COLUMNS ⌄ EYE ON INTEGRATION

EYE ON INTEGRATION

The Double Challenge in Engineering Complex Systems of Systems [2007 | 5]

PHILIP BOXER, EDWIN MORRIS, DENNIS SMITH, BILL ANDERSON

Traditional software engineering practices were defined when development was largely controlled by organizations that could set relatively stable requirements, build to those requirements, and deliver a system to the customer. More recently, increasingly complex and dynamic customer demands have focused attention on coordinating activities of multiple organizations and systems within an enterprise to perform a number of tasks or deliver tailored responses. This change in focus from a specific delivered system to the need for flexible capabilities is reflected in product lines, families of systems, and other recent advances in software engineering practices.

However, to meet customer expectations with the emerging, complex systems of systems required to support integrated military strategies, homeland security responses, and nationwide health information networks, system developers must meet a double challenge (see Figure 1):

1. A **governance challenge** of collaborating with an increasing number and diversity of enterprises.
   This challenge includes developing approaches that support cooperation across unrelated enterprises with no unifying controlling structure.

2. An **agility challenge** of providing situation-appropriate responses in changing situations.
   This involves selecting technologies, processes, and structures that are sufficiently agile to support the desired response.
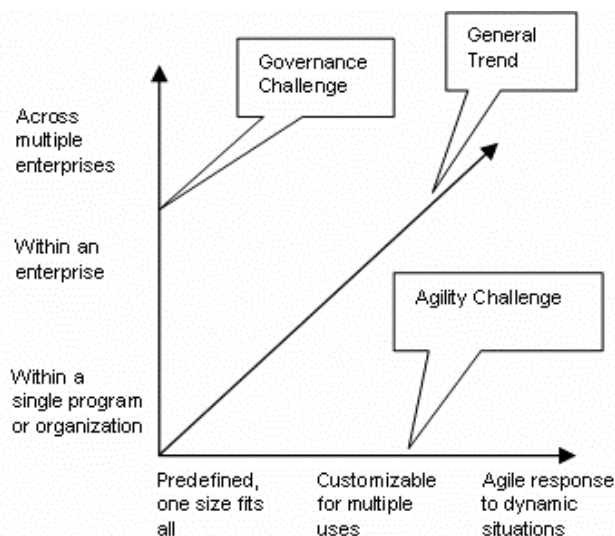

Figure 1: The Double Challenge

The Governance Challenge

The governance challenge involves the changing nature of the collaboration needed to build computer systems. For most of the history of computing, collaboration was needed only to coordinate activities involving point-to-point interfacing of systems within a well-defined program. As anyone experienced with building or maintaining relatively complex software can attest, even this level of collaboration required coordination of the activities of multiple personnel, perhaps with competing interests. However, there was normally one authority that could make and enforce decisions.

In more complex cases, collaboration was needed to coordinate the activities of multiple systems within a single enterprise. This situation was typical of an organization that attempted to relate and integrate the activities of several systems. The need to integrate multiple systems became the driving force behind efforts to integrate data and processes in an enterprise-resource-planning (ERP) system or to construct a common situational picture of the location of friendly and enemy assets by fusing the data contained in multiple military systems operated by the Department of Defense.

Of course, enterprises of any significant size are not homogeneous; they typically consist of multiple levels of relatively autonomous sub-enterprises (e.g., branches, divisions, directorates, teams). However, a hierarchy is assumed to exist that can resolve conflicts in such a way to produce an internally consistent whole.

But system-integration activities become increasingly complex as they cross organizational boundaries. These activities are commonly managed by creating a special management organization with authority that spans these boundaries. The success of these cross-cutting organizations depends on the degree to which they can establish centralized coordination and control of activities—that is, to establish authority over the activities and systems to be integrated as well as over the integration activity itself.

We are familiar with the difficulties this presents to such cross-cutting organizations. The interests of the cross-cutting organizations often come into conflict with those of individual organizations that make up the whole. In principle, these difficulties can be resolved by appeal to the higher authority of the enterprise as a whole. In practice this can be infeasible.

When the cross-cutting approach is extended to multiple enterprises without a unifying hierarchy, a discontinuity occurs (illustrated along the vertical axis of Figure 1). Moving from bottom to top, collaborators become increasingly autonomous and their motivations, policies, procedures, and capabilities become increasingly diverse. In fact, because of the fluid environment in which complex systems must execute, the owners of those systems cannot fully anticipate who these collaborators will be.

