United States Air Force
Scientific Advisory Board

Report on

Ensuring Successful Implementation Of Commercial Items In Air Force Systems

SAB-TR-99-03

April 2000

Cleared for Open Publication February 2000

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (FROM - TO)
01-04-2000	Technical	xx-01-1999 to xx-11-1999

4. TITLE AND SUBTITLE
Ensuring Successful Implementation of Commercial Items in Air Force Systems
Unclassified

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Grant, Jeffery E ; Author
Rankine, Robert ; Author
Brown, Kenneth M ; Author
Carter, William R ; Author
Foreman, John ; Author

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME AND ADDRESS
USAF Scientific Advisory Board
Room 5D982
1180 AF Pentagon
Washington, DC20330-1180

8. PERFORMING ORGANIZATION REPORT NUMBER
SAB-TR-99-03

9. SPONSORING/MONITORING AGENCY NAME AND ADDRESS
SAF/OS
AF/CC
Pentagon
Washington, DC20330-1670

10. SPONSOR/MONITOR'S ACRONYM(S)
SAF/OS; AF/CC

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
SAB-TR-99-03

12. DISTRIBUTION/AVAILABILITY STATEMENT
APUBLIC RELEASE
,

13. SUPPLEMENTARY NOTES

14. ABSTRACT
Taking advantage of commercial off-the-shelf (COTS) products seems like a logical way to achieve significant cost savings. However, based on an assessment of 34 programs and organizations, only a few are realizing significant benefits. Most are struggling with its complexity and a few have failed miserably. The complexities are numerous and less than obvious. Arguably the biggest pitfall of all is inflexible requirements. A balance must be achieved between desired performance and what can be reasonably attained by integrating available and projected commercial products. In total, the COTS Study Panel observed about 25 common pitfalls that programs are experiencing. Most could be avoided or mitigated if appropriate processes or procedures were in place that people understood and followed. Requirements must flow into an architecture that can truly exploit the advantages of COTS. Contractors must shift from ?design and build? unique products to ?buy and integrate? standard products. Everyone coping with a COTS development is on a very steep learning curve and those that seem to do it well have been at it for many years. They freely admit that they have made every mistake imaginable along the way. Unfortunately, others can?t imagine the mistakes they are about to make

15. SUBJECT TERMS
AAAV, Advanced Amphibious Assault Vehicle; CI, Commercial Item; GOTS, Government Off-The-Shelf; JDAM, Joint Direct Attack Munitions; LCC, Life Cycle Cost; MOTS, Modified Off-The-Shelf; NSSN, New Attack Submarine; NDI, Non-Developmental Item; PEM, Plastic Encapsulated Microcircuits; RCOTS, Ruggedized COTS; ROTS, Research Off-The-Shelf; SEE, Software Engineering Environment; TOC, Total Ownership Costs, Spiral Development, Technology Refresh

16. SECURITY CLASSIFICATION OF:

a. REPORT	b. ABSTRACT	c. THIS PAGE
Unclassified	Unclassified	Unclassified

17. LIMITATION OF ABSTRACT
Same as Report (SAR)

18. NUMBER OF PAGES
102

19. NAME OF RESPONSIBLE PERSON
Ripperger, Robert
robert.ripperger@pentagon.af.mil

19b. TELEPHONE NUMBER
International Area Code
011
Area Code Telephone Number
703692-5097
DSN
222-5097

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39.18

United States Air Force Scientific Advisory Board

Report on

Ensuring Successful Implementation Of Commercial Items In Air Force Systems

SAB-TR-99-03

April 2000

Cleared for Open Publication February 2000

(This Page Intentionally Blank)

Table of Contents

List of Figures

List of Tables

1 Summary

Pervasive cultural and procedural changes are needed to exploit the benefits of COTS products.

The United States Air Force faces enormous challenges in dealing with a budget climate characterized by stringency at a time when wrenching changes are required. Many of these changes are described in Joint Vision 2010, the AF Space Command Strategic Master Plan, and other planning documents. Finding reasonable ways of saving money is essential. There are a number of possibilities, but most involve hard choices. Taking advantage of commercial off-the-shelf (COTS) products seems like a logical way to achieve significant cost savings with very little sacrifice. In many technological areas important to the Air Force, commercial industry has assumed the leadership role. In addition, there are other potential benefits including faster deployment time, improved quality and reliability, reduced development risk, and a support system that is already in place. Fielding the most advanced weapon systems requires leveraging commercial products.

