If Nobody Uses It, It "Ain't" a Standard

Thoughts on Retooling DoD Data Standardization Efforts

Gary A. Ham and Douglas D. Mann
Battelle Memorial Institute

The Department of Defense should abandon the unified data model and single data representation approach to data standardization in favor of a shareable and open repository approach in which standards are chosen on the basis of quality and competitive merit.

Department of Defense (DoD) data standardization policy[1] (particularly as implemented in the DoD Data Model) has inherited much of its structure from the Corporate Information Management (CIM) concept. The data side of CIM, as implemented in DoD Directive 8320.1 [1], is based on the assumption that a single data structure, designed from the top down by selected subject matter experts, can be crafted to meet the needs of all development efforts.[2] True believers in CIM consider the admitted high cost of maintaining a single complicated relationship structure and a single approved representation for each information concept to be warranted in light of the benefits to be received. Benefits cited include well-defined, usable data structures, effective reuse of data in multiple systems, and higher-quality systems with lower maintenance costs.

Standardization Problems

We are not true believers. There are at least three significant problems.

Independent Definition

First, usable data structures cannot be defined independently from system requirements. While the same data structures can and should be reused in multiple systems, their structure must first and most important be based on mission activities. Data does not exist independently of mission, and mission implies a functional requirement. The problem: Data standards "defined" without a direct tie to a specific mission requirement have no basis for standardization. They may exist, but they have no purpose.

Differing Requirements

Second, different systems have different missions and, therefore, different requirements. Although there may be common data structures in different systems, the relationships these data structures have with other data structures may be different. Trying to maintain them all in a single, over-arching model is complicated. It is often argued that data is easier to model than process because data structures are more stable than processes. Relationships, however, often represent the processes that connect data structures. Imposing the relationships defined in a single, highly detailed model inhibits appropriate reuse of the data structures in the model. The problem: Standard data models that impose fixed process-oriented relationships restrict process change just as rigidly as any hierarchically defined process model.

Standardization May Not Reduce Costs

Third, standardization does not necessarily improve software maintainability or save on maintenance cost. Standardizing internal data structures removes the benefit of module encapsulation because it creates unwarranted coupling[3] between systems. When a data structure must be changed for one system, it has a ripple effect on all other systems using the data structure. The net effect is the creation of brittle systems that cannot be changed effectively for fear of side effects. The larger and more comprehensive the "standard" data structure, the more pervasive this "quality killer" becomes. The alternative is to develop work-arounds to avoid changing the standard data structures. Such work-arounds impose increasing degrees of maintenance brittleness onto a system, which increases future costs and decreases the flexibility to introduce additional change. Perhaps the best example of data coupling in the real world is the Year 2000 problem. Fixing this one badly chosen standard[4] will be expensive. Imagine if the structure were a bit larger. The problem: Building multiple systems around a single standard data structure is likely to *add* cost and increase maintenance effort.

The Most Beneficial Standards

On the other hand, communication of any kind is impossible without standards. Neither humans nor systems can understand each other without understanding both representation (the commonly agreed-on "sign," such as a word, character, or gesture) and concept (the object or idea to which the sign points). The more widely used a language is, the more useful it is for general communication, regardless of the quality of language construction.[5] The writing system for the English

The opinions in this article do not necessarily represent those of Battelle or its DoD clientele. Data standardization in DoD has been difficult—the solution is open to debate. Critical response is welcome.

language, for instance, is a hodgepodge of conventions from several languages. Consistency is not its strong suit. Nevertheless, in a world where English is increasingly becoming the common language of the business world, poorly spelled English functions better than no common language at all. Data standards work the same way. The most beneficial standards may or may not be the best in terms of any arbitrary standard of quality; rather, they are the ones *perceived by the user* as the most beneficial, because of adoption and common usage.

Sometimes, standards can be imposed by a common authority. The Health Care Financing Administration (HCFA), for instance, will probably have some success with the individual standards it chooses to impose because it has the power of enforcement under the Health Insurance Portability and Accountability Act of 1996 [2]. Even HCFA will not succeed, however, if it chooses to impose standards that are not *perceived by the user or the developer* as usable. Simply put, no developer will attempt to achieve something that is not perceived to be possible. The complicated nature of the current DoD Data Model is not generally perceived by developers to be implementable. In fact, the Government Accounting Office (GAO) found that only nine of 43 major DoD systems had plans to use standard data [3]. Smaller systems, with lesser resource allocation, are probably even less compliant. It is just too hard.

