

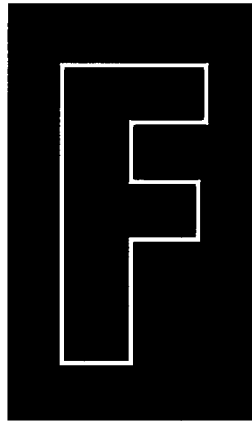


The Business Benefit of Standards

Graham B. Bird

THE OPENGROUP

■ An interesting challenge is currently in vogue: “Is there a business benefit to standards?” First, a resounding answer to the question: Yes! Then an assertion: “for most people and organizations in the value chain, the business benefit of open standards is, and remains, latent.” Turning the benefit from latent to real is up to you. If you don’t do that, it is like collecting only part of your company’s outstanding invoices. Now for the questions: Who benefits? How do they benefit? Why is there so much skepticism about standards? Where does it all go wrong? What should we do to really gain business benefit from standards? The purpose of this article is to offer convincing, and possibly compelling, answers to these questions.



First the standard definition. All standards articles, good or bad, seem to require an opening definition in order to begin. I join in the craze only because it is necessary (which is probably why all of the other articles do the same). The reason that it is needed is simple: public perceptions and available definitions are so widely varying that both international standards AND proprietary interfaces, along with every possible variant in between, are believed to be “standards.” As an aside, we should ask ourselves who is to blame for this appalling state of affairs? We—the entire information technology industry—are. We allow the label “standard” to be applied, often blatantly misapplied, to any product, without question or challenge. (While it is easy—and appealing—to blame “marketing” for this, it is just as true of the technical side of the house.) We will continue to confuse buyers of our products and services—or even our concepts—as long as we allow it to continue. Let’s get a grip on the problem and make a strong start. I offer the following (and oft cited) definition.

An *open standard* is a publicly available specification that is developed and maintained by an open, public consensus process and that is consistent with international standards, where relevant. Additionally, an “Open System” is one built to conform to one or more open standards.

The definition, I am aware, carries the term “standard” in two places—but there is a reason. Let’s look in more detail at that, particularly the accessibility of the standard and the control of the standard. First, any standard must be available to be implemented in product without encumbrance, no royalties, no excessive charges to gain access to the document. Secondly, the standard must be evolved through a known and predictable process that is open to input and influence by all interested parties. Those are the key principles of an *open standard*. The key to the definition lies in understanding that an open standard does not describe a product, but rather, a class of products are built to conform to a standard. (It is a point that is often overlooked.) The difference is that the *open standard* is one which is used as a basis for producing interoperating products from a large num-



ber of providers—who can compete on any of a multitude of competitive advantages to the market buying their product. Many standards (international and others) have an unfortunate tendency to contain “maybe” bits (the result of seeking consensus by accepting multiple options), which make heterogeneous interoperability difficult. Sorting out the maybe bits—that is, selecting which options are needed in a business—changes a standard from a document describing possibilities to one that describes a solid interface that can be used by heterogeneous vendors to satisfy a multitude of user requirements.

Conversely, any specification (or product) that is owned and its development controlled by a single commercial entity is proprietary. It is, of course, much easier to provide and implement a proprietary solution—in a world where the needs of an entire business, including the computing environment, can be met (in the past, present, and future) by a single provider.

Therefore, open systems (products) are those that are built to—and ideally guaranteed to conform to—open standards.

Who Benefits?

There is potential benefit for buyers, software authors and system vendors. I'll look at each briefly.

First and foremost, buyers and users of open IT systems must benefit. If they do not benefit—or believe that they do not and hence don't use them—then all our work is for naught. (While this may appear axiomatic, it is a point often overlooked by many participants in the standardization process.) Standards based products (open systems) offer the following benefits for the taking:

Increased Flexibility

For example, the ability to move applications and data from one system and another is vital. There is an old saying in the construction industry which asks “When is a bulldozer not a bulldozer?” The answer is, of course, “When it's in London and it's supposed to be in Lincoln”. The same is true of information—but with the added imperative of time, as well as location. Information can perish over time, where time is sometimes defined in seconds. There is nothing more irritating than receiving necessary and valid information after a decision has been made—usually, it turns out, wrongly.