However, based on an assessment of 34 programs and organizations, only a few are realizing significant benefits. Most are struggling with its complexity and a few have failed miserably. The complexities are numerous and less than obvious. Arguably the biggest pitfall of all is inflexible requirements. The maximum utilization of COTS products demands flexible requirements from the outset. The user must interact with potential bidders before the Operational Requirements Document (ORD) and Request For Proposal (RFP) are established. A balance must be achieved between desired performance and what can be reasonably attained by integrating available and projected commercial products. During system development trade studies should be conducted to further refine the balance between performance specifications, operational procedures, total ownership cost and the extent of COTS product utilization. The government must require continuous performance trades to maximize the use of COTS products. As an example, the AWACS Computer Modernization Program made considerable environmental changes in the spirit of acquisition reform to reduce overall cost as illustrated below.

Table 1. AWACS Computer Modernization Requirements

Specification	Original	Revised
Operating temperature	-54 to +55°C	0 to 50°C
Shock	15g peak, 11 msec	6g peak, 11 msec
Vibration	Based on Mil Std	Based on measurement
Operating humidity	100%	85%
Salt spray	Yes	No

Rigid requirements may result in relatively few suitable COTS products. Demanding the maximum use of COTS products while constraining requirements flexibility is a recipe for disaster. Something has to give. The government customer must be willing to accept the 80 percent solution. If not, the government cannot count on the much touted benefits of COTS.

Inadequate consideration of COTS product volatility, particularly software, is another common pitfall experienced by many. Integrating forty or fifty COTS software packages, each on an asynchronous 18 month upgrade cycle, is a challenge. A successful integration effort must deal with a constant state of flux.

In total, the COTS Study Panel observed about 25 common pitfalls that programs are experiencing. Most could be avoided or mitigated if appropriate processes or procedures were in place that people understood and followed. Requirements must flow into an architecture that can truly exploit the advantages of COTS. Contractors must shift from “design and build” unique products to “buy and integrate” standard products. Everyone coping with a COTS development environment is on a very steep learning curve and those that seem to do it well have been at it for many years. They freely admit that they have made every mistake imaginable along the way. Unfortunately, others can’t imagine the mistakes they are about to make.

The cultural differences between a traditional custom Mil Spec and a COTS intensive environment are enormous for both contractor and government personnel. COTS demands new skills, knowledge and abilities. The traditional skills that are acquired over many years do not provide an adequate understanding to address the additional complications in selecting, specifying, buying and using commercial products for military applications. Roles and responsibilities change dramatically. As shown in figure 1, processes are radically different. The ramifications of these shifts are enormous. Aerospace and government personnel have been conducting business a particular way for decades. Now we are asking them to do it in an entirely different fashion. Many feel job insecurity and a loss of control .