DoD is not alone in the practice of building data models that are little more than shelfware. Developed primarily by IBM as a standards proposal, the *Information Resource Dictionary System-Information Model* (draft dated April 8, 1992) consists of 763 pages [4]. The model was developed as the IBM Information Model in MVS-based Repository Manager. It was probably an extremely expensive development project. Unfortunately, the model is so complicated that it was not adopted. Whether IBM has made other use of this document is unknown. There are undoubtedly many other examples (usually unpublished).

DoD Standards

If DoD (or any organization) wants a successful data standardization program, standardization authorities must recognize that they have two objectives: develop or adopt *usable* standards and convince users and developers that the standards are usable. If the first objective is not realized, the second will not be, either. Without the second objective, the first is useless.

Usable Standards

Usable standards development requires the participation of developers. It must be system requirements based. Data structures must track directly to defined system information requirements. Simply, if you cannot state a specific use for a piece of information, you cannot consider it usable for standardization. Development of usable standards means cooperation and teamwork with actual development systems. If no developers are actually using a standard you develop, it ain't[6] a standard.

Adoption of usable data standards implies that the standards are *already in use* somewhere. They may be industry, government, or standards-organization sponsored. Usability is a function of quality, but the real measure of usability is widespread acceptance and implementation. Standardization may require compromise where the most widespread standard is "not as good" as its less widely used competitor or the one developed in-house. The point is, there may be no one standard for any particular concept or representation of that concept. Instead, there may be several. The best standardization programs choose the "best" standards by reviewing them all against mission activity and system development requirements. It is conceivable that more than one representation of the same concept could be adopted to meet differing mission requirements.

Not a Top-Down Process

While requirements definition should be done from the top down to ensure completeness, effective use of data standards is not a top-down process.[7] Choosing or building standard components to meet functional requirements should be done at the level at which the requirement is to be implemented. Standards should apply only to that information that is brought in or sent out from the requirement. Data internal to a particular requirement solution should remain decoupled from its interface with other requirements. Externally visible data should be standard within its sphere of visibility, which means that a particular concept must use the same name and structure within its context of visibility. Each layer of encapsulated visibility must meet its own set of standards for that layer. If passed beyond that layer, data must be wrapped to the set of standards applicable the next layer of visibility. By "wrappered" encapsulation, internal changes to system structures are not held hostage to changes in outside standards. Similarly, changes needed internally within a system are less likely to cause external system side effects. Only the interfaces need to be maintained.

It is at the interface level that standardization is particularly important. Systems that must interface with another system's nonstandards-based interface or with several different sets of standards must maintain multiple interfaces—one for each standard and one for each nonstandard system. Choosing a particular set of standards at each context level, i.e., level of visibility, reduces this interface to one. On the other hand, if reducing the set of standards to one creates a highly complicated set of intricate relationships, the one level may be harder to maintain than a multiple set of interfaces. The trade-off must be managed.

Implementation Management

The management of standards implementation is a necessary but inherently difficult process. To be successful, standards must be used to interface between systems and system components at the same level of visibility, without inhibiting encapsulation at layers above and below that level. A particular data standard can be adopted for use at all levels, if warranted, but only at the interface definition should such an adoption be

enforced. In fact, effective encapsulation requires some separation between interfaces and internals so that changes to one do not require extensive changes to the other. Making everything the same may make it easier to write the initial code. Maintenance costs, however, can be expected to increase.

The best overall standardization guideline is to adopt the most widely used standard for interfaces in general. Implementation, however, should only be enforced at given levels of visibility. Standardization becomes the process of choosing the standard representation for data to be absorbed or provided at a given level of visibility. Standardization within a particular system should be left to that system. Data passed from system to system for systems managed or owned by a particular functional area should be standard in name and representation throughout the functional area. Data transfers between DoD systems should meet DoD standards. Data passed to or from commercial sources should meet the appropriate commercial standard even if an additional interface is required.

