Transforming Military Command and Control Information Exchange

Key words: military, vocabulary, modernization, migration

Blurb:

The DoD has employed text-based messaging to support command and control for 30 years. A modernization effort to express these exchanges using XML is presented, with an emphasis on lessons learned in providing a migration path from proprietary formats to an XML based approach.

Abstract:

The DoD develops and manages information exchange requirements and their associated business practices to support all military operations. To support the required information exchanges, messaging standards and associated business rules for information exchange among command and control systems have been developed and used for the past thirty years. These messaging standards serve a global user community and are subject to international agreements. Despite their usefulness in promoting interoperability among automated information processing systems of many nations, the proprietary nature of these messages has presented cost, development and maintenance challenges.

An international initiative was launched to investigate the use of XML to improve the quality, capability and affordability of text based, military messaging standards. To satisfy military information exchange requirements, the initiative leverages an extensive investment in the metadata defined in the proprietary standards. The initiative likewise leverages industry XML standards to improve information location, retrieval, exchange and processing across system, organizational and international boundaries. It enables the military to take advantage of low cost, high quality, rapidly evolving commercial software.

Consequently, US services and agencies have recently approved specifications for representing these messages using XML and XML Schema. This provides a common method for a number of nations to represent their military information exchanges in an industry standard format. An overview of the events leading to this accomplishment and current activities supporting the effort is presented with an emphasis on the approach taken and on lessons learned in providing a migration path from proprietary formats to an XML based paradigm.

INTRODUCTION

Purpose

This paper is intended to examine the strategic steps concerning an XML transformation effort. The effort involved the large and various communities of military command and

control information exchange. The highlights of this effort are discussed to convey some general principles regarding the introduction and insertion of XML technologies into an established community of users, designers and developers.

Background

The US DoD as an Enterprise

The term enterprise frequently is associated with business organizations, but more and more the word describes any large organization that uses computer-based technologies. Basic to enterprise development is the ability to locate and gather pieces of information that are essential to making decisions with success in mind. The Department of Defense (DoD) certainly meets the broader definition of "enterprise" and requires goal-oriented decision support at every hierarchical level.

We hear briefing after briefing on the need for information superiority and for migrating military systems to a "web-like" or "network" environment to support increasingly fast-paced and diverse operations. The massive amount of information made available through net-based technologies has challenged us to investigate new ways to integrate digital information into every process supporting the DoD enterprise, particularly those processes that support timely decision-making.

C2 Messaging

To be effective, organizations must share information across dissimilar, independently developed systems that involve diverse languages, cultures and command/management structures. In particular, military processes and the systems that support them are meant to work together as a coordinated whole in conjunction with joint and combined doctrine. This is accomplished through the exchange of services, most typically information-based services. Traditionally external information sources have been tasked to provide (push) information on a procedural basis and in accordance with the doctrine.

A major means of addressing this tasking is through formatted, text-oriented information exchanges. A particular standard of hierarchically structured messages, called Message Text Formats (MTFs), has been used to relay battlefield information since the 1970's. MTFs are similar to commercial messages such as Electronic Data Interchange (EDI). The MTF military standard comprises about 600 message types and over 6000 simple and complex data types, all defined, agreed upon and implemented by more than 70 nations.

Because the purpose of these messages is to promote interoperability, the MTF community is concerned with more than just message syntax. They govern agreed terms, definitions and information exchange requirements needed for conducting military operations. In addition, this community also defines and implements the process to

configuration manage the message definitions and this associated operational meta-data. The agreed structure and semantics of the messages and their contents help ensure information received from source processes is properly interpreted into meaning and produces the required effects in receiving processes.

Overview

In the course of establishing a strategy to employ XML in C2 applications, we developed objectives that we believe can be applied to other communities. We present these as generally as possible, in order to facilitate their application in other communities. We also give treatment to relevant aspects of the military messaging community for purposes of illustration.

The objectives we applied in our community were as follows:

- Gain Community Acceptance
- Foster Community Ownership
- Provide a Voluntary Migration Path
- Achieve Community Agreement

We feel that each of the other objectives is equally important in moving from community accepted proprietary approaches to XML technologies.

GAIN COMMUNITY ACCEPTANCE

Establish Credibility Via Community Participation

Any established community users, designers, and developers has its own "culture." It is important that a migration effort take the social factors of the community into account, as well as the technical issues. A successful migration must involve some members of the given community. Contacts within the community should certainly be leveraged, but it is necessary that ideas for change come from actively participating members.

