string together" to implement a variety of interesting twists on existing data. Users can enhance the M

system in three ways: contribute a core machine, develop a complex machine from the core machines, or add new words or associations to the database.

Contributing a core machine is an advanced task that users familiar with XML and the M framework can do if they wish to add functionality not currently available with the existing modules. For instance, if a popular website publishes a web service with information that would be useful for another M machine, a user could create a left edge to import this information. As part of our M Language development, we recently added a left edge that queries a UPS web service, fetches the XML output, and converts it to M. By adding this module, the scope of M machines now includes package tracking data supplied by UPS.

Another way for users to add to the Web machines interface is by creating compound machines from core modules. For the example above, a user created a left edge that translated UPS XML into M. Once this module was in place, it was combined with an existing right edge to create a UPS-to-KML machine. The M-to-KML right edge translated M data into Keyhole Markup Language used to annotate Google Maps. The final machine allows a user to input a tracking number and then receive a visualization of the status of the package on a map. Any time a user puts together different modules, the new compound machine is saved in that user's profile, and others are given access to the functionality that it provides.

The final contribution for users is at perhaps the most fundamental level – additions to the M database. Since all the M functionality relies on having an up-to-date database with M-words and their associations, it is important to allow users to add their own words and linkages. For example, a company that develops a new product called 'widgets' would add that product name to the dictionary, as well as relevant associations like type-of, part-of, etc. Now, anytime the product is referred to in a news item or other data entry, it will be recognized by M and is able to take advantage of any of the machines that may operate on it. By allowing and encouraging users to fill in the dictionary and associations as they see fit, the M dictionary will remain updated and relevant. Of course, the dictionary itself will be versioned, so that any additions or modifications by malicious users can be simply "rolled back" to an earlier form.

2.6 User Profiles and Networks

One key component of social networking sites is the ability to link together the preferences of users. The networking begins at the user profile page, where a user can see basic profile statistics as well as their friend network and some information on web machines that are relevant to them. The relevant web machines are categorized as: 1) machines the user created, 2) machines the user designed as 'favorites', and 3) the user's friends' favorite machines.

Together, this forms a sort of dashboard for the user to interact with the community of the site. The friend network is designed to allow users to link with others that share similar interests.

Because the Web machines frontend to M stands to benefit so dramatically from specialty contributions by users experienced in a certain field, we are actively promoting

subcommunities within the site. Much like Wikipedia has different core groups of contributors that are responsible for the content in various areas, we also hope to tap the interest of specialized subgroups with deep knowledge in a given field. Once "critical mass" is attained, by gaining several key advanced users in a variety of different areas, both word-of-mouth and the social networking component of the site will help to enhance the content (in this case, Web machines and modules) in that area. From here, the rest of the community will follow in a top-down manner. The optimal progression of content creation involves three groups of users: advanced users with deep knowledge in a given field, intermediate users with cross- disciplinary interests, and basic users with broad interests. First, the specialized users will create the initial modules and pure machines needed to interface with data sources within a specific field. Next, the intermediate users will begin to link together machines from different fields, creating useful twists on the data coming from different sources. The UPS to KML machine described above is an example of this – once the advanced users built the necessary edges for UPS and KML; an intermediate user could easily link them together. The compound machines will have novel and useful application to the broad users, who will be drawn to the site for the unique functionality created by the intermediate users. Once the basic users are familiar and comfortable with the site, they will start building web machines and machines of their own, thereby transitioning into intermediate users and starting the cycle anew.

3.0 FUTURE WORK

The first iteration of M has been completed – the underlying framework of M-words and semantic relations has been set up, and the public interface is live and accessible to anyone. From this point, experienced users can create modules and compound machines that are accessible from anywhere as a standard web service. However, in order to enhance the user experience and enrich the associative abilities of the M dictionary, a few improvements lie ahead.

3.1 The M Dictionary

Although the current M dictionary has a large amount of links between words, it is always advantageous to have more. As it stands, there is no public interface for adding and editing words and associations. This is an important ability required to ensure that the M dictionary remains up-to-date and flexible enough to describe any type of data.

The process of adding words should be somewhat similar to the process of creating Web machines. When a user adds a word and defines some of the linkages between that word and others, the new word and its associations should show up on the user's profile page. This will help other users determine who shares similar interests when they are looking to build a network. The combination of favorite machines, machines built, and words added will be a powerful heuristic for determining user type. Later revisions of the project can use this data to study the most common usages of the M system in order to better cater to those users or broaden the site to attract new ones.

3.2 Building Web machines

The Web Machine building framework is almost entirely in place. Users can create web machines and edges by a variety of methods including the Translator Factory, raw XML, and the Assembly Line. The tools that exist now eliminate much of the tedious work required to create a web machine by hand, but still required a significant knowledge of the underlying XML architecture and schema. Many of the most valuable users will be unwilling to delve too deeply or at all into the code currently required to create a web machine. Therefore, we are in the process of simplifying both the process to create web machines as well as the steps needed to use the web machine. When this is finished, the system will be accessible to a wider variety of users with a more diverse set of backgrounds, which can then create essential tools for their interests.

4.0 CONCLUSION

The data aggregation and sharing problem is complex. If companies wish to share data, they must use an organized system rather than attempting to develop individual converters to translate data sets on a point-to-point basis. The functionality of M, rooted in its full dictionary and rich linkages between words, allows this conversion to be far less time and resource intensive, while at the same time adding functionality that would otherwise not exist, such as non-ambiguous description tags.

The public front-end to M is a crucial component in marketing this system to the public. By releasing an open framework that enables any user to create their own M structures using any data available to them; we can take advantage of the recent trend toward social networking on the Internet. Since the collective experience and expertise of thousands of users throughout the world is far greater than a single research group could hope to possess, opening the system enables uses for M never conceived by the original creators. In a manner similar to the way Wikipedia has evolved from a small group of initial contributors, we hope that once the proper tools are in place the M frontend will be filled with interesting and useful web machines. The basic framework is in place to make this truly a disruptive technology on the Internet – as the project is continually tweaked and enhanced, we hope that it will begin to grow on its own and develop into an essential Internet tool.

REFERENCE

Schuster, E.W., S. J. Allen, and D.L. Brock. *GLOBAL RFID: The Value of the EPCglobal Network for Supply Chain Management*. Berlin: Springer Verlag (2007).

NOTES