Standards Composition

Just as the level at which a standard is appropriate varies in scale, so does the composition of the standard. An adoptable standard may be as simple as an individual code list or the structure of a single data element. It may also be as complex as an entire system interface (or a defined interface to a commercial-off-the-shelf package). In the object-oriented view, adoptable standards will consist of interface definitions for reusable components, varying in size from a single object class to an entire system.

Standards Adoption

Standards adoption is a process, not a localized, one-time event. It means comparing requirements with existing standards, picking an appropriate one where available, adapting one where it "almost" meets needs, or developing a new one where requirements are not compatible with what is available. Success in such a process has nothing to do with "correct" model building. Success comes from adopting standards that can and will be used. The key to that success is access to competing standards and visibility of how they are used. In the marketplace of ideas, the most usable standards will be adopted. Poor definition and incoherent design will be abandoned. In some cases, the best design may not win due to early adoption and wide dissemination of an otherwise competent predecessor. The value of reuse may outweigh the quality of later improvements. This is a decision that must be based on functional requirements and available resources.

Standards Registry

In the marketplace world, there is no "standard" set of standards. There are, however, multiple standard-setting organizations that offer their goods to the world. A standards registry can be used as a tool to provide an effective marketplace for these standards. Standard-setting organizations act as registration authorities, entering their adopted standards to their own space on the registry. Other organizations can then adopt standards from the registry for their own use or put up competing standards of their own in their own space. An international standard, ISO/IEC 11179, Information Technology – Specification and Standardization of Data Elements [5], provides the foundation for defining a registry for data elements and concepts. The six-part standard addresses

- Framework for the specification and standardization of data elements.
- Classification for data elements.
- Basic attributes of data elements.
- Rules and guidelines for the formulation of data definitions.
- Naming and identification principles for data elements.
- Registration of data elements.

The draft American National Standard, dpANS X3.285, Metamodel for the Management of Sharable Data [6], takes the international standard and extends it into data-value domains and concepts. A draft technical report, Concept of Operations for a Data Element Registry of July 1996 [7], from the American National Standards Institute (ANSI) National Committee for Information Technology Standards L8 – Data Representation subcommittee addresses the operation of a registry based upon the International Organization for Standardization (ISO) and ANSI standards. A registry based on these standards has the capability of registering concepts, data elements, data-value domains, classification schemes, structures, and name contexts.

An organizational registry should support access to multiple registries for various registration authorities. When components may be viewed, compared, evaluated, and selected from multiple sources, the marketplace factors of quality and cost become important selection factors. Multiple registries also facilitate harmonization through cooperative consensus and peer pressure. Information sources and subject experts are identified for consultation. Registries also identify work in progress, approved future components, and older versions of components. An interesting side to the registry is that it improves data-standard quality by exposure to the public. What is not understood or is incomplete can be questioned and clarified.

Several organizations are developing registries based upon ISO and ANSI standards. The U.S. Environmental Protection Agency has the Environmental Data Registry. The Australian Institute of Health and Welfare has developed a health-care-related registry. The U.S. Census Bureau is about to release its registry. All these meet the international standard for registries. The Defense Data Dictionary System (in contrast to the DoD Data Model on which it is supposed to be based) also serves as a registry, although it does not provide the full functionality of the international registry standard.

Conceptually, the DoD Data Dictionary should be retooled to conform to international repository standards. Its management should act as the registration authority for information standards that apply at the DoD level. It should adopt standards from other registries where appropriate and make all such registries visible. It should pro-

vide a central registry and appropriate function area-based subregistries, overseeing a consensus-building effort toward mutually compatible systems interfaces based on well-defined, usable standards. Selected standards (particularly, systems interfaces and code domains used in multiple systems) could be dictated for use in all systems. DoD's current "new idea" in data standardization, the Defense Information Systems Agency-sponsored Shared Data Environment (SHADE) segment registration process [8], is an excellent beginning for this kind of well-grounded standards development process, although visibility to alternate registries would enhance quality and usability. In the meantime, attempts to require "standardization" for entire database structures internal to developing systems should be abandoned. Requiring such structures is actually quality inhibiting and not enforceable in any real sense.