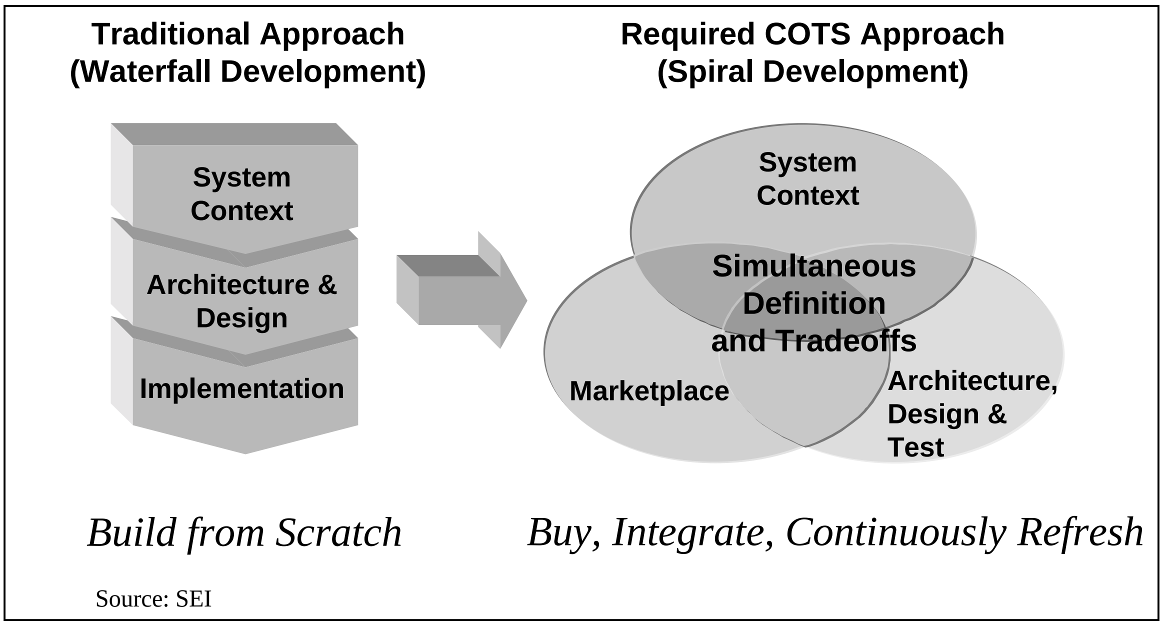

Figure 1. Fundamental Culture Shift

The primary purpose of this report is to capture the issues, pitfalls, myths, lessons learned, best practices and critical success factors associated with COTS. This information should enable the reader to identify either new processes or modifications to existing processes in order to realize the benefits of COTS. Section 5.0 of this report lists specific acquisition, development and sustainment processes that must be addressed. Success is absolutely dependent on good COTS processes that are refined over time. Expecting success by using traditional business practices, as many have learned, is a fool hardy notion.

Of the 24 programs interviewed, 5 were considered exemplary. The exemplary programs exhibited 12 common characteristics that the Study Panel considered critical success factors. The government needs to assure that:

1. All operational requirements and procurement specifications are negotiable.

2. Open system architecture and the spiral development process are utilized.

3. The prime contractor is incentivized to provide a credible estimate of support costs.

4. Total ownership cost (TOC) is used as a source selection cost criterion.

5. The contractors past experience employing COTS products are assessed.

6. The contractor's processes for assessing, selecting, integrating, supporting and refreshing of COTS products are adequate.

7. TOC is used to determine suitability of COTS versus custom products.

8. The contractor's understanding of the commercial marketplace and relevant COTS products are evaluated.

9. The system application matches the COTS product functionality.

10. The contractor proposes to use the COTS product without modification.

11. Trade-off analyses of all changes versus total ownership cost are conducted.

12. There is continuous interaction between government personnel (operations and acquisition) and the prime contractor in Integrated Product Teams.

The COTS Study Panel strongly recommends that these success factors form the basis of an implementation policy that serves as a framework to drive acquisition strategy, source selection, program management and, indirectly, the aerospace industry. In addition, since everyone is on a steep learning curve, we recommend that a periodic or annual review be conducted to incorporate additional lessons learned into the policy until the situation stabilizes. Ideally, the policy should be adopted by DoD to assure uniformity across the services and in keeping with the Single Process Initiative.

(This Page Intentionally Blank)

2 Background

2.1 Definitions/Acronyms

The introduction of COTS products into system acquisition has resulted in new terms and acronyms associated with this process.

These acronyms and definitions were taken in part from the Ninth Edition of GLOSSARY: Defense Acquisition Acronyms and Terms, November 1998, available at the following web site: http://www.dsmc.dsm.mil/pubs/glossary/preface.htm.

2.1.1 List Of Acronyms

ADP	Automated Data Processing
AF	Air Force
API	Application Programming Interface
ASIC	Application Specific Integrated Circuit
AWACS	Airborne Warning and Control System
CAIV	Cost as An Independent Variable
CALCE	Computer Assisted Life Cycle Engineering/ Computer Aided Life Cycle Evaluation
CBD	COTS-Based Development
CBS	COTS-Based System
CDR	Critical Design Review
CI	Commercial Item
CM	Configuration Management
CMM	Capability Maturity Model
CO	Contracting Officer
CORBA	Common Object Request Broker Architecture
COTS	Commercial Off-The-Shelf
COTSCON	COTS Conference
CURE	COTS Usage Risk Evaluation
DCOM	Distributed Component Object Model
DoD	Department of Defense
DMS	Diminished Manufacturing Source
DTC	Design-To-Cost
ESC	Electronic Systems Center
FASA	Federal Acquisition Streamlining Act
FFP	Fixed Firm Price
G&A	General & Administrative
GBS	Global Broadcast Service
GD	General Dynamics
GNIE	Global Network Information Enterprise
GOTS	Government Off-The-Shelf
GPS	Global Positioning System
GSA	General Services Administration
IC	Integrated Circuit
IEEE	Institute of Electrical and Electronic Engineers
IPT	Integrated Product Team
JDAM	Joint Direct Attack Munitions
JPATS	Joint Primary Aircraft Training System
JTA	Joint Technical Architecture
KPA	Key Process Area
LCC	Life Cycle Cost
LMCO	Lockheed Martin
LOB	Line Of Business
LRU	Line Replaceable Unit
MIS	Management Information System
MOTS	Modified Off-The-Shelf
MOU	Memorandum of Understanding
MTBF	Mean Time Between Failures
MTBCF	Mean Time Between Critical Failures.
NSSN	New Attack Submarine
NDI	Non-Developmental Item
NRC	Non Recurring Cost
O&M	Operations & Maintenance

