The Yin and Yang of Complex Open Systems
(Balancing Integration with Innovation)

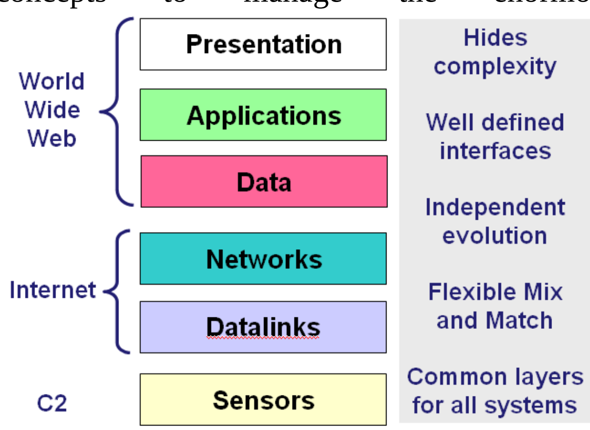
Rich Byrne, The MITRE Corporation

The DOD is faced with two daunting and potentially conflicting imperatives. ***Integration*** is the cornerstone of net-centric operations which acts as a force multiplier by enabling all information to be shared across all systems to achieve superior situational awareness. ***Innovation*** is the cry from today's warfighters as they demand a faster and more agile OODA (observe, orient, decide, and act) loop by adopting new technologies, products, and tactics to respond to constantly changing threats and missions. DOD acquisitions must balance both integration and innovation demands while managing low costs through healthy contractor competition and incentives.

To develop a strategy that balances these objectives, it is beneficial to combine best practices from a number of past successes related to these challenges. Because of the diversity of systems, information technology (IT) intensive systems critical to C4ISR will be the focus of this paper. Given this, key concepts to be adopted include: Layering, Convergence Protocols, and Loosely Coupled Services.

Layering

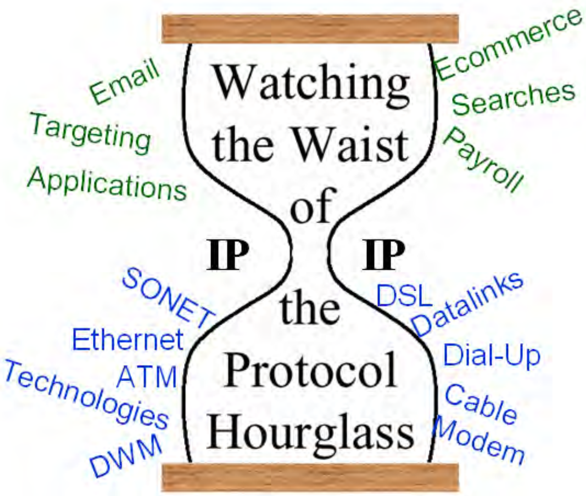
The use of layers is one of the most powerful concepts to manage the enormous complexity of DOD C4ISR systems. The Internet and the World Wide Web are examples that successfully employed layering to hide the internal complexity within one layer from the complexity within a different layer. Layers only communicate through well-defined interfaces. In the figure we show a coarse layering that C4ISR systems could adopt. Innovation in one layer can be independent of innovation inside another layer. Layering also allows a flexible mix and match capability across layers. For example, any application can access any data and send it over any network asset. This provides the agility to string new system combinations to meet emerging missions as needed without major redesign. More layers potentially allow increased agility; however, more layers also means more interfaces to manage. If these benefits are to go beyond single systems and scale across many systems, all systems must have the same definition of layers. If not, these benefits are local only to each system. The layers shown are considered the most important for scalability across the enterprise.



Convergence Standards

Open systems use hardware and software based on standards that are widely recognized, non-proprietary, and freely available to many vendors. These standards promote portability of software between hardware platforms and the communication between software resident on the same or different hardware platforms. Such a philosophy promotes standards as a way of empowering users to avoid vendor lock-in. An example of such an open standard is the Internet Protocol (IP). Many vendors have independently produced interoperable products based on IP, driving costs down and enabling networks across the world to merge together. There are many more market-proven standards for email, graphics, video, etc. The downside of ubiquitously adopted standards is that the tight integration across the many instances of the standard makes it very difficult to adopt the next generation standard. For example IPv6 was proposed as a better alternative to IPv4 over a decade ago, yet today only a few percent of the

US uses it over IPv4. The cost and pain of transitioning every system is so large that the perceived benefit of changing a popular standard must be enormous. Adoption of popular open standards promotes integration, but hurts the agility to change the standard itself. However, offsetting this is the enormous benefit that they nurture innovation above and below the standard. The internet's IP protocol is a classic example of this. Any new or modified application 'above' IP can then be sent around the world on any 'lower' network router as long as all the applications and all of the routers converge to the common IP point of convergence. Similarly, any new or modified router that speaks IP can carry all the existing applications in the world that convert down to IP.