Similarly, there is the imperative organizational need for systems from different vendors to easily and simply exchange information in real time. The average business organization in the United States has between 11 and 14 operating systems deployed and in use at any time. These include legacy systems, desktop systems, specialized scientific workstations, and newly deploying systems. Heterogeneity in business computing is the normal case, and it is one of the major reasons for the success of the Open Systems market.

As an example of a successful implementation of open systems, one need look no further than the Internet and the World Wide Web. Neither could exist without open standards. (Yes, to stretch a point, it would be possible to create the Internet using proprietary standards, but the choice of applications would then be limited to the imagination of a single company, rather than the several thousand communications experts who gather to make the Internet work and the hundreds of companies who create and manage web sites using standards. It would also be a different environment—probably a regulated monopoly, similar to that of the telephone companies before the AT&T decision by Judge Greene.) With a single proprietary provider, you could also be faced with unwelcome economic choices—such as possibly having to pay a tax to send your data in someone's proprietary format.

The simple response—yes, but the market won't let that happen—is probably correct. But a more likely scenario is that the web would never be developed unless the standards had been open in the first place, primarily because the people who now cooperate on the Web would still be competing to see who would control the market with “their” proprietary format. (Just look at the state of “set top boxes” today.) There are multiple competing formats—and none seem to be widespread. Openness is a necessary pre-condition to success in business and the market.

This applies only to businesses that may change in an unpredictable way or as a response to an external market change. If a business's needs are stable, slow to change, and predictable, then there is the leisure of being able to use a proprietary product set supplied by a vendor who will change at your pace and at your request. Unfortunately, this scenario seems to be one that is less and less relevant; however, the pattern of trusting one's computing environment to a single vendor seems to be a pattern that continues to plague the business world.

Freedom of Choice

Open systems can also provide a necessary freedom of choice in selection of systems from competing suppliers today and—more importantly—tomorrow. There are very few purchasing agents who do not insist on a “second source” for important components of their product line—yet how many users demand a “second independent source” of computing for their organization? (If you stop to think that even paper towels are second sourced, but information creation, flow, and access aren't, you can get cold chills up and down your spine.) Freedom of choice provides the ability to choose a strong local supplier for each of a multitude of national and regional organizations of a multinational company (as yet no supplier has uniformly high and truly acceptable levels of supply, service and support on a global basis) and have the heterogeneous systems work together. Because busi-



ness conditions also change, the freedom of choice (second sourcing) also allows you to switch suppliers when needed because your business (or worse, theirs) changes. To buy the best solution from the best supplier at any given time and have all of the parts interoperate is only possible with open systems.

Lower Costs of Integration

Putting together complex systems requires one of the following three approaches: single vendor solutions, either a sufficiently massive business benefit, or complete lack of business requirements in order to justify the cost of integrating non-standards-based products, or the ability to integrate products relatively simply because they are built to common, open standards. All three are viable in certain areas.

Single vendor solutions are, as noted above, good for organizations that enjoy the luxury of a simple, predictable, and usually not too dynamic business environment.

The massive business benefit is usually associated with a spectacularly expensive and large budget activities (high potential returns also have commensurately high risks)—which have become more and more difficult to justify in today's world. Even when the risk is justified, the costs, effort, and time to market are all positively impacted by the deployment of an open system.

The lack of a business requirement is usually used in situations where there is a specialized need, usually a governmental program of some sort, most often with weapons systems purchases. However, the drive in many governmental procurement activities is to common off-the-shelf products—from hammers to computer systems. This newer approach (buy specialty items only where the commercial sector does not meet specifications) is becoming increasingly common as the public sector sees reduced budgets and increased demands for services. And this leads to the phenomena of the final option.

Buy goods that are reasonably simple to integrate because they are built to common, open specifications, and have been proved to interoperate.

Only the latter makes sense in today's business environment if the other benefits listed here are to be obtained.