Modeling

Modeling plays a different but extremely important role in this type of standards registration process. Instead of making sure that a potential standard meets the structure of some formal data model, standards that are proven to meet defined functional requirements are modeled to show their relationship with other adopted standards to improve their accessibility and provide opportunities for reuse. The change in focus is important. *Do not standardize the models. Instead, model the standards.* In this environment, functional area models are important navigational tools for using and integrating standards during systems development. Data relationships are modeled with data models. Component relationships are modeled as object models. Finally, all components must be mapped to a mission model.

Mission-based requirements models should be the *only* top-down-defined models in the DoD information management program. Even these models should be based on the required sets of measurable results needed to accomplish a mission rather than process steps involved in getting there. System functional requirements should be validated as supporting mission requirement components. Approved standards should support defined mission requirements through system functional requirements. Traceability is important. Ad hoc requirement definition is not inherently bad, but ad hoc requirements that cannot be validated in terms of specific mission activity support should be considered invalid for further exploration. Similarly, if a registered standard cannot be shown to support at least one defined mission requirement, it should be deregistered as an approved standard. Data models and object models remain players in this arena but should become models of approved standards tied to defined requirements. They should be composed from the bottom up using validated standards.

Conclusion

A change toward competitive registration of standards and bottom-up standards model development and away from dictated single data structure models would result in a data standardization program that makes sense. Standards would be defined in usable form. Standards could be traced to mission-based requirements. Most important, standards would be used to enhance communication between systems without the side effects of retarded development and increased cost. ◆

About the Authors

Gary A. Ham is a principal research scientist for Battelle Memorial Institute, National Security Division, Information Systems Engineering and Process Modernization Department in Arlington, Va. A former Marine Corps comptroller and Naval Academy computer science instructor, he is currently researching value metrics definition processes to support object-oriented requirements analysis and design of DoD systems. He has a bachelor's degree in economics from Whitman College in Walla Walla, Wash. and a master's degree (with distinction) in information systems management from the Naval Postgraduate School in Monterey, Ala. He is currently a doctoral student in information technology at George Mason University in Fairfax, Va.

Battelle Memorial Institute
2101 Wilson Blvd., Suite 800
Arlington, VA 22201-3008
Voice: 703-575-1072
Fax: 703-820-8817
E-mail: gham@erols.com

Douglas D. Mann is a senior research scientist at Battelle Memorial Institute in Arlington, Va. He has 27 years experience with Control Data Corporation as a consultant in data management. He has also worked at InfoSpan Corporation implementing repository systems using the Information Resource Dictionary System standard. He is currently on the L8 – Data Representation Standards subcommittee as editor of the draft ANSI standard X3.285 – Metamodel for the Management of Sharable Data. He also has prior experience as a member of the H4 – Open Repository Standards subcommittee. He is currently involved in the implementation of a shared data registry for the U.S. Environmental Protection Agency.

Battelle Memorial Institute
2101 Wilson Blvd., Suite 800
Arlington, VA 22201-3008
Voice: 703-575-0333
Fax: 703-575-0383
E-mail: mannd@battelle.org

References

1. DoD Directive 8320.1, "DoD Data Administration," September 1991.
2. Public Law: 104-191, "Health Insurance Portability and Accountability Act of 1996," ftp://ftp.loc.gov/pub/thomas/c104/h3103.enr.txt.
3. "Defense IRM Poor Implementation of Management Controls Has Put Migration Strategy at Risk," U.S. General Accounting Office, Report to the Ranking Minority Member, Committee on Governmental Affairs, U.S. Senate, GAO/AIMD-98-5, October 1997.
4. Information Resource Dictionary System – Information Model (working paper of X3H4 Information Resource

Dictionary System Committee, X3H4 92/0xx), April 8, 1992.
5. ISO/IEC 11179, "Information Technology – Specification and Standardization of Data Elements," ftp://sdct-sunsrv1.ncsl.nist.gov/x3l8/11179.
6. dpANS X3.285, "Metamodel for the Management of Sharable Data," ftp://sdct-sunsrv1.ncsl.nist.gov/x3l8/x3l8docs/x3.285.
7. Draft Technical Report, "Concept of Operations for a Data Element Registry," ftp://sdct-sunsrv1.ncsl.nist.gov/x3l8/x3l8docs/drconops.rtf.
8. Defense Information Infrastructure SHADE Capstone Document, Version 1.0, July 11,1996.

