MOSA Principles

Introduction

As previously described, a Modular Open Systems Approach (MOSA) is both a business and technical strategy for developing a new system or modernizing an existing one. It is a means to assess and implement, when feasible, widely supported commercial interface standards in developing systems using modular design concepts. MOSA is also a significant part of the toolset that will help meet DoD's goal of providing joint combat capabilities required for 21st century warfare, and supporting these evolving capabilities over their total life-cycle.

MOSA is an enabler that supports program teams in the acquisition community to 1) design for affordable change, 2) employ evolutionary acquisition and spiral development, and 3) develop an integrated roadmap for weapon system design and development. Basing design strategies on widely supported open standards increases the chance that future changes will be able to be integrated in a cost effective manner. Designing a system for affordable change requires modularity. An evolutionary acquisition strategy provides a foundation that meets existing needs while providing the capability to meet evolving requirements and threats. An integrated roadmap is a tool for detailing the strategy to deliver a weapon system that is capable, upgradeable, affordable, and supportable throughout its planned life-cycle. MOSA supports achieving the following:

- *reduced acquisition cycle time and overall life-cycle cost*
- *ability to insert cutting edge technology as it evolves*
- *commonality and reuse of components among systems*
- *increased ability to leverage commercial investment*

MOSA Principles Explained

The realization of MOSA benefits is dependent on adherence to five major principles, namely; establishing a MOSA enabling environment, employment of modular design, designation of key interfaces, use of open standards for key interfaces, where appropriate, and certifying conformance. These principles lay the foundation for identification of a set of indicators that could be used by acquisition executives and program managers to assess the progress of implementing MOSA in acquisition programs.

Principle 1: Establish an Enabling Environment

To adhere to this principle, the PM must establish supportive requirements, business practices, and technology development, acquisition, test and evaluation, and product support strategies needed for effective development of open systems. Assigning responsibility for MOSA implementation, ensuring appropriate experience and training on MOSA, continuing market research, and proactive identification and overcoming of barriers or obstacles that can potentially slow down or even, in some cases, undermine effective MOSA implementation are among the supportive practices needed for creating an enabling MOSA environment.

Principle 2: Employ Modular Design

Partitioning a system appropriately during the design process to isolate functionality makes the system easier to develop, maintain, and modify or upgrade. Given a system designed for modularity, functions that change rapidly or evolve over time can be upgraded and changed with minor impact to the remainder of the system. This occurs when the design process starts with modularity and future evolution as an objective. Modular designs are characterized by the following:

- Functionally partitioned into discrete scalable, reusable modules consisting of isolated, self-contained functional elements
- Rigorous use of disciplined definition of modular interfaces, to include object oriented descriptions of module functionality
- Designed for ease of change to achieve technology transparency and, to the largest extent possible, makes use of commonly used industry standards for key interfaces

Principle 3: Designate Key interfaces

The focus of MOSA is not on control and management of all the interfaces within and between systems. It will be very costly and perhaps impractical to manage hundreds and in some cases thousands of interfaces used within and among systems. As shown in Figure 2, MOSA manages the interfaces by grouping them into key and non-key interfaces. It distinguishes among interfaces that are between technologically stable and volatile modules, between highly reliable and more frequently failing modules, and between modules with least interoperability impact and those that pass vital interoperability information. Key interfaces should utilize open standards in order to produce the largest life-cycle cost benefits.