A number of important aspects require active participation. A complete understanding of the informal or "unwritten" rules (i.e., oral tradition) of how the community works is essential. Along with this comes a historical perspective of long-standing, unresolved issues, and knowledge of any past failure(s) in technology migration. It also includes familiarity with "dead horse" topics to be avoided lest they distract the community from the goal of migration.

The required level of fluency in the community's vocabulary will involve active participation. An understanding of the community's subtle technical aspects will not be evident to an outsider. This is obviously important because such aspects must be treated

carefully. But in addition to this, the members of the community should be convinced that these have not been overlooked.

Most importantly, active participation in the community gives credibility to technical proposals. Many members in the community will not likely have extensive knowledge in XML technology, and will initially proceed via trust in the proposal's advocate(s).

At the time of our proposal to initiate a migration to XML technologies (1999), we had been actively participating in the military messaging community for over 15 years. This participation was the key to gaining the acceptance of the community. In contrast, we have observed the failure of outsider attempts to bring new technology into the military messaging community.

Demonstrate XML Utility

The utility of new (XML) technology must be shown to an adopting community. When we made our proposal in 1999, XML was rather new, and a demonstration of utility was particularly important. Even with XML well established, however, this is still necessary in any community. Indeed, the endeavor to develop a demonstration for a given community may lead to the conclusion that employing XML is not appropriate.

Our initial demonstrations focused on filtering and repurposing XML embodiments of certain important and typically large messages. Because of the terse format used, and their size, these messages are difficult to interpret without some type of parsing/rendering application. These demonstrations showed our community that migrating to XML would make transformation tools (i.e., XSL) freely available. This provided a clear alternative to the expensive and sometimes brittle applications in use for processing the community's proprietary syntax.

"Sell" XML

With the credibility of the advocate(s) and the utility of XML addressed, the job of salesmanship begins. The basis for arguing a migration to XML may be laid, but the argument must still be successfully conveyed. Once again, this is a social, not a technical pursuit.

The members of the community, especially the decision-makers, must be convinced on their own terms. The emphasis of the proposal's presentation should be directly tied to "audience" interests. For example, management might be most interested in how XML can reduce costs, while users would appreciate shorter development times for requested features.

We presented the case for XML to numerous organizations and nations to gain support for our proposal. We began with our immediate customers, and continued with other

members of the military messaging community. There is a wide variance of technical expertise in our community. In most cases, people were unfamiliar with XML, and different approaches to "selling" XML were needed.

In our community there is a typical mix of software developers and decision makers. But there are also some categories of participants somewhat unique to the military messaging community. Members also include experts in military processes, and the case for XML had to be argued from the perspective of increasing mission effectiveness. Finally, there is a portion of the military messaging community concerned with message design and configuration management.

FOSTER COMMUNITY OWNERSHIP

A community must have a sense of ownership in the products associated with an XML migration. The obvious advantage of this is that the community will place more importance on the effort coming to successful completion. But thought must also be given to what happens after the migration has gained momentum. The migration effort cannot solely depend on those who first proposed it. At some point it must be "institutionalized" and self-sustaining, even if its founders move on.

Our approach to build this sense of ownership was to establish a "development team" to pursue the employment of XML technology in the community. The concept of a development team is patterned after W3C working groups. It is a team of qualified staff contributed by organizations with a mutual commitment to developing agreed specifications and software to validate concepts and facilitate deployment of an XML-based alternative to proprietary technology. Participation is voluntary and open to organizations that commit 20% (or more) of qualified staff time.

Only those who have a stake in the community's migration to XML technologies should be offered for team membership. They should also be considered to have expertise in at least two of the following areas:

- Current community information exchange formats
- XML technologies
- Software design and development
- Community business processes and information exchange requirements

The development team should operate independently of the main community forum, but be answerable to the overall community. The independence of the team has two benefits. The first is that normal activities of the community are not interrupted, thus avoiding the perception that XML migration is too great a task. The second is that, as a smaller group, the development team can reach consensus more readily.

As stated above, the development team should develop proposals for the consideration of the overall community. The development team should also bring issues requiring

interpretation and clarification of current practices to the larger community to ensure a common understanding of the team's starting point. In addition, the development team should make every effort to be receptive to community feedback, and to take visible action upon it.