O&S	Operation & Support	ROI	Return On Investment
OCS	Operational Control Segment	ROTS	Research Off-The-Shelf
OEM	Original Equipment Manufacturer	SAB	Scientific Advisory Board
OMG	Object Management Group	SEE	Software Engineering Environment
OP	Obsolete Parts	SEI	Software Engineering Institute
OSD	Office of the Secretary of Defense	SOW	Statement of Work
OT&E	Operational Test and Evaluation	SPC	Software Productivity Consortium
PC	Personal Computer	SPO	System Program Office
PEM	Plastic Encapsulated Microcircuits	T&E	Test & Evaluation
QA	Quality Assurance	TOC	Total Ownership Costs
R&D	Research & Development	UML	Unified Modeling Language
RAD	Rapid Application Development	US	United States (of America)
RCI	Reifer Consultants, Inc	VAR	Value Added Reseller
RCOTS	Ruggedized COTS	WWW	World Wide Web
RFP	Request For Proposal		

2.1.2 List of Definitions

Commercial Off-The Shelf (COTS) - Items which can be purchased through commercial retail or wholesale distributors, as is, and are generally available as a catalog item

Cost As an Independent Variable (CAIV) - Methodologies used to acquire and operate affordable DoD systems by setting aggressive, achievable life cycle cost objectives, and managing achievement of these objectives by trading off performance and schedule, as necessary. Cost objectives balance mission needs with projected out-year resources, taking into account anticipated process improvements in both DoD and industry. CAIV has brought attention to the government's responsibilities for setting/adjusting life cycle cost objectives and for evaluating requirements in terms of overall cost consequences.

Government Furnished Equipment (GFE) - See Government Furnished Property (GFP).

Government Furnished Material (GFM) - Material is government property which may be incorporated into or attached to an end item to be delivered under a contract or which may be consumed in the performance of a contract. It includes, but is not limited to, raw and processed material, parts, components, assemblies, and small tools and supplies.

Government Furnished Property (GFP) - Property in the possession of or acquired directly by the government, and subsequently delivered to or otherwise made available to the contractor.

Computers - The physical equipment that makes up a computer system, e.g., terminals and storage devices, as opposed to programming software.

Harmonization - Refers to the process, or results, of adjusting differences or inconsistencies in the qualitative basic military requirements of the United States, its allies, and other friendly countries. It implies that significant features will be brought into line so as to make possible substantial gains in terms of the overall objectives of cooperation (e.g., enhanced utilization of resources, standardization, and compatibility of equipment). It implies especially that comparatively minor differences in "requirements" should not be permitted to serve as a basis for the support of slightly different duplicative programs and projects.

Modified COTS (MOTS) - COTS items that have been modified to meet a specific form, fit, function or interface requirements.

Non-Developmental Item (NDI) - Any item of supply that is available in the commercial marketplace; any previously developed item of supply that is in use by a department or agency of the United States, a State or Local government, or a foreign government with which the United States has a mutual defense cooperation agreement; Any item of supply that requires only minor modification in order to meet the requirements of the procuring agency; or any item of supply that is currently being produced that is not yet in use, or is not yet available in the commercial marketplace.

2.2 Legislation and Policy

Congress and DoD strongly encourage the use of COTS products.

In 1986 Congress passed legislation requiring the Department of Defense to give preference to non-developmental items. Use of existing, previously developed items, whether commercial or military, saves research and development costs, shortens fielding time, and reduces the risk associated with new development.

The Federal Acquisition Streamlining Act (FASA) of 1994 increased DoD's ability to tap the commercial marketplace. FASA specifically requires that, to the maximum extent practically, contract requirements and market research should facilitate use of commercial items. Preference should be given to non-developmental items (NDI) other than commercial items when commercial items are not available. Contract requirements that impede acquisition of commercial items should be eliminated. In addition, FASA requires DoD agencies to conduct market research prior to development of a new specification and before soliciting bids and proposals for a contract in excess of $100,000 and shall use the result of this research to determine whether commercial items or, if commercial items are unavailable, NDI, or modified commercial or non-developmental items will meet the agency's needs.