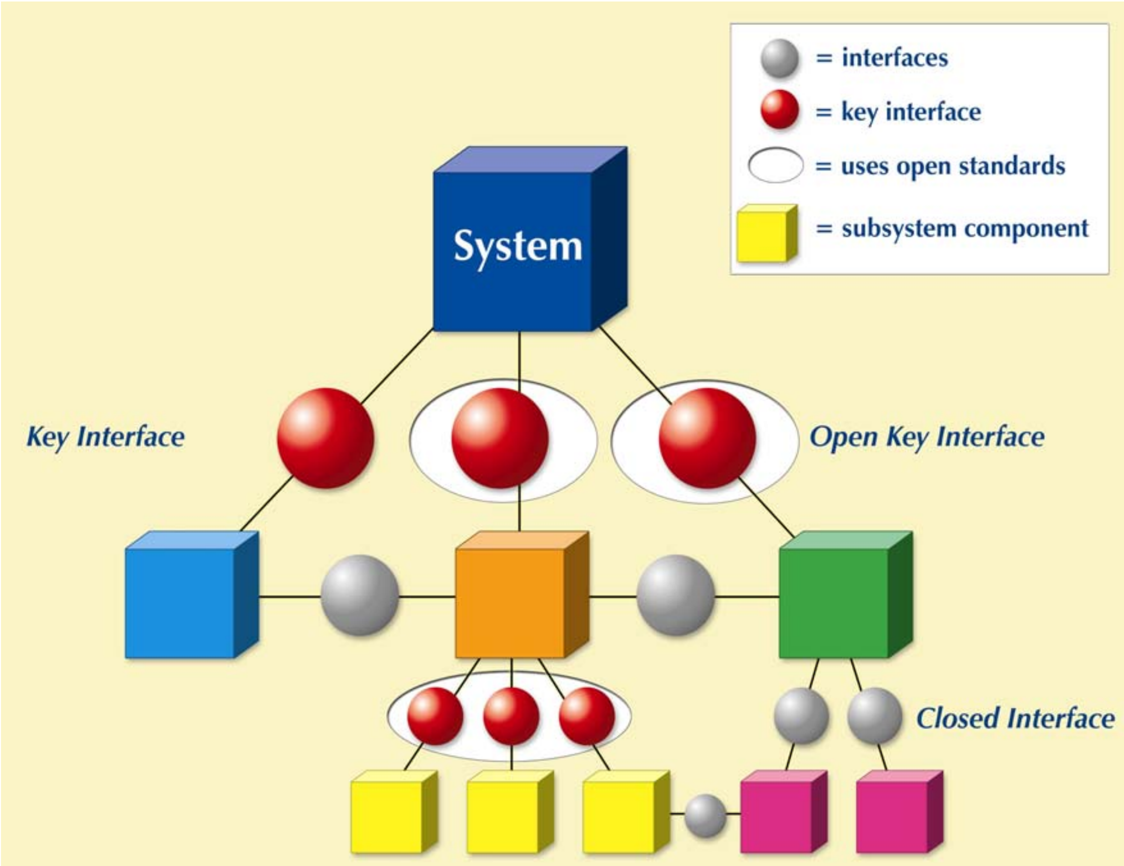

Figure 2: Types of System Interfaces

Principle 4: Use Open Standards

Interface standards specify the physical, functional, and operational relationships between the various elements (hardware and software), to permit interchangeability, interconnection, compatibility and/or communication, and improve logistics support. The selection of the appropriate standards for system interfaces should be based on sound market research of available standards and the application of a disciplined systems engineering process.

In order to take full advantage of modularity in design, interface standards must be well defined, mature, widely used, and readily available. In general, popular open standards yield the most benefit to the customer in terms of ease of future changes to the system and should be the standards of choice. However, there are situations where proprietary standards are the correct choice.

Standards should be selected based on maturity, market acceptance, and allowance for future technology insertion. As a general rule, preference is given to the use of open interface standards first, the de facto interface standards second, and finally government and proprietary interface standards.

Open standards allow programs to leverage commercially funded or developed technologies and to take advantage of increased competition. They also allow faster upgrade of systems with less complexity and cost. Bottom line, systems can be fielded that are more affordable.

Principle 5: Certify Conformance

The program manager, in coordination with the user, should prepare validation and verification mechanisms such as conformance certification and test plans to ensure that the system and its component modules conform to the external and internal open interfaces allowing plug-and-play of modules, net-centric information exchange, and re-configuration of mission capability in response to new threats and technologies. Open systems verification and validation must become an integral part of the overall organization change and configuration management processes. They should also ensure that the system components and selected commercial products avoid utilization of vendor-unique extensions to interface standards and can easily be substituted with similar components from competitive sources.