The Role of Government in Standards

Gio Wiederhold
ARPA and Stanford University

1. Introduction

The role of government, at a high conceptual level, is to assure the country's stability while promoting progress, and standards certainly encourage stability. Standards efforts are also intended to promote progress. However, when we consider the actual imposition of standards we find these goals conflicting. Stability is achieved by adhering to accepted, i.e., old, standards. Progress may require moving to new standards, and requires new investments. Stability avoids costs due to change. In this note we focus on this conflict when discussing the role of government with respect to standards, mainly from the point of view of a research agency, such as ARPA (The Advanced Research Agency of the U.S. Department of Defense, known from 1972 to March 14, 1993 as DARPA). We have to deal with the tension of promoting progress, through research, versus stability.

To focus the discussion we present first what items are appropriate for standardization, and how acquisition policies can indirectly hinder progress. We continue by considering what kinds of direct government involvement are effective in the sequence of activities which make up the process of standardization. We report on some topics where ARPA is an active participant in the process. We elaborate on the problems encountered in these activities and the lesson learned. We close by assessing the apparent explosion of standards activities seen today.

2. Progress versus Stability

Having standards in their area of investigation is frightening to most researchers. An innovation will not be accepted if the users cannot accept the risk of moving beyond the stable boundary provided by existing standards. Unless an order-of-magnitude improvement can be shown, the payoff of an innovation is not likely to warrant a user's re-investment. And, unfortunately, most innovations only provide fractional improvements, especially in larger systems, that are composed of many parts. In such systems, a component's task, even if brought down to zero cost, will affect the total systems cost by only the fraction of its role in the overall system operation [Winner:93].

At the same time, these same researchers concerned about barriers to acceptance of their innovation will prefer that all components outside of their area of concern remain stable. A stable environment simplifies their experiments and permits evaluations to be comparable from beginning to end.

Cost and benefits of standards

Investments in systems based on no or on obsolete standards are typically so great that old systems survive much longer than is technologically warranted, leading to so-called legacy systems. A legacy system is one where the investment needed to move to a technology known to be better is unaffordable, either because of its absolute magnitude or the time needed to achieve adequate return on the investment.

But standards also promote progress. Standard interfaces among components permits manageable groups to work independently, focus on developing better components, and be assured that their results will interoperate, even in complex systems. Having standards promotes competition, because now components can be replaced. Improvements can occur as needed, and each increment has a much lower cost than replacing the entire system. Modularity, achieved through standard interfaces, can hence lower the barrier to replace legacy systems, at least those that adhere to interface standards.

The systems we are building now, the legacy for future generations, must adhere to standards, and those standards should be wisely chosen.

3. Roles of Standards

From the preceding discussion we observe that standards should focus on interfaces, not on components. This, seemingly obvious rule, is sometimes forgotten in system acquisition. The federal government is a major culprit here, but corporate decisions are often equally short-sighted. Poor insight into the mechanisms of stability and progress leads to misinterpretation of the role of standards. We will elaborate on that point.

3.1 Interfaces, not components:

Since a purchaser tends to think in terms of acquiring *things*, it is natural to standardize on things: "let's determine what the best *thing* is, and then buy lots of them." Deciding on a particular brand of personal computer is a case of specifying a *thing*. Such an attitude is acceptable when *things* change slowly. However, failure to specify the interface prevents incorporation of better components. A personal computer, of course, has many interfaces: communication, languages, some critical software packages, and the aggregate specification is difficult. In earlier times the interfaces of the *things* were often self evident. For instance, early U.S. purchases of British locomotives led to an interface standard of railroad trackwidth derived from British mining cars. Today, trackwidth is the crucial standard and standard rail dimensions are extremely well specified.

Software can be especially difficult to specify in terms of its interfaces. The focus must be on its ability to exchange data according to some standard representation. But the quantity and dimension of the data, the limits of its computational capabilities, the scope of the meaning of variables, and the means of reporting errors, are often as crucial. Specifying one vendor's software package by name is the primary culprit leading to unimprovable legacy systems. There is no economic benefit for the vendor in improving system performance by interoperating with newer software components or moving to better hardware.