The use of commercial and NDI applies to the entire range of goods and services purchased by the defense department. Acquisitions of major weapon systems, basic consumable items, and everything in between offer opportunities for the use of commercial items and NDI to varying degrees. Although complex defense systems may not be manufactured as end items on commercial lines, their subsystems and components may well be.

Federal Acquisition Regulation (FAR) Part 12 implements the Federal Government's preference for the acquisition of commercial items contained in FASA by establishing acquisition policies more closely resembling those of the commercial marketplace and encouraging the acquisition of commercial items and components. It states that all agencies shall conduct market research to determine whether commercial items or NDI are available that could meet the agency's requirements. Commercial items or NDI should be acquired when they are available to meet the needs of the agency. Contractors and subcontractors at all tiers are required to incorporate, to the maximum extent practicable, commercial or NDI as components of items supplied to the agency.

In Defense Secretary Perry's February 1994 report, Acquisition Reform: A Mandate for Change, he also recognized that American companies most dependent on defense business are laying off hundreds of thousands of workers. These jobs will be gone for good unless former defense-only companies can convert to manufacturing commercial products. If DoD does not aid in this conversion, by adopting procurement practices that encourage commercialization, it will lose access to the industrial base upon which it relies for technological superiority.

Perry explained that for years DoD pioneered technological advances in many areas - but today, the tables have turned. Commercial technology advancements are outpacing DoD-sponsored efforts in many sectors key to military superiority (e.g., computers, software, integrated circuits, communications, and advanced materials). From R&D to practical application and production, DoD simply takes too long. The design cycle for commercial technology is approximately 3-4 years; in DoD it is 8-10 years. Many of the advanced technologies DoD implements are grossly obsolete before even fielded. Perry reasoned that to maintain our military superiority, we must gain access to commercial technologies more quickly and more economically than other countries.

Perry concluded that DoD acquisition reform coincides with our most important national goals: saving the taxpayer money; reinventing Government; strengthening our military; and improving our economy. To meet these goals, DoD must:

- Rapidly acquire commercial and other state-of-the-art products and technology from reliable suppliers who employ the latest manufacturing and management techniques
- Assist in the conversion of US defense-unique companies to dual-use production
- Transfer military technology to the commercial sector
- Preserve defense-unique core capabilities (e.g., submarines, armored vehicles, and fighter aircraft)
- Integrate, broaden, and maintain a National Industrial Base sustained by commercial demand but capable of meeting DoD needs
- Adopt the business processes of world-class customers and suppliers (including processes that encourage DoD suppliers to do the same)
- To the maximum extent practicable, stop placing government-unique terms and conditions on its contractors

2.3 Perceptions

The Air Force is frustrated over the lack of success and those attempting COTS products implementation are struggling with its complexity.

Although there are notable exceptions, the Air Force is generally frustrated over the lack of cost savings attributable to COTS products. In some instances, total ownership costs of systems employing COTS have been greater than they would have been using a traditional custom Mil Spec design approach. In addition, there is evidence that a lack of implementation policies, guidelines, standards and processes has led to this lack of success.

Those charged with developing COTS based systems are equally frustrated over its complexity. They are concerned that the customer expects miracles. Just being told to “maximize use of COTS” without guidance, training, infrastructure, processes, tools, metrics, incentives, and leadership won’t make it so.

2.4 Scope of Study

The panel looked at a broad range of applications of COTS hardware and software products involving varying degrees of integration complexity; however, the COTS hardware considered was limited to computers and electronics.

This COTS study covered three broad domains – management information systems (MIS); command, control, communications and intelligence (C3I); and weapon systems. In the past, these system needs were typically met by integrating custom building blocks, components and software designed specifically to meet rigorous military specifications.

The potential extent of COTS technology utilization varies depending on the application. A MIS application can often be satisfied with one substantial commercially available product. A C3I application may consist of multiple commercial products from multiple suppliers with very few custom designs required. On the other hand, a weapon system usually requires a mix of custom and commercial products.

The engineering disciplines considered in this study were primarily electronics, computer systems and software. The commercial market place has taken the technological lead in these areas and offers a wide assortment of products that are suitable for military use. In fact, fielding the most advanced weapon system demands that commercial products from these disciplines be exploited to the maximum extent possible.

2.5 Interviews

Panel members reviewed 34 programs or organizations representing all military services.