To meet the challenge posed by collaboration with an increasing number and diversity of enterprises, processes and strategies must be developed that support negotiation of relationships with these sometimes unanticipated partners. Therefore, we must develop ways of negotiating *collaborative governance* across unrelated enterprises in rapid order.

The Agility Challenge

The agility challenge, illustrated in Figure 1 along the horizontal axis, describes the way that an enterprise's[1] computing systems can respond to the needs of its customers. In simple cases, an enterprise can define the systems it needs to provide to its customers, put together (or otherwise acquire) these capabilities, get them into the right hands, and keep them there. This is the strategy used by enterprises in building tightly focused applications for general use (e.g., Microsoft Word).

In more complex cases, a single predefined system capability is insufficiently flexible to meet the customer's demands. In such cases, enterprises try to set parameters for their applications, processes, and organizational structures that customize them to multiple customer situations. Technologies and systems with customizable interfaces and assembly strategies, such as product lines or families of systems, are employed to increase the range of customer needs that can be met by the organization. Another example is service-oriented architecture (SOA), which is intended to maximize the extent to which services can be composed.

Solutions to these two cases are driven by what the supplier can provide, either in a predefined system or through a customizable set of applications designed to work within some prescribed set of behaviors. However, customers are expecting ever more flexible responses to rapidly changing situations. In this evolving situation, the specific context in which the customer wants to employ a capability becomes a critical determining factor. As a result, it is no longer possible for the technologies and systems to be under the control of specific individuals or organizations. The customer is adding a new level of composition and synchronization of components.

Thus, the agility challenge represents a second discontinuity—between situations controlled by suppliers and situations controlled by the situational needs and demands of users

Along the vertical (governance) axis, interactions between different organizations become increasingly common. This tendency is evidenced in the engineering community by the widespread push for increasing interoperability and standards that support it. It is at the root of efforts to integrate the diverse systems of commercial enterprises into supply chains, to develop common operational pictures across domestic and allied military forces, and to provide emergence response capabilities crossing military, police, government, health care, and other networks.

Along the horizontal (agility) axis, there is increased recognition in commercial, government, and military sectors that advantage is best gained by developing system capabilities that can be rapidly aligned in new ways to support responses to changing demands. For commercial organizations, this means that supplier relationships must meet changing needs—whether by supporting rapid changes to products and manufacturing approaches or by providing new ways for service delivery as determined by the customer. Ideally, the alignment of organizational structures, processes, and systems capability is determined by the demand.

Thus, the general trend is clearly up and to the right in as depicted in Figure 1. However, given the traditional methods of building systems, the comfort zone for building systems is actually down and to the left as shown in Figure 2.
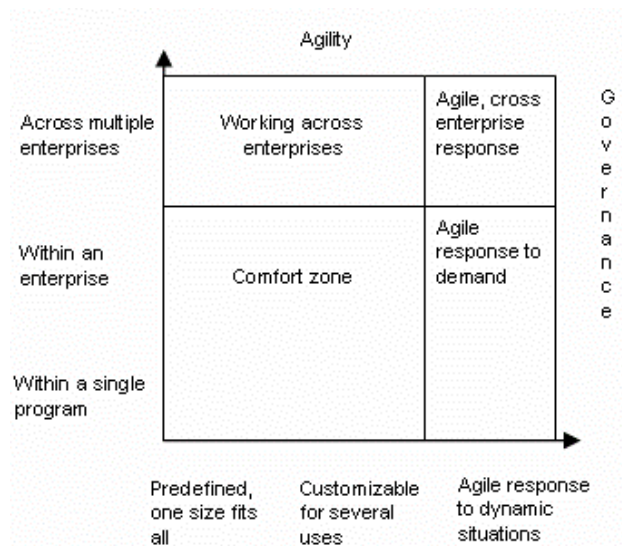

Figure 2: The Comfort and Other Zones

The top right quadrant of this chart involves cross-enterprise and agile response to changing needs and demands of users. This goal is perhaps best developed today in the power-to-the-edge strategy of the U.S. Department of Defense (DoD), which has recognized that only at the edge can it respond appropriately to the demands being placed on it for cross-command (e.g., services, allied militaries, and non-traditional allied) responses.