The winning hardware and software vendors love an acquisition approach based on components, rather than interfaces. System vendors do not have to make their internal interface specifications or their stored data representation public. Lack of interface specifications effectively shuts out competition and potential progress. We have here a case where government policies, meant to protect the public's purse, can lead to huge long-range costs by promoting the establishment of undecomposable legacy systems.

3.2 Complex interfaces

In communication the standards have always focused on the interface itself: ANSI RS232, IEEE 802, etc. In stored data the standards will focus on representation as an indirect interface: ANSI format for 9 track tapes, storage conventions for sharing spreadsheets, and product interchange standards in mechanical (PDES) and electronic (EDIF) engineering.

Both types of standards support interoperation of components. However, when a representation is chosen as the standard, then there is also, although maybe only implicitly, an associated interpretation. The meaning of bits, or characters, or words in a representation requires a shared understanding. Representation-based standards can accommodate a wide range of information because of the combinatorics of their content, whereas most basic communication standards are relatively contents-free, or limit themselves to counts and finite code sets.

To deal with the problem of interpretation of complex representations, we can move to an object-oriented paradigm, where some of the interpretation is encapsulated in the object. Now the behavior of the interpretation becomes a standardizable item. Standardization of large objects becomes similar to, and as complex, as the standardization of software packages [OMG:92]. The need for composition of these objects is, however, obvious to all concerned.

4. Promoting standards

Standards require exposure and promotion if they are to be adopted by a sufficiently broad community to become effective. An unused standard isn't a standard.

Standards can be formalizations of specific existing industrial conventions (IBM style 9 track tape, Adobe Postscript, ...). Such a standard derives its exposure from practice. Getting exposure from practice works well if there is a dominant or leading supplier. Some suppliers in fact promote their interface as a standard in order to improve its market or marketability. The use of a proprietary standard may require licensing fees and a commitment to adherence.

In other cases, proposed standards do not have initial practical exposure, typically because the suppliers are weak and diverse. In that case standards may be defined by a consumer group which determines that it has a problem in purchasing supplies that can be solved by consumer cooperation. The type of consumer problem that motivates standards efforts is inadequate and unreliable access to an emerging technology when there is not a dominant supplier. Many manufacturing and materials standards are motivated by a need to have a broad and reliable supply base. For instance, common specifications for steel and concrete were originally initiated by consumers. If the consumer group possesses enough market share such a standard can be effective.

Sometimes a single consumer tries to establish standards for its suppliers. Recently some automobile manufacturers have instituted such standards, and require adherence by their suppliers. Cooperating suppliers may be guaranteed some level of orders, but the mutual dependence has risks, and hinders the customer from future innovation as well.

Consumer driven-standards rarely push technology, since risks are to be minimized. However, they may be rapidly established, because the business interests of the participating consumers are likely to be coherent.

Standards can also be instituted by a consortium of suppliers who realize that the users will not commit to their technology unless there is commonality. If the number of potential users is large and diverse this approach is more feasible than the consumer-driven standard. The consortium provides anti-trust protection, as enabled in the U.S. by 1984 legislation, and becomes a surrogate for the dominant leader.

Supplier-based standards negotiations are often protracted, because each supplier wishes to gain acceptance of its unique contribution and thus gain market advantage. The cost of participating in standards is relatively high for small businesses, further lengthening the development period. The best people may find the process tedious and unprofitable. The promulgated standard often becomes the lowest common denominator of what the consortium partners can commit to. It is likely to be open-ended to be compatible with specific augmentations.

4.1 Assuring fair and broad representation

Helping coalitions of consumers or suppliers in formulating standards is a prime candidate for government involvement. In a field where progress is rapid, businesses tend to be small and lack the resources nor long-term view to participate in standard activities. An example is seen today in the world of object-oriented databases, where many small companies compete, but where the lack of a standard makes it difficult for industrial users to make substantial commitments. ARPA is actively engaged in making standards come about in this arena, so that its promising technology can be accepted [WeBlTh:92].