Program/Organization	Service	Organization
Advanced Amphibious Assault Vehicle	USMC	General Dynamics Amphibious Systems
AF Operational Test and Evaluation Center	USAF	AFOTEC/CNR
AFPEO/LI for Logistics Info SPO, Gunter AFB	USAF	AFPEO/LI
AFRL COTS Initiatives	USAF	AFRL/MLM & /IFTA
AWACS Computer Modernization[1]	USAF	ESC/AWC
B-2 Data Storage	USAF	ASC/YSA
B-2 EFX 99	USAF	Northrop Grumman
Boldstroke, commonality initiative Open Systems Architecture & Software Component Technology		The Boeing Company
Bradley Fighting Vehicle[1]	USA	United Defense LP
CALCE Electronic Products and Systems Consortium		University of Maryland
DCAC/MRM – Define & Control Airplane Configuration / Manufacturing Resource Mgt		The Boeing Company
COTS Supplier Approaches		DY 4 Systems
Earth Sensor[2]		TRW Space & Technology Division
F-117 & F-119 Engine Electronics	USAF	ASC/LPC & /LPR
F-15E COTS-based Products & F-16 Upgrade	USAF	ASC & ASC/YPV
F-22 Program	USAF	ASC
Global Broadcast System	USAF	Raytheon Systems
GPS, Ground Control Segment	USAF	SMC/CZG
GPS Receiver[2]	USAF	TRW Space & Technology Division
Ground Station[2]		TRW Space & Technology Division
Reuse of COTS Software		GTE Information Systems Division
JASPO, Signal Intelligence Infrastructure	USAF	ASC/RAJ

1 COTSCON 99 Conference Presentation
2 Written Response to Study Questionnaire

Program/Organization	**Service**	**Organization**
Joint Direct Attack Munition[1]	USAF	Program Director
Large ADP Systems & Software Development Process		TRW Federal Enterprise Solutions
Manufacturing Resource Planning[3]	USAF	MRP II Program Office
COTS Implementation in the Mobility SPO	USAF	ASC Commercial Aircraft Program
New Attack Submarine and Acoustic Rapid COTS Insertion Programs	USN	Lockheed Martin Undersea Systems
Enabling E-Commerce & Distributed Computing		Interoperability Clearinghouse
Office of the Department of Defense Chief Information Officer	OSD	ASD/C3I CIO
PVS/EVS – Enterprise Visibility Service		Boeing Information Systems
Deputy Assistant Secretary for Management Policy & Program Integration	USAF	SAF/AQX
T-38C Avionics Upgrade Program	USAF	ASC/EN
T-6A Joint Primary Aircraft Training System	USN USAF	ASC/EN
TacTech (Parts Management)		Transition Analysis of Component Technology, Inc.

[1] COTSCON 99 Conference Presentation

[3] Teleconference

3 Benefits and Risks

The use of COTS hardware and software in USAF systems has many well-recognized benefits, and several not so well recognized risks.

3.1 Potential benefits

The incorporation of COTS products into USAF systems offers anticipated advantages, such as initial price, reliability, availability, and support, which have raised expectations for cost and schedule savings.

When COTS products are incorporated into military or commercial systems, the following benefits are possible:

<u>Lower cost</u>: If a commercial product (hardware or software) in widespread use can be found that provides a function needed in a new system (or system upgrade), then developmental costs (e.g., NRE, coding) which would generally be more expensive than the commercial product, can be avoided. Furthermore, if the commercial product vendor can be relied upon to maintain the product during its lifetime, support costs can potentially be reduced, also.

<u>Faster deployment</u>: Since the commercial product is already available, custom development, which would generally take longer than integrating the commercial product, can be avoided.

<u>Improved quality and reliability</u>: Since the commercial product is in widespread use, defects are likely to have been already identified and eliminated.

<u>Leverage fast-paced commercial product evolution</u>: Competitive market pressures will cause the vendor of the commercial product to periodically offer improved versions of the product. These improved versions are available for incorporation into the system, resulting in potential system improvements.

<u>Reduced development risk</u>: Since the commercial product is market proven, the risk of providing its intended function in the system is mitigated.

<u>Support system in place</u>: Since the vendor is already providing support of the commercial product, the system operators need not create their own support infrastructure.

<u>Upgrades provided</u>: Since improved versions of the product are offered periodically by the vendor, the cost of developing an upgrade to that function of the system can potentially be avoided.

<u>More stable industrial base</u>: Because the supplier is not dependent solely on small volume military business for survival, but rather on a large commercial market, the supplier is more likely to remain in business.

<u>Decreased reliance on sole providers</u>: The large market for a successful commercial product attracts competitors, creating alternate sources.

<u>Improved surge capability</u>: The military's requirement for a COTS product will be such a small portion of the commercial demand that an increased military demand is likely to still be negligible and easily met.