In addition to those with XML expertise, our development team consisted of formal representatives of US services or other nations, as well as contractors and developers of MTF processing software. The formal representatives on our team were influential in the community, and this bolstered the team's credibility. The inclusion of the software developers was also important, as they were able to identify technical problems with implementing the specifications. They were also positioned to add XML features to software currently in use by our community.

Our team produced specifications that were incorporated into our community's standards for message exchange. We released a number of drafts to the community for feedback. When the development team reached consensus on a specification, the specification was formally proposed to our community, using the existing change proposal processes.

PROVIDE A VOLUNTARY MIGRATION PATH

A Single Method for Managing Change

A technical migration must be as simple as possible. Therefore, it is important to establish a single method for deriving and maintaining the XML representations of the community's data definitions. This simplifies the adoption process and minimizes the effort needed to both design and implement a new XML vocabulary.

However, innovation and correction must be allowed to occur in the life of the adopted XML vocabulary. Therefore, it is also important that there exist an organization, department, etc, to act as a responsive focal point for collecting feedback and implementing proposed improvements. The embodiment of this focal point may take on different forms, depending on the nature of the enterprise.

In our case, the process for considering changes to the adopted XML vocabulary was made part of an existing agreement-based process for proposing and deciding upon updates to the community's data definitions. Other communities may have different starting points for XML vocabulary configuration management.

Whatever the starting point, the XML management process must be as close as possible to the existing community's methods for managing change while still being functional. The initial step must be an achievable transition, preferably leveraging the existing configuration management mechanisms and involving a minimal amount of effort. It also leverages the institutional "momentum" of existing processes to increase the probability for viable XML adoption.

Importance of Automation

The importance of automation to facilitate the transition to XML technologies may appear to be an obvious requirement for XML migration. However, we have observed migration efforts which involved only hand-crafting DTDs and/or XML schemas. There were no attempts to leverage existing, electronically stored, data definition information.

If the community's electronic configuration management records can be leveraged, the cost and error associated with managing change in the XML representation can be greatly reduced in the long run. DTDs and/or XML schemas can be automatically generated from the configuration management information which is already maintained by the community. In addition, an automated translation capability to maintain backward compatibility between the existing community syntax and XML can be also be updated with minimal cost and error by leveraging the configuration management records. A discussion of how these points were addressed in our community is presented in the next section.

"Mapping" Specification

In the DoD, as in other enterprises, the rate of new technology adoption varies among organizations and the systems they maintain. While new systems can be developed from the start to utilize XML, most existing systems cannot be migrated from a proprietary data format to XML in a timely manner. Any new information representation will most likely be considered for use in the next system upgrade. In some cases, a system may even be retired before it can take advantage of new technology.

In a realistic enterprise migration to XML, the collection of systems that need to communicate will be heterogeneous with respect to the data format. A significant amount of time will likely pass before all systems can "speak" XML. Some allowance for backward compatibility must be made. During this transition period, a capability to translate between the established format and its corresponding XML representation must be provided to maintain interoperability across the enterprise.

Purpose

The purpose of a "mapping" specification is to define a community agreed method of translating between the established format and its corresponding XML representation. The mapping is intended to be implementation neutral. So, where ever possible, the specification is conveyed declaratively. In this way, the specification can be used as guidance for multiple application development efforts and promote interoperability among them.

Design Principles

Before beginning work on the mapping specification, it is useful to decide upon principals to guide the effort and reduce the number of decisions (and arguments) in the development process. Below are the mapping design principles which were agreed in our community. "MTF" stand for Message Text Format, which is the generic name given to the text-based messages which have been in use. "XML-MTF" refers to the XML representations of these.

1. XML-MTF shall be easy to read, use, and understand.
2. XML-MTF shall be designed to ensure widespread military adoption.
3. XML-MTF documents should be easy to construct from basic rules mapping it to MTF formats.
4. XML-MTF schemas should be easy to construct.
5. Operations on XML-MTF documents, such as a query, should be resilient to schema changes.
6. XML-MTF shall as much as possible draw on industry adopted standards and technologies to save time and money.

The majority of effort was spent to satisfy principle number three. The mapping rules were expressed in a BNF-like syntax, which included production rules and variables to be used in the translation process. These were stated using both XML and community terminology and concepts. The rules were supplemented by more detailed explanations expressed in technical prose.

We discovered that the mapping specification provided additional benefits for our community. Because of the need to refer to both community and XML terminology, developing the mapping specification served a learning experience for many participants. It provided a correlation between our community's established concepts and those in the XML community. Thus, experts in our community's data formats became familiar with XML, and vise-versa.