Government support of standards consortia can help in achieving a balanced representation in such coalitions by providing support for meetings and publication. Some groups are supporting themselves today by charging stiffly for its publications. While the material may be worth its price to directly involved parties, such a tactic hinders broad review and critique. Government infrastructure support can also help in bridging the gap between consumers and suppliers, when one of the groups is weaker. It can also support unbiased evaluation expertise.

Where the concepts to be standardized are complex, it is beneficial to submit the proposals to formal, most likely academic, scrutiny. Although the academic community should have the competence to perform deep analyses of the correctness and scope of standards, such work is relatively rare. The academic focus on original publication, invention, and novelty inhibits such efforts, unless mitigated by funding.

At the same time, presenting new technology to academics for scrutiny provides useful experience about real-world issues to academics. Having joint review by industrial users, suppliers, and academics will reveal assumptions on preconditions, resources, costs, and other elements that hinder successful and rapid adoption of isolated inventions. Such reviews provide complementary feedback to the typical review of a journal paper, and can often lead to new topics that need research.

We see the benefit of joint reviews in Japan, where the Ministry of Industry (MITI) supports mainly the review efforts of its large projects, rather than the research laboratories where the research is being performed. Furthermore, MITI is inhibited from direct support of academic research, so that academics welcome the travel support and the challenging interactions that occur during these review processes [JTEC:92].

4.2 Demonstrating viability

It is unlikely that a standard that has not demonstrated practical effectiveness will be adopted. Here is another potential role for government: helping build prototypes that are sufficiently comprehensive to demonstrate both the benefit and the implementability of services satisfying the proposed interfaces.

Early use of products conforming to new, candidate standards may also be of interest in academia. If such products can be provided in a form that measurement and adaptation is convenient, mutual intellectual benefits can ensue. That flexibility to interact deeply with a new product is less likely with early versions of purely commercial products, since there the code tends to be wholly proprietary. Government support can make pre-competitive prototypes freely available, but should stop when productization becomes a real possibility. The boundary is hard to establish. An academic prototype typically requires seven times the original investment to become of industrial value, so that the concern about favoritism is minor. But if the prototype is produced by a commercial entity, then there will be a conflict between maximizing governmental support of an emerging product and restricting its free distribution.

Software prototypes present some unique opportunities and problems. Distribution of binary or object code permits broad use without direct revelation of its proprietary innards, but makes adaptation difficult and lessens academic interest. Inadequate software construction processes tend to delay adequate documentation, making any type of evaluation difficult. Software is cheaper to copy and easier to distribute illicitly than any other valuable product.

4.3 Market support

The support of a standard for evaluation, and hence early distribution creates an initial market. Non-academic organizations participating in standards assessment will only do so if they have a need, and hence are prime candidates for early products as well.

The same concerns apply to governmental participation, but more care is needed here. Early governmental adoption of an emerging standard can give direction, but can also be misleading. Early standards are often poorly validated, and will eventually be rejected in the marketplace. In the meantime, much legacy may have been created and must be replaced. Broad and unbiased consensus should be obtained before embracing a new standard.

Whenever the government is both a supporter of the standards process and a candidate consumer its roles can easily be confused and confusing.

Government support of standards should not be overly directive. It can be simple for departmental bureaucrats to define standards for an acquisition, since few people will argue against the benefits of standards. Suppliers that stand to benefit will be enthusiastic, and others are easily ignored. For instance, many MILSPEC standards have undergone little public scrutiny. Here the double role of government is worrisome.

The desire of purchasers to avoid apparent risk by subscribing to any standard can also be a hindrance, because an early, unvalidated standard will be more attractive than one with better prospects in the future. Currently, DoD adoption of PCTE for software tools is one such issue. Rapid changes in software technology make it difficult to determine standards. The government as customer can be important, but its tradeoffs rarely match those of industry. In the near future, its influence as a purchaser is diminishing.