<u>Facilitates innovation from small businesses/academia</u>: Intense competition in the commercial marketplace causes suppliers to actively seek technology that will differentiate their product from the others.

3.2 Risks

The use of COTS products in USAF systems has many consequences that are not well understood by program managers or senior executives.

Although the use of COTS products in USAF systems can have many benefits, such use can also create difficulties in acquisition procedures, product development, and logistics support, as well as in the skills and experience required of Air Force and contractor people. These difficulties create cost drivers that are unique to systems incorporating COTS products. This section will address the issues arising from the use of COTS products and the unique cost factors that result.

3.2.1 Acquisition

Some traditional USAF acquisition procedures are incompatible with the effective and economical incorporation of COTS products into USAF systems.

<u>Inflexible requirements</u>: Traditionally, the operational community sets requirements and the acquisition community comes up with alternative concepts for meeting those requirements. COTS products are designed to meet the desires of the mass market, and it is unlikely that the supplier of a COTS product will offer a modified version for a single, low-volume customer such as the military. Hence, if operational requirements are viewed as not negotiable, and the suppliers are unwilling to modify their COTS products to meet a unique military need, then the probability of finding an exact match between requirement and COTS product is diminished. Often the solution to this dilemma has been for the Government or the Government's prime contractor to purchase the data rights to the supplier's source code (assuming he is even willing to sell the rights) and modify the COTS product. Although this approach can avoid the writing of some code, it voids the warranty on the COTS product and renders it no longer COTS. The advantages of maintenance support and evolutionary upgrades are lost. Government and commercial programs that were successful in incorporating COTS products were able to trade-off requirements with the operational and acquisition communities in order to achieve a best value solution. Rigid requirements or an overly-specific RFP deny the use of COTS products that could offer acceptable performance at lower total cost of ownership. It's better to adapt the requirements to the COTS product than the COTS product to the requirements. If field operators and program managers are unwilling or unable to do this, then they should not use COTS.

<u>Inappropriate application</u>: There are many applications where COTS products are inappropriate. For example, commercial software should not be utilized where absolute trust in the software is essential, such as the control of nuclear weapons. Of particular concern for such applications is an embedded "Trojan Horse" or trap door. Often the functionality of the COTS product is not a good fit with the functional needs of the system. More time may be consumed adapting the product than developing it from scratch.

<u>Lack of Thorough COTS Product Search</u>: There are a tremendous number of COTS hardware and software products in the marketplace that have potential application in Air Force programs. The problem is, how does the program manager conduct the search to find the match? In a not-so-successful program, the COTS product choice was based solely on marketing information gleaned from the internet. Successful programs employed contractors considerable knowledge of relevant COTS products as well as considerable domain knowledge of the intended military application. It is essential that the government choose a contractor with such credentials when COTS products are to be incorporated into a military system.

<u>Inadequate Product Volatility Consideration</u>: The prime contractor must also have a proven methodology for coping with the frequent, asynchronous revisions to COTS products. New versions of a COTS software package may appear as frequently as every 18 months. After three or four upgrades, the software vendor may choose to no longer maintain the earlier version incorporated in the military system.

Indeed, COTS product obsolescence will occur even in the design stage, making it necessary for the initial production item to be different from the prototype. The prime contractor must have a program management process that accommodates this fact of life. This process would include an open system architecture that allows "plug and play" of replacement objects and a spiral development approach that plans for cyclic repetition of design, development and test to create sequential versions of the system devoid of obsolescence. Several of these cycles will be going on simultaneously, albeit in different phases. The contractor has to predict the change cycle for each imbedded COTS product and plan for regular refresh of the system throughout the design, development production and sustainment phases of the program.

Misunderstanding of Commercial Practices: The prime contractor and the government program office must understand the business practices typical of the commercial world in which the COTS vendors operate. Companies that are doing well in the commercial marketplace have no desire to abide by the restrictions and procedures specified in the Federal Acquisition Regulations (FAR) and typically will refuse to do so. For example, the commercial vendor will sell the product based on the price that the market will allow, not on the basis of his cost plus some maximum profit that the government allows. The prime contractor must provide the "impedance match," contracting with the government on FAR terms and subcontracting with the COTS product vendors using commercial business practices.

Inadequate Treatment of Total Ownership Cost: Treating cost as an independent variable (CAIV) in trade studies during system definition has become a common practice in Air Force programs. When COTS products are incorporated into a military system, it is essential that the cost used in the CAIV analysis be total ownership cost (TOC). If CAIV analysis focuses only on initial procurement cost, the impact of several sustainment costs (see paragraph 3.2.5 below) can be overlooked and lead to later misfortune. In addition, all COTS-product trade studies conducted by the prime contractor during the course of the program must carefully evaluate the impact on TOC.

