OPEN SYSTEMS: The Key to Achieving Portability, Interoperability and Affordability

By
Boniface C. Nwugwo

February, 2000
Introduction

Some of the factors that will drive the definition and design of future information systems are portability, interoperability, and affordability. These factors are being addressed on many fronts such as standardization, software reuse, and hardware reuse. In the future, when everyone has a home computer, a television, telephone, fax machine, video conferencing, Internet access and any other service, all these services will need to be integrated. Providing such integrated services requires that we design and build products based on the "open systems" approach today as opposed to "closed" or proprietary systems.

Designing commercial and military systems based on open standards makes economic sense in today’s environment of rapidly evolving commercially based technologies and declining budgets. The need to upgrade systems in a modular, cost-effective, best value manner has never been greater. Using widely accepted commercial interface standards as opposed to proprietary standards provides the added cost-avoidance benefits of leveraging a large market base. This paper discusses Open Systems principles as the key to portability, interoperability and affordability, and identifies the positive and negative aspects of open systems.

What is an Open System?

Several definitions and interpretations of the term open system exist today. Although no formal agreement on any one definition exists, the various definitions have many elements in common. For the purposes of this paper, the definition of open system that will be used here is
the one endorsed by the Joint Task Force (OSJTF) and Technical Architecture Framework for Information Management (TAFIM) of the United States Department of Defense, and provided by the IEEE. The IEEE P1003.0/D15 standard - Draft Guide to the POSIX Open Systems Environment defines open system as:

"A system that implements sufficient open specifications for interfaces, services, and supporting formats to enable properly engineered applications software:
(a) to be ported with minimal changes across a wide range of systems,
(b) to interoperate with other applications on local and remote systems,
(c) to interact with users in a style that facilitates user portability."

(Open Systems Acquisition, 1996, p.4).

Use of the term “sufficient” in the definition is an indication that degrees of openness may vary. There is an old saying that goes like this, “I may not know art, but I know what I like.” This is a saying that can easily be applied to open systems. If you ask ten people about their definition of “open”, you are certainly bound to get ten different answers. To some, open system means plug-and-play. For others it might mean integration with higher-level computer systems. To the military it might mean incorporation of the best commercially available technologies into their weapons systems, cheaper, faster and better. To others it may mean strict adherence to a standard.

Whatever the definition is, in almost all definitions of open system, vendor independence, non-proprietary, publicly available, and widely accepted are some of the key common properties. Ideally, open systems represent a transparent environment in which users can inter-
mix hardware, software, and networks of different vintages from different vendors to meet differing needs.

Interoperability, portability, and affordability are some of the many good characterizations of an open system, which reflect economic goals of open systems. Interoperability for example, deals with the ability of two systems that are not similar to exchange and use data. A typical example of interoperability is the network paradigm. Because each system adheres to some agreed upon standard interface, dissimilar systems are able to communicate with other systems they want to interoperate with. Portability on the other hand, deals with the ability to reduce cost and schedule efforts associated with moving some functionality to another platform. Non adherence to standards means very costly effort when porting software to platforms that it was not designed for.

Applying open systems principles to the systems engineering process helps achieve an integrated design solution that is resilient to changes in technology throughout the life of the system. Engineering systems according to the following principles and practices would help achieve portability, interoperability and affordability.

**Open Systems Principles**

Open systems are built (engineered) and not bought. Open systems do not just happen, but can only be achieved with good engineering discipline. As part of a disciplined systems engineering process, choices for interfaces should be made after considering programmatic, performance, portability, interoperability and affordability requirements. Discussed below are some of the open system principles.
Identify and define the interface

Interfaces must be identified and defined early in the development cycle. Implementation decisions or rush to implement syndrome should be delayed as long as practical. The system architecture should be addressed early in a program to maximize the number of potential solutions, and thereby help reduce program cost. By developing the architecture early in a program, the specific technology used in its implementation can then be chosen as late as possible.

Because open systems engineering is relatively a new discipline, lack of experience in using an open systems engineering approach can lead to inadequate planning, budgeting and scheduling activities that can impact open systems engineering discipline. Although each program has its own priorities and constraints, which may mandate engineering compromises, the rush to implementation which can cause significant risk to the “openness” of the resulting product must be avoided as much as possible.

Define Criteria for Interface Choices

Criteria for interface choices should be defined. At a minimum, critical interfaces must be specified and defined. An approach for doing this has been suggested in a paper presented at the proceedings of 1995 avionics conference by Roark et al. The approach is based on a three dimensional representation where the three axes represent the measures of an interface's openness, maturity, and applicability (Roark, et al, 1995).

Openness includes the technical and source aspects of the interface standard. The technical aspect considers interoperability,
scalability, and portability attributes of a standard. The source aspect considers whether the standard is an international standard, national standard, consortium standard, defacto standard, military standard, or proprietary.

Maturity measures the implementation of a standard from a business perspective. Maturity considers how widespread the use of the standard is by looking at the number of vendors and users of the standard, age of the standard in terms of COTS (define COTS here) availability, and the certification/conformance testing available for the standard.

