

The COTS Spot

Changing Focus

Patricia Oberndorf

Over the course of the past few years, this column has highlighted a large number of topics important to the successful use of commercial off-the-shelf (COTS) software in software-intensive systems. These topics have ranged from evaluating COTS products to issues relating to modernization of legacy systems. The discourse was generated from the work of the SEI COTS-Based Systems (CBS) initiative, the focus of which has been to learn, mature, and transition principles, methods, and techniques for creating systems from COTS products.

The CBS initiative has thoroughly explored a wide range of issues. Our products have included an instrument and method for CBS risk evaluation (COTS Usage Risk Evaluation, or CURE); a process for developing and sustaining COTS-based systems (Evolutionary Process for Integrating COTS-Based Systems, or EPIC); courses for executives, managers, and practitioners; and technical reports, special reports, and technical notes.

Almost from the outset, we recognized that this tight focus on systems built from COTS products was to some extent artificial: no one builds systems solely from COTS products. Most systems today *include* COTS products, but they also contain components from other sources, such as legacy systems, a “sibling” system with some similar characteristics built by a sister organization, and a development effort that creates a functional component not available in the COTS marketplace. And all of them incorporate some “glue” code that is created to integrate all these components into a coherent system. It turns out that a good deal of what has resulted from the close examination of using COTS products applies equally to using *any* off-the-shelf (OTS) component—the “c” in “COTS” often has little to do with what is necessary to successfully integrate an OTS component into a system.

This realization, coupled with a sense that the CBS territory was largely exposed (even if many problems remain to be solved), has brought us to a crossroads. It is time to step up to the larger scope of improving the creation and sustainment of systems from any set of largely OTS (but not just commercial) constituents. This is a transition from a CBS-focused endeavor to one that considers integration more broadly. It also moves closer to addressing the interoperability and systems-of-systems issues that are so ubiquitous in the world of software engineering today and that are causing so many headaches in so many software projects.

Thus we are announcing the transformation of the CBS initiative into the Integration of Software-Intensive Systems (ISIS) initiative starting in October 2003. While the detailed plans are still taking shape as of this writing, we can outline the intended start and direction.

One part of the initial effort will be to work on terminology and basic concepts: while the essential notions of “integration” and “interoperability” are shared throughout the engineering community, we see the need for greater precision. For instance, are these terms the same? If not, what distinguishes them? (And if they really do connote the same thing, why do we all insist on using them as though they were distinct?)

The rest of the initiative and its work will be founded on several bases in addition to this definitional one. First, we are not intending to relinquish all of the intellectual currency we have gained through our work in COTS-based systems; we intend to put the wisdom we have gained to use wherever appropriate. Second, we have the great benefit of inheriting the work of several research and development projects that have been underway at the SEI for the past year. Perhaps the most significant of these is an effort in system-of-systems interoperability (SOSI); this research effort will provide a head start for our work in understanding the basic issues in interoperability across multiple systems. Another R&D effort during this past year focused on sustainability, and we intend to leverage that work particularly as we grapple with the maintenance problems that confound so many systems developers today: even if interoperability can be achieved between and among systems, and even if we can coerce two or more systems to behave the way we want them to, how do we manage to keep that interoperation going as these systems evolve? How do we cope when components—whether COTS or other OTS—are constantly being upgraded, refreshed, and retired, outside the control of the sustaining agency?

We intend to place a primary focus on the study of existing and planned systems of the U.S. Department of Defense (DoD) and other large organizations, examining them for identification of the important integration problems, for successful methods and techniques that are in use, and for lessons learned that can be shared with others. We will also explore more detailed integration and interoperability techniques, those already in use, and others that may need to be created and matured. We will explore current practice particularly in the areas of component evaluation, architectural qualities, opportunistic evolution, and the management of assumptions. We will also investigate techniques for rapid integration of systems. Another path will ensure that metrics considerations are a part of this work from the outset. And we will examine applicable DoD policies that may foster—or inhibit—the achievement of a desired state of integration or interoperability.

We will consider the proposition that full integration, especially at the system-of-systems interoperability level, depends as much on integration at the programmatic level as on detailed engineering methods and techniques. We expect to explore the use of an acquisition modeling approach to compare the acquisition strategies and plans of two programs and determine how adjusting and harmonizing those strategies and plans can improve program-to-program (and therefore system-of-systems) integration.

We are excited about this enlargement of scope and the opportunity to build on all of the CBS work to create solutions for a broader range of systems. While the SEI will continue to support and transition the results of the CBS initiative, the readers of this column can look forward to articles that explore problems and solutions regarding integration and interoperation. In keeping with this change, we will retire the “COTS Spot” name.

As for our new initiative, we wryly admit that with a name like “ISIS” we could be in for a bit of ribbing: how many of our colleagues can claim the protection of an ancient Egyptian goddess? But for the purpose of this venue, we have decided that a name that is a bit more expressive would be appropriate. So starting in the next issue, look for the new column called “Eye on Integration.” We’re looking forward to an exciting time!

About the Author

Tricia Oberndorf is Director of the Dynamic Systems Program at the Software Engineering Institute (SEI). She has been a principal of the COTS-Based Systems Initiative and concentrates on the investigation of issues in integration and acquisition of systems from commercial and other off-the-shelf components. Early in her career focused on the investigation of integration and open systems questions, primarily in the context of computer-aided software engineering environments. Prior to the SEI, she was with the Navy for more than 19 years. Oberndorf has co-authored a book titled *Managing Software Acquisition: Open Systems and COTS Products*. She received an MS in Computer Science from UCSD and a BS in Computer Science from Oregon State University.

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.

The Software Engineering Institute (SEI) is a federally funded research and development center sponsored by the U.S. Department of Defense and operated by Carnegie Mellon University.

® Capability Maturity Model, Capability Maturity Modeling, Carnegie Mellon, CERT, CERT Coordination Center, CMM, CMMI, and OCTAVE are registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

SM Architecture Tradeoff Analysis Method; ATAM; CMM Integration; COTS Usage Risk Evaluation; CURE; EPIC; Evolutionary Process for Integrating COTS-Based Systems; Framework for Software Product Line Practice; IDEAL; Interim Profile; OAR; Operationally Critical Threat, Asset, and Vulnerability Evaluation; Options Analysis for Reengineering; Personal Software Process; PLTP; Product Line Technical Probe; PSP; SCAMPI; SCAMPI Lead Assessor; SCAMPI Lead Appraiser; SCE; SEI; SEI-Europe; SEPG; Team Software Process; and TSP are service marks of Carnegie Mellon University..

TM Simplex is a trademark of Carnegie Mellon University.