Schemas and Other XML-Based Constraints

For validation and documentation purposes, XML Schemas, DTDs, etc., should be derived for the set of information exchanges important to a community. If possible, these should be derived automatically from existing community products. This investment yields the benefits of reducing error-prone manual effort, and enforcing a common method of translating the information definitions (e.g., messages). It also supports a period of transition, when both the proprietary and XML formats will be used. The community's existing configuration management processes can also be leveraged to manage the evolution of the XML representation.

The approach to generating the constraints for a community's XML instances will differ, depending on the constraints to be expressed and the level of formality in the existing information definitions (e.g., messages). Regarding our community, there are basic rules governing how to assemble the constituent components of a message, together with specifications regarding the repeatability or optionality of those components within individual message types. In addition, XML types can be derived with some success from existing data element (and aggregated data element) definitions. The generation of XML-based constraint expressions (including XML Schemas) is based on maintained configuration management records, and is automated.

However, these basic definitions are insufficient for capturing all the internal logical requirements pertinent to our community's information exchanges. The specification of each message type used within our community includes additional types of constraints that must also be satisfied. A mathematical language was designed to formally express these additional types of constraints in a machine-processable form.

Tools have been developed and deployed to enforce both types of constraints in our community's message processing systems. A technology such as XML Schema supports only the first type of constraint discussed above. So in order to provide the same level of XML-native validation, a means of re-expressing our community's mathematical constraint language in an appropriate validation framework using XML technologies was devised. This effort is detailed in [XML-Native Constraint Evaluation, M. Cokus, Dr. R. Costello, Dr. M.A. Malloy, E. Masek, D. Winkowski, Proceedings of XML 2004 Conference].

ACHIEVE COMMUNITY AGREEMENT

The community must agree to the XML representation developed by the team. If at all possible, this agreement should be formal. This is often a "painful" process, and may not actually lead to the "optimal" solution, in a technological sense. In the case of a large number of participating organizations, there are bound to be differing conclusions concerning what the best XML representation would be. The most important goal is that these diverse, but "connected" organizations agree on how information will be passed among them.

It is also important to remember that the XML representation is not the only aspect of information exchange, even if the information is represented exclusively in XML. There is more to an agreed information representation than syntax. Other components are essential:

- Information exchange requirements (What information must be exchanged?)
- Terms and Definitions (What is the vocabulary for conducting the information exchange?)
- Processes for information usage (What is the context for the information exchange?)

- Configuration management (How is the evolution of the information managed?)

Therefore, achieving agreement on the community's XML representation is only a portion of the work necessary for effective information exchange. It is necessary to determine what information must be exchanged. In addition, the XML representation will include simple semantics in the form of element names, for instance. But the terms used and their corresponding definitions must be understood and agreed by the community. Also essential is the knowledge of the context for the information exchange. Finally, all of the above, as well the XML vocabulary will most likely evolve. The community must agree how these changes will be managed.

In transitioning to XML, existing community agreement on the aspects discussed above should be leveraged to the maximum extent possible. It is likely that at least some of the aspects described above have already been addressed to some degree in the community. Any products of these agreements should be adopted by the community's XML representation. Community investments should be leveraged to reduce the cost of transition.

In our community, terms and definitions were well established. We leveraged the community's terms to build element, attribute and type names. The corresponding definitions were also retained in our proposed XML representation. This allowed the community to retain its "language", making the transition more feasible and increasing its chance of success.

Our community also has pre-existing configuration management procedures and records. Initially, the configuration management of the XML representation has been integrated into our community's existing change management processes. In the future, changes to our community's XML vocabulary will be managed directly. In addition, our community's agreed business processes, which are agreed but not formally defined, will be the starting point for applying XML-based workflow solutions.

CONCLUSION

At the time of this writing, our community has adopted specifications for representing their information exchanges in XML. The transition to XML has been ongoing. During the transition, our community will support two methods of expressing its information, which are translatable, each to the other. Newer and transitioning systems are employing the XML representation (called XML-MTF). Systems which have not completed or are unable to make the transition continue to use the community's proprietary syntax.

The next step in our community's transition to XML is to manage the XML representations directly, rather than as an additional step in the configuration management of the proprietary information exchange format. After this is achieved, the proprietary syntax will still be supported in the community, but it will instead be derived from the XML-based representation. At some point in the future, the proprietary format

will longer be needed. However, many legacy systems in the military can be long-lived, because of their utility and associated replacement costs. Thus the use of our community's proprietary format, while constantly diminishing, may continue for some time.