Easier, Simpler Purchase

Using open standards as the specification for systems purchasing decisions can make the procurement cycle much shorter and much less complex. For an example (with which I am exceedingly familiar), requiring bidders to offer a UNIX 98 licensed product is the same as asking for POSIX.1, POSIX.2, ISO8859-1, and so on. Actually, over thirty separate standards are accommodated within, and guaranteed by, the UNIX 98 brand. And now the benefit of what I've described as "open systems" begins to emerge. The benefit from having a suite of standards merged into a product standard produces both significant time and cost

savings, which are the result of shorter, simpler procurement documents, a briefer procurement cycle, and a much easier "apples for apples" comparison of the offers tendered.

The evaluation of subsequent bids is also easier. If all suppliers are *required* to offer a product that is guaranteed to conform to your set of standards, then attention can focus on the business-specific aspects of the purchase. To use the car analogy (it also seems to be compulsory in standards articles), when buying a vehicle, you first work out what it is that you intend to do with the thing you're buying. The difference between a Volkswagen and a tractor trailer is significant—yet each provides "transportation." Once you decide what you want, you create a set of specifications that make certain assumptions about what it is you're buying. In a car, you assume four wheels, brakes, an engine, and so on. You do not ask about the technical aspects of the engine from the metallurgy to the integration of the Bosch injectors and the Champion spark plugs; you assume that that is done for you. You simply ask for the V6, 150hp version (or whatever) and then get on with the more important tasks of working out if the family will fit in it comfortably, the car will corner well, the fuel economy and type of fuel needed, and whether the car will perform the job for which you've bought it. It rarely occurs to ask if the engine and suspension and electrical systems are integrated. This is expected. Why should IT systems be different? They are a tool that is meant to be used like other management tools.

Software Authors and Integrators

Software companies (and software developers inside companies) should be investing their scarce and expensive human resources in providing the functionality that customers and users want and which differentiates their product or service. This is especially true of large firms that have acquired or inherited other companies or divisions, and are now trying to make the systems interoperate. While they realize that they should be creating added value on top of what was supposed to be a "standard" system, what they find is that they are more likely to be involved in "porting" or resolving gratuitous differences between products that should be (and which claim to be) compatible. Rather than help their company, they spend time and effort re-testing components that someone else built to "standards" to ensure that the products that they use really do comply.

The message for integrators is that the potential benefits of open systems are real and substantial. The problem is that, instead of spending time actually integrating systems, a large part of the work seems to be focused upon why the systems don't integrate and then fixing the "don't integrate" problems, rather than focusing on the gains that are available. Non-conformance and gratuitous differences in the way products work do nothing more than add to the cost base of



the work—without adding anything except frustration and, occasionally, cynicism. It is within the industries' grasp to control and manage such issues, but we have to take positive action to turn the latent benefit into real returns.

System Vendors

For system vendors the issues are somewhat the same—but with an additional benefit. Just as the open system gives the purchaser an expanded base from which to purchase, the open system gives the provider an enlarged market in which to sell. This benefit is often overlooked by providers who want to establish a proprietary specification, and hence proprietary product. The market for a product grows as more users come to it—and as more users come to it, more uses for it are discovered—feeding a growth cycle that most proprietary products never achieve. While many providers fear the “commoditization” of their product, one need look no further than the automobile industry; all cars come with four wheels, a steering wheel, brakes, and so on—usually all standard equipment. And yet, there are few who do not know the difference between a Toyota and a BMW, or a Chevrolet and a BMW. The differentiation lies in the brand, and the differentiation that users want, not in gratuitous technical differentiation. In today's marketplace, vendors should be adding value, not struggling with the extra cost of creating incompatible components. The added value comes from product features that help them win customers, from more reliable products and more relevant services, not from making their operating system different so that customers find it more difficult to move applications, data and users. If you don't believe me, I challenge you to drive an older Bentley (not that old, for it's well within my lifetime) that has the gas, brake and clutch pedals (remember clutch pedals?) in a different order, with the gas pedal in the middle.