4.4 Tests and measures

To demonstrate the viability of a standard it must be tested, and the results must be disseminated. However, the measures are applied to actual implementations, i.e., *things* and conclusions will be drawn about the thing as well as the standard. It is very difficult to make comparison reports fair. Products will be at different stages of maturity, and results will differ for peripheral reasons. The issue is so worrisome that industry is rarely willing to publish its comparative findings, lest it be challenged.

Government involvement with this phase of the standards process is awkward. It can lend an unwarranted stamp of validity on data that can never be fully truthful. Even if the numbers are produced by an academic group, which is expected to be impartial, it is rare that no linkages can be made to commercial interests [BiDeWTu:83].

5. The Multiplicity of Standards

A well-worn joke is that standards are wonderful, that's why we have so many of them. But the multiplicity of standards is unavoidable. As standards move beyond syntactic interfaces, semantic issues arise that induce domain differentiation. Syntactical standards ignore the meaning of the data being communicated or stored, for instance the RS232 standard for terminal connections is at a purely syntactic level, which is common to all application domains. But soon we go beyond. Even ASCII has different extensions for different national languages. Different applications impose yet more radical distinctions.

5.1 Domain standards

Standards for design of electronic parts may have little overlap with those for mechanical equipment. Also, the supplier and user communities for such domains are initially quite distinct. They will work effectively and independently, focusing on their domains of expertise. There are now ongoing standardization efforts in every possible domain of engineering, of healthcare, and, increasingly, in software. As these initially disjoint standards develop they tend to broaden. Successful standards groups tend also to take on related topics.

Eventually, the broadening standards start intersecting each other, and conflicts arise. For instance, file format standards for the exchange of mechanical parts information expand to consider their object storage in databases. Here they intersect with database language standards that are being broadened to encompass object representations. The candidate users are confused. The software specialist in a group is likely to prefer the computer-technology-based standard, the engineer has more faith in standards that are grounded in engineering disciplines.

5.2 Interoperation of standards

Major projects require access to multiple standards. The standardized electrical parts drive mechanisms defined in the mechanical other domain. In healthcare there are standards for the vocabulary for billable procedures, for the formatting of medical orders, for the transmission of drug orders, for treatment rules, and for the organization of reports.

Initially, these standards benefitted from being developed in relative isolation. Efforts to establish standards require a focus on detail and specialized contributions. When problems arise at the intersection the contributors find themselves frustrated and misunderstood.

There appears to be a role for government here, but acceptable mechanisms are unclear. Partitioning of the world into non-intersecting spheres seems impossible. It seems best to follow the partitioning used among and within professional societies. If such authority is given to the societies, it must also impose a responsibility to focus on interoperation among standards.

A technical issue is the choice of the meta-language used to describe standards. If all standards were expressed in the same language interaction among standards would be easier, even though terms will have to be defined in each domain. If such a meta-language were formal, it could become feasible to check automatically for intersections and conflicts.

5.3 Computer language interoperation

An example of an early government sponsored standard was the Ada programming language. But early government adoption and controls kept it from maturing. Its complexity kept it from being taught widely. It was eventually overtaken in practice by languages as C, that are hard to defend as being better on an objectively scientific basis. But, similarly, no one will defend English objectively, and yet, like C, it seems to outpace all alternatives.

In retrospect, it seems unlikely that there will be a common programming language. An ARPA effort to establish a Common Prototyping Language (CPL) [Carlson:92] has led to half a dozen languages. Can multiple languages interoperate, or more precisely, can programs from multiple languages interoperate? While we all have examples of some interoperation, sometimes even in one environment (Fortran and FAP is an example from the sixties), the general case seems dim. With a language comes a panoply of support structures, libraries, and tools, and these will depend in detailed representation choices. Languages with much semantic content are unlikely to agree on a common underlying representation.

At a high- or mega-level interoperation may be feasible. Components at that level use dedicated processor nodes, and their linkage is via communication interfaces. Since for communication over networks representation changes are nearly always needed, the cost of changing language-motivated representations may add relatively little overhead [GelCar:92] [WWC:92]. This concept can be seen as a scaling up of the message-driven object paradigm. Many overhead costs reduce relatively in that scaling, although global program optimization will require new approaches [Wiederhold:92].