Figure-1, below, presents a temporal view of our community's XML transition. It depicts earlier work in supporting the proprietary format with specialized message preparation and parsing tools. In addition, it shows past and current XML transition work

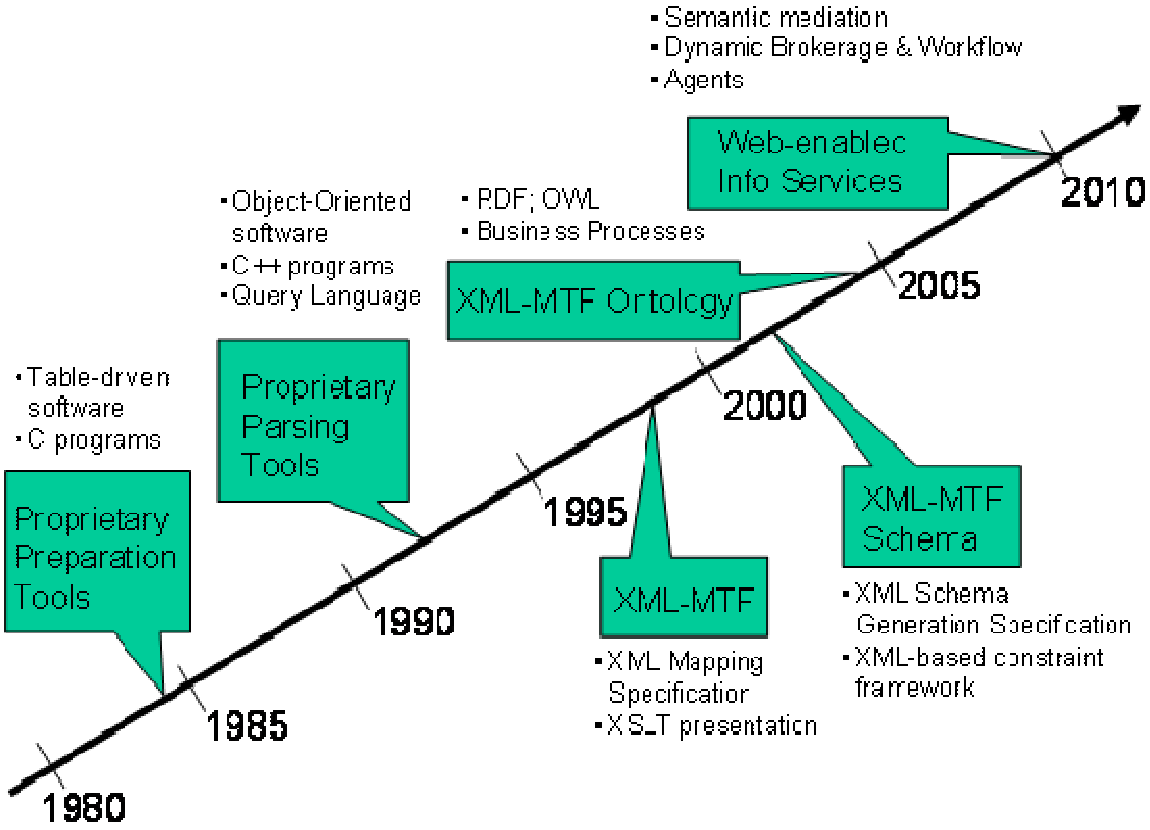

Figure-1 Transition Timeline

Recently, we have begun investigation into developing ontological models which correspond to the information defined for exchange. Some of these semantics can be inferred from our community's agreed information definitions. However, much of the information concerning relationships among the information items is either implicit in documentation, or resides only in the minds of subject matter experts. We are currently investigating methods to uncover and track this semantic information.

In addition to information definitions, our community also maintains documentation concerning which information exchanges are required between given organizations and

systems. Future investigations will include evaluation of means to formally represent the required exchanges as workflows. This promises to eliminate errors introduced by verbal descriptions of the required information exchanges.

Finally, we also endeavor to establish the employment of XML technology to enhance information exchange in other DoD communities. For historical reasons, there exist duplicative representations for information common across communities in the DoD enterprise. We see transitioning DoD information definitions to XML as a way to normalize and compare information exchange requirements. This holds promise for the convergence to common data definitions and a shared change management process.