Achieving the Vision—Making It Real

First and foremost, it's all there, waiting to be used. The effort involved in the extra step is merely an extension of what is largely necessary to implement and run IT systems to the benefit of the business—a goal to which all commercial (and a goodly number of governmental) organizations ascribe.

The recipe goes like this: Define your organizational and business needs—both present and future. Define the functional requirements that the organization must have satisfied. Define a set of standards that meet your business needs. This will usually be a combination of formal standards, publicly available standards, and (last but by no means least) a set of (internal) standards for your business, e.g., your business application logic. The failure to define the business needs and strategy for the IT group—as a derivative of the organizational goal—puts the buyers at the mercy of the IT vendors. Vendors will establish their own standards, making the flow of organization-

al information conform to their capabilities, not to user needs. This is somewhat akin to a supplier limiting the products that a firm can establish because of the suppliers' limitations. (To use the car analogy again, it is as if the brake supplier were to limit the size and power of the engine because they could not build a brake to stop a car going more than 45 miles per hour.)

Next: Implement a strategy of buying and implementing products that conform to the specified standards. To verify that you are in fact buying conforming products, it is necessary to have a method of measuring conformance. This can be either internal tools and validation (very costly) or a warranty from the vendor that their products *do* conform. (The trouble with a vendor-based conformance statement is the “warm and fuzzy” feeling that the vendor is telling the truth.) At the current time, there is one mechanism that does provide such a warranty: the “Open Brand” from The Open Group. The Open Brand indicates that the provider has passed a set of conformity assessment tests, and has guaranteed to keep products conformant to the Open Standard for the life of the product. The guarantee of continued conformance is the value add of the brand, and the commercial rationale for The Open Group's brand program.

Finally, a little internal discipline: make sure that your developers use only standard APIs and services from the standards base you define. The catch here is that it is all too easy to use services or parts of products that are outside your own standards envelope—and this is when costs starts to mount. One of the major difficulties here is that the majority of the total life cost *is not* apparent at this stage. It only becomes visible when a change needs to be made—to the application, to move data, to integrate a new system or piece of functionality. It can be from several months to several years later when the cost of these earlier decisions has to be paid for. And, of course, the person making the decision today is focused on (and probably measured on) completion date, project cost, and delivering the functionality. Because the person who made the original decision has probably moved on, the present and future manager is faced with an impossible situation. By insisting that deferring costs (short term versus long term implications of change) is unwise, and that “buying to standards” is necessary, some of the problems (but not all) can be mitigated. This is one of the places where the discipline can only hope to make the plan fool resistant, not fool proof.

The Payback

Those organizations that implement policies similar to that outlined above—NASA is a good example—have dramatically shortened the procurement cycle, increased flexibility and fit to their mission, and allowed their people use of the latest and best technology. In a larger sense, the requirement is that the



organization must take charge of information management strategy. Management of information is one of its primary strengths, and "information" is one of its largest competitive advantages. Using "open systems and standards" as a buying vehicle will allow the organization to create an "information management tool"—similar to the quality program or the financial and production

systems to help the organization change and succeed as the external environment changes. Those firms that succeed in rationalizing their IT purchases using "open standards" as a systems procurement approach will still be in business; those that don't, won't. **SV**

Reprinted with permission of the author.

PROCEEDINGS

Title: SIGCSE 99
Technical Symposium on Computer Science Education
ISBN: 1-58113-085-6 ORDER#: 457990
Member: \$30.00 tentative Nonmember: \$60.00 tentative

Title: SIGCSE 98
Technical Symposium on Computer Science Education
ISBN: 1-89791-994-7 ORDER#: 457980
Member: \$30.00 Nonmember: \$60.00

Phone: 1.800.342.6626 (USA/Canada)
+212-626-0500 (outside USA/Canada)

Fax: +212-944-1318

Internet: <http://store.acm.org/acmstore/>

Postal Address:

ACM Member Services
P.O. Box 11405
New York, NY 10286-1405 USA



CONTACT ACM and ORDER TODAY!