Applicability of the interface standard to the system of interest measures whether or not an interface standard meets the minimal system requirements. If an interface standard does not meet the minimal requirement, it can be dropped immediately from further consideration. The preferred solution can be determined by evaluating the candidate interface standards against these three criteria.

**Consider system evolution when identifying interfaces**

Identify firewall interfaces for managing evolution and complexity. Since interface choices can determine the impact that a change in one part of the system can have on other parts, firewall interfaces can be used to isolate the impact of such changes. It eliminates dependencies, which also reduces system complexity. The key here is to evolve the system with different levels of granularity. In other words, incremental approach should be used. Firewall interfaces that support small steps in evolution such as increasing memory density on a board should be used. Also, firewall interfaces that support
larger steps in evolution such as replacing a module with a new processor should be used (Roark and Kiczuk, 1996).

**Consider Reusability when identifying interfaces**

A framework for reuse should be identified early within a domain. Development and production costs can be reduced via reuse. Reuse is achieved by "identifying critical interfaces that support portability and interoperability, by identifying interfaces that define firewalls for evolution, and by identifying interfaces that are widespread within the domain of interest." (Roark and Kiczuk, 1996, p.3). In order to get the biggest bang for the buck from reuse, we need to define and maintain a catalog of preferred standards for specific domains. These preferred set of standards or best practices can provide input into the systems engineering process as a starting point for interfaces to consider.

**Use Open COTS Products**

Critical interfaces need to be managed to minimize the risk of using widespread, commercial off-the-shelf (COTS) products and then become unknowingly dependent on a specific vendor's implementation. This is not to say that COTS products should not be used, rather, they should be used carefully. The best way to use COTS is to define the Open Systems architecture first and then evaluate the COTS to determine the ones that are open. The use of non-open COTS leads to vendor dependence whereas use of open COTS increases chances of vendor choices.
Conduct Market analysis when choosing interface standards

When choosing interface standards, a market analysis needs to be performed to determine where future support would be. Strategic supplier alliances with vendors should be considered to insure vendor support for maintenance and integration testing of the off-the-shelf products. Make sure that products (both bought and self-produced) comply with standards to insure interoperability. One of the criteria for selection of a standard should be the ability to test for compliance to a standard. Innovative approaches should be used early in the engineering process to ensure that the system will meet performance and environmental requirements.

Positive and Negative Aspects of Open Systems

Open systems have evolved in the last decade from the favorite playground of academics and technicians to the mission-critical business processes of financial institutions, government and industry. With this evolution comes certain positive and negative aspects of open systems. The remainder of this paper summarizes some of the positive and negative aspects of open systems.

Positives

Open systems are designed to improve performance and lower costs by taking advantage of competition and innovation in the commercial market. They mitigate obsolescence by facilitating technology insertion. Easier technology insertion leads to state-of-the-art systems.

Open systems lead to reduced life cycle cost. The ability of systems to exchange or use information by adhering to a standard functional interface, or be used interchangeably has a major impact on
modification, repair and replacement strategy. It is also possible to
gain reduction in early investment because of the availability of some
of the required components. The military establishment has more to gain
here. The new technological and economic revolution has made the
commercial sector the technology drivers. Since the military no longer
drives technological innovations, they can leverage commercial support,
and invest little or nothing to create and maintain it itself.

Open systems breed competition. Commercial businesses have one
thing as a motivator, profit. To be profitable, a company has to have
better products, stand behind its products, introduce new features and
be competitive in price. Consumers (including the department of
defense) can take advantage of this competition in the commercial
environment to buy better products at a lower cost.

Negatives

Along with the acceptance of open systems comes the specter of
computer crime. The infrastructures of many networks have enough cracks
in their armor to allow computer criminals access to critical data and
processes. The perceived risks are so great that despite their
widespread acceptance, networked open systems form an unacceptable
business risk.

The irony about open systems is that our adversaries (real and
potential) know and have access to the same technology that we do. Open
systems levels the playing field to the point that the competition has
the potential of surpassing us technologically. This means that we have
to keep modernizing our systems or our adversaries will have more
capability than we will.
It is a sobering experience to accept that standardization process is slow compared to the evolution of technology (state of the art). To get a standard drafted and approved is an arduous process. At the current pace of technological innovations, some of these standards become obsolete by the time they get adopted.

Conclusion

As the preceding discussion has shown, application of open systems principles within open systems based systems engineering process is the key to portable, interoperable and affordable systems. Conformance to standards is the foundation of achieving open system benefits. Without adherence to standards, porting software to platforms that they were not designed for is usually a very costly effort. Without adherence to the principles of open systems, two separate systems that may have a need to share information cannot interoperate. The open systems process is a focal point for integrating affordability thrusts such as reuse, cycle time reduction, and design to standards.

We are entering an incredible new era of product development. Advances in technology are giving us an incredible array of choices for building new products. Consumers are also getting more sophisticated and more demanding in terms of products. Not just consumer products or commercial products, but military applications as well. Not only do consumers have high demand for high-quality, low cost and fast delivery of products that fit their needs, they also need these products to be portable and to interoperate. The solution for these lies in building systems that adhere to the open systems principles.
References