Such 'convergence standards' are extremely powerful. This hourglass or bowtie shape is a fundamental feature of complex systems that are agile. (Complex systems here refer to systems that experience significant uncertainty in their technologies, their environmental constraints, and user needs.) Carefully picking the key convergence standards that have bowtie relationships is a major strategy to balance integration (the common standard) with innovation (done above and below the standard). Note if every detail of the layers has a common standard, there will be no room to innovate; if no standards are present, the plethora of innovation will have no way to integrate.

Loosely Coupled Services

It is instructive to review the evolution of software. Totally integrated code that maximized mainframe performance gave way to a two tier client-server model to better manage complexity. This then gave way to a three tier data-apps-presentation layer approach to improve flexibility of lashing up different data to different apps to different user displays. Today, service oriented architectures are moving to n-tier layers that are customized to different scenarios. The increased number of layers provides increased flexibility, but also increases the number and complexity of the layer interfaces. This has led to the adoption of loosely coupled services which are used to interface to a layer with the minimum amount of knowledge or effort from another layer (or user). For example, a first time user of www.amazon.com can easily interact. This doesn't prevent the Amazon service from providing a much richer interface experience if the user is registered, but it is not a requirement for the unanticipated user. By shifting to very simple, loosely coupled services for each layer, agility to mix and match between layers is greatly enhanced. Even within an application, there is a push to build it as a set of modular components, each of which is accessed via a loosely coupled service allowing new applications to be built on-the-fly by composing these components in real time. Loosely coupled services enable managing the interfaces of increasing use of layers while component thinking allows for even more variety of applications to be rapidly constructed.

Complex Open Systems

Three concepts (Layering, Convergence Protocols, and Loosely Coupled Services) are essential to building the basic framework of IT intensive complex open systems. However, there are literally many thousands of pages of documents that also provide guidance on how to build net-centric, service oriented, or layered architectures. How are these reconcilable since they can often be confusing, overlapping, or even contradictory?

These documents often focus on specific scenarios or certain parts of the overall problem.

As such they are not necessarily applicable equally to all systems. However, the three critical concepts discussed should be present no matter what set of additional guidance is chosen to be followed. For example MOSAS, NESI, STP, RAPIDS, NCIDS, etc all have valuable insights to acquisition models that can be applicable; various combinations will be used for different parts of different systems. It is imperative though that a basic agreement on the coarse layers, the key convergence standards, and the approach to loosely coupled services be the same. These agreements will provide a framework that ensures a minimal integration capability while leaving room inside the framework to innovate, experiment, and optimize for local needs.

Continuous Competition

If DOD adopts these tenets of balancing integration and innovation, contracts can be written to enforce the key definitions of the layers, convergence standards, and definitions of loosely coupled services. This will ensure a minimal set of integration and net-centricity across all programs. Communities of interest may require additional convergence standards and even sub-layers that may not be merited at the DOD enterprise level. Finally, individual systems can specify further standards and sub-layers as locally needed. This hierarchy has a simple relationship: the more systems being governed, the smaller the number of standards and layers that should be mandated. Developing these for different scopes at the DOD, AF, and each community of interest level would be valuable steps towards an open systems vision.

A crucial point to leveraging this framework is the need for the government to ensure data rights to the overall framework, convergence protocols, and loosely coupled service interfaces. It is not necessary (though possibly valuable) to have the data rights of the individual components inside the framework; however, the government must have sufficient data rights to fully test and validate the components independently and be able to substitute other components for 'black box' comparisons. This will enable continuous competition as components can be evaluated for the same function and data driven results used for selection. If two black boxes are comparable, preference should be given for non-proprietary solutions which enable others to innovate on that component as well. Such data rights should include access to any required testbeds, test sets, or procedures needed to validate performance of components 'inside' the framework.

If contracts are created with a pool of qualified contractors and awards are spread out over multiple spirals of capability throughout the contract, then contractors will have incentives to keep their A teams onboard to win subsequent spiral awards. Having a strong government team to objectively assess and choose among offerings based on data driven results will move decisions towards a meritocracy based on real value to the warfighter. Such 'transparency' in assessing and awarding contracts on validated results will ensure early warning on problem areas as well as motivation for non-traditional contractors to compete.

Keys to continuous competition then include agreeing to a common integrating framework, getting the proper data rights to ensure 'transparency', structuring awards across multiple spirals with multiple contractors competing, and having strong technical government teams to arbitrate. These can contribute to a revolution in the responsiveness of our acquisition process.

Summary

The needs for net-centric integration and for warfighter innovation and agility can be met only if there is a delicate balance between standards and non-standards in system architectures. Over-standardization will suffocate innovation; under-standardization will suffocate integration. Coupled with the right contract structures to harness the talents of industry, such a balance will provide an explosion of capabilities to the warfighter through the introduction of new technologies and capabilities while safeguarding essential levels of ubiquitous information sharing.