We believe that the future of software engineering will be dominated by the double challenge of developing governance approaches that can work across enterprises and identifying ways to meet demands for increasing agility in the capabilities that are provided.

We are concerned that focusing on developing engineering strategies to improve or fix individual symptoms of the double challenge, such as poor coordination of development efforts across organizations and problems with configuration management, may be helpful in some situations but ultimately will not solve the problem.

It is more likely that fixes for individual symptoms along one axis of the double challenge will actually complicate problems along the alternate axis. For example, developing a new, virtual organization that imposes a hierarchy across enterprise boundaries may lead to reduced flexibility in response to new user demands. The key question is how enterprises can develop the ability to work across enterprise boundaries while simultaneously providing the agility to respond to the changing demands of the customer.

A starting point involves distinguishing different types of systems of systems based on the type of authority possible and the ability of that authority to control behavior. This approach allows us to

begin to characterize requisite engineering practices. Of particular interest to us are those practices that support distributed collaboration[2] (e.g., power-to-the-edge). A critical activity for the future is to consider the double challenge (i.e., governance and agility) in relation to achieving distributed collaboration.

In response to the double challenge, the SEI is developing the System-of-Systems Navigator[SM], an integrated set of principles, tools, models, techniques, and improvement cycle activities. The SEI is currently developing capabilities to recognize different types of systems of systems along the dimensions outlined above. Of particular interest are those practices that support distributed collaboration (e.g., power-to-the-edge) risk assessment in this complex space through modeling and gap analysis between the required collaborative constituent components.

For more information on the System-of-Systems Navigator, contact Suzanne Garcia (smg@sei.cmu.edu).

---

1 We have used *enterprise* here to mean any sort of entity that must respond to the second challenge. That enterprise may entail one or more organizations.

2 As opposed to directed collaboration that has a controlling authority

About the Authors

Philip Boxer has been a strategy consultant to organizations since the late 1970s, supporting leadership teams across many different industry sectors, both public and private. His focus is on the challenges organizations face from asymmetric forms of demand and the mitigation of risks associated with failing to develop requisite agility. He has developed a number of supporting methods and tools needed to support strategy formation and collaborative design processes, including visual PAN and its associated forms of analysis. He is a senior member of the technical staff of the Integration of Software-Intensive Systems (ISIS) team at the Software Engineering Institute.

Edwin Morris is a Senior Member of the Technical Staff at the Software Engineering Institute, assigned to the Integration of Software-Intensive Systems (ISIS) Initiative. He is currently investigating approaches to achieving technical interoperability between complex systems and programmatic interoperability between the organizations that build and maintain them. Previous activities involved improving processes and techniques for the evaluation and selection of COTS products, and the development of the COTS Usage Risk Evaluation (CURE) technology. Before coming to the SEI, Morris developed custom operating systems for embedded microprocessors along with support tools to predict and monitor the performance of real time systems.

Dennis Smith is the lead for the SEI Integration of Software Intensive Systems (ISIS) Initiative. This initiative focuses on addressing issues of interoperability and integration in large-scale systems and systems of systems. Earlier, he was the technical lead in the effort for migrating legacy systems to product lines. In this role he developed the method Options Analysis for Reengineering (OARS) to support reuse decision-making. Smith has also been the project leader for the CASE environments project. This project examined the underlying issues of CASE integration, process support for environments and the adoption of technology. Smith has published a wide variety of articles and technical reports, and has given talks and keynotes at a number of conferences and workshops. He has an MA and PhD from Princeton University and a BA from Columbia University.

William B. Anderson is a senior member of the SEI technical staff. Bill's research interests include integration and interoperability of complex software systems, COTS and reuse management, cost

estimation, and business case justification of complex systems. A former Vice President for a Fortune 500 company, Bill is broadly experienced with factory floor and business; processes, support systems, automation, and management. He has many years of experience in large system project management and has successfully led operational, financial, product line, and new product launch groups.

The views expressed in this article are the author's only and do not represent directly or imply any official position or view of the Software Engineering Institute or Carnegie Mellon University. This article is intended to stimulate further discussion about this topic.

Return to COLUMNS list