Notes

1. Http://www-datadmn.itsi.disa.mil provides information on current DoD data standardization policy, including access to the most current update of the DoD Data Model.
2. Official policy requires data models that are based on the structure of the DoD model, but the policy is not specific as to the detail required. Standards constructed using standard naming conventions and representations can theoretically be approved without imposing the rigidity of a standard model. Unfortunately, all functional areas with which we are familiar (four out of more than a dozen) have interpreted both written and verbal guidance from DoD to require detailed standard models. Furthermore, we have witnessed potential standards submitted without detailed compatibility turned down as standards in two functional areas.
3. The term *coupling* refers to the situation in which one module in a system shares internal information with another module to the extent that modification to either automatically requires modification to both. In programming, global variables used in multiple procedures "couple" the procedures together for maintenance purposes. We can say that data coupling occurs when disparate modules directly access a database structure. In such cases, changes to the database required in support of one module affect all other modules that access the same data. With modular encapsulation, change can be limited to the interface level, which reduces the degree of maintenance required.
4. To be fair, the two-digit year standard was not so "badly chosen" at its origin. With memory space at a premium, it was a good idea at the time. But "time" is the operative word here. Over time, good standards can become bad standards. Forcing data standardization into the bowels of otherwise disparate systems makes the inevitable correction process much more difficult.
5. Specialized languages, human and computer, may be more useful for specialized purposes (encapsulated purposes). They will still require translation into a more generalized "standard" if communication with outside people (or systems) is required.
6. "Ain't" is a well-understood, generalized representation for a concept whose more preferred representations are "am not," "are not," and "is not." As a generalization, ain't is a more "standard" term than any of its substitutes.
7. The development of human language constructs is not top down, either. The only known human language constructed from the top down is Esperanto. Although there is an Esperanto language authority, there are no native Esperanto speakers, and adoption of Esperanto has gone essentially nowhere. To adopt standards that are not already in general use in some form is likely to achieve the same lack of success.

Call for Articles

If your experience or research has produced information that could be useful to others, *Crosstalk* will get the word out. Not only is *Crosstalk* a forum for high-profile leaders, it is an effective medium for useful information from all levels within the Department of Defense (DoD), industry, and academia.

Published monthly, *Crosstalk* is an official DoD periodical distributed to over 19,000 readers, plus uncounted others who are exposed to the journal in offices, libraries, the Internet, and other venues. *Crosstalk* articles are also regularly reprinted in other publications.

We welcome articles on all software-related topics, but are especially interested in several high-interest areas. Drawing from reader survey data, we will highlight your most requested article topics as themes for 1998 and 1999 *Crosstalk* issues. In future issues, we will place a special, yet nonexclusive, focus on

Software Quality – *November 1998*
Article Submission Deadline: July 6, 1998

Requirements Management – *December 1998*
Article Submission Deadline: Aug. 3, 1998

Year 2000 – *January 1999*
Article Submission Deadline: Sept. 1, 1998

Process Definition and Technologies – *February 1999*
Article Submission Deadline: Oct. 5, 1998

Configuration Management – *March 1999*
Article Submission Deadline: Nov. 2, 1998

Test, Inspection, Peer Reviews – *April 1999*
Article Submission Deadline: Dec. 1, 1998

Look for additional announcements that reveal more of our future issues' themes. We will accept article submissions on all software-related topics at any time; our issues will not focus exclusively on the featured theme.

Please follow the *Guidelines for Crosstalk Authors,* available on the World Wide Web at http://www.stsc.hill.af.mil. Hard copies of the guidelines are also available upon request. All articles must be approved by the *Crosstalk* Editorial Board prior to publication. We do not pay for articles. Send articles to

Ogden ALC/TISE
ATTN: Heather Winward, *Crosstalk* Features Coordinator
7278 Fourth Street
Hill AFB, UT 84056-5205

Or E-mail articles to features@stsc1.hill.af.mil. For additional information, call 801-777-9239.

Forrest Brown
Managing Editor