We see such language interaction today among programs, say written in Fortran and database systems, written in C. They communicate via a standard language, SQL, that would be nobody's choice today, but is crucial, since it provides a common ground [Stonebraker:89]. ARPA is sponsoring generalizations of the concepts as Modular Interface Formalism (MIFs) [AkVaVe:80] [CalPur:91] for interfacing software components, a Knowledge Interchange Format (KIF) for sharing represented knowledge, and a Knowledge Query and Manipulation Language (KQML) [NFFGPSS:91] to establish a multi-representation asynchronous transaction protocol.

6. Conclusion

While summarizing government, and specifically ARPA, involvement in standards we have seen a number of areas where government interaction can be easily destructive, and some topics where it's support is proper. In several phases of the standards process government's proper participation seems unclear. These open areas comprise the dissemination of comparative metrics, early adoption of standards to provide leadership, and the partitioning of intersecting standards for interoperation.

The object paradigm raised its head several times, both because of the intense interest in this technology and because its encapsulation concepts indicate a means to deal with interoperation of competing standards.

As Department of Defense (DoD) purchases become a smaller part of the economy, and as DoD moves to greater reliance on commercial products, it is likely to become a more evenhanded and cooperative participant in the standards processes.

References

[AkVaVe:80] K.A. Akopyan, E.A. Vartapetov, and I.V. Vel'bitskii: 'Description of a Methodology for the Design of Automated Information Systems'; *Cybernetics* (translation of *Kibernetika*, Vol.15 No.3, May-Jun.1979), Jan.1980.

[BiDeWTu:83] Dina Bitton, David DeWitt, and Carolyn Turbyfill: 'Benchmarking Database Systems: A Systematic Approach'; *Very Large DataBase 9*, Morgan Kaufman pubs., Los Altos CA, 1983, pp.8–19.

[CalPur: 91] J. Callahan and James Purtilo: 'A Packaging System for Heterogeneous Execution Environments'; *IEEE Transactions on Software Engineering*, vol. 17, 1991, pp. 626-635.

[Carlson:92] William E. Carlson: 'Protyping: An Engineering Approach to Software Development'; Darpa Software Technology Conference 1992, April 1992, Meridien Corp., Arlington VA, pp.428-432.

[GelCar:92] D. Gelernter and M. Carriero: 'Coordination Languages and Their Significance'; *Comm. ACM*, vol.35, pp.96-107, Feb.1992.

[JTEC:92] *Database Use and Technology In Japan*; Japanese Technology Evaluation Center, Loyola College in Maryland, Baltimore MD, April 1992.

[NFFGPSS:91] Robert Neches, Richard Fikes, Tim Finin, Tom Gruber, Ramesh Patil, Ted Senator, and William R. Swartout: 'Enabling Technologies for Knowledge Sharing'; *AI Magazine*, Vol.12 no.3, Fall 1991.

[OMG:92] Craig Thompson (ed.): *Object Services Architecture*; Revision 6.0, OMG Document 92.8.4, Object Management Group (DEC, HP, Hyperdesk Co., NCR, Object Design, Sunsoft) August 1992.

[Stonebraker:89] Michael Stonebraker: 'Future Trends In Database Systems'; *IEEE Trans.on Knowledge and Data Engineering*, Vol.1 No.1, Mar.1989.

[WeBlTh:92] David Wells, José Blakeley, and Craig Thompson: 'Architecture of an Open Object-Oriented Database Management System'; *IEEE Computer*, October 1992.

[Winner:93] Robert Winner (task leader): *Information Infrastructures for Integrated Enterprises*; Institute for Defense Analyses, Alexandria, VA, IDA paper P-2664, May 1993.

[Wiederhold:92] Wiederhold, Gio: "Model-free Optimization"; Darpa Software Technology Conference 1992, April 1992, Meridien Corp., Arlington VA, pp.83-96.

[WWC:92] Gio Wiederhold, Peter Wegner, and Stefano Ceri "Towards Megaprogramming"; Comm. ACM, November 1992.