Tools Lacking for Estimating Total Ownership Cost (TOC): Independent cost analyses, "should cost" analyses, etc. are conducted using cost models based on historical data from past programs. Since COTS products are only recently being introduced into Air Force systems, cost factors unique to COTS product insertion and the historical data required to parameterize those factors are absent from current models. The factors that need to be incorporated into the cost models are enumerated in paragraph 3.2.5 below. Incorporation of these COTS-unique cost factors into the cost estimating models used by the government is essential to enable the government to evaluate contractor proposals and to determine the suitability of each COTS product for its intended application.

Lack of Thorough COTS Product Evaluation: The suitability of a particular COTS product must be evaluated not only on the basis of Total Ownership Cost (TOC), but also against a set of performance specifications. These specifications will address the manner in which the COTS product must perform in order to satisfy a particular function in the system being procured. Failure to thoroughly test the COTS product to such a performance specification may result in surprising disappointments later.

Inability to Incrementally Test (OT&E) an Evolving Product: Operational test and evaluation (OT&E) is the actual or simulated employment, by typical users, of a system under realistic operational conditions. OT&E is structured to assess attainment of technical performance parameters, and determine whether systems are operationally effective, suitable, and survivable for intended use. A major defense acquisition program may not proceed beyond low-rate initial production until initial operational test and evaluation (IOT&E) of the program is completed (Title 10 U.S.C., Armed Forces, Subtitle A, Sec. 2399). Successful completion of IOT&E is also the criterion often used to declare Initial Operational Capability (IOC) of an Air Force system. In the past, unique custom designs were static so that the system tested was the system to be manufactured and deployed. Because of the volatility of systems incorporating COTS products, however, the system that is subjected to IOT&E is likely to be different from the system that enters production, since upgrades will have been incorporated. This constant evolution of a system is a cause of concern for the T&E community, since the consequences of the changes introduced into the initial production design are unknown. Hence, OT&E must become an integral part of the continuous spiral evolution of a system throughout its lifetime.

COTS products demand thorough testing particularly in the operational environment. When COTS products are modified, the system may exhibit behavior different from the baseline requiring additional testing.

Safety-related concerns are normally identified during OT&E through attendance at various system safety forums. Certain circumstances may not allow adequate safety review prior to OT&E. For these cases, initial system safety reviews should be conducted before actual testing begins.

3.2.2 Product

Systems employing COTS products require a radically different approach to design, integration and test.

<u>Trend towards integrated versus federated:</u> System cost, size and weight considerations generally favor a higher level of integration. However, COTS products are more easily adapted to a federated architecture where functional partitions are more clearly defined. Highly integrated architectures do not necessarily preclude the use of COTS products, but the task is more difficult.

<u>Evolving industry standards:</u> An open system architecture supported by commercial industry standards is an essential design consideration for a COTS-based system. These standards evolve and change in response to commercial market needs. Therefore, it is important to keep track of where things are headed. Road maps should be generated that attempt to predict the future course including jumping off points from one technology to another.

<u>Claims and performance do not always match:</u> COTS product claims are not necessarily a good indication of actual performance and capability. Buying decisions should be based on a thorough evaluation of both hardware and software products.

<u>Interoperability:</u> Integrating multiple COTS products within one system can lead to many interoperability problems. COTS software is a particular challenge. Extensive integration testing is essential.

<u>Performance feature clash:</u> COTS software typically includes more features and functionality than are normally needed. Precautions must be taken during system development and subsequent upgrades to assure that these unused features do not clash with other software products.

<u>Frequent upgrades</u>: COTS software upgrades are offered about every 18 months. Suppliers typically don't support versions that are more than two or three revisions old. The system design and field upgrade process must assure that upgrades can be accommodated with relative ease.

<u>Questionable trust:</u> Most COTS software is developed outside this country. Of particular concern is the possibility that a trap door or "Trojan Horse" may be embedded in the code.

<u>Lack of support for niche features:</u> Occasionally a COTS product is selected for a particular niche feature. If it turns out that the commercial market is not interested in this capability, there could be a lack of support or subsequent revisions may exclude the feature entirely.

<u>Environmental performance:</u> Most COTS hardware products and components do not meet the military environmental specifications. Specifications must either be relaxed or special testing performed to assure that the product is suitable for the intended use.

Plastic encapsulated microcircuits (PEMs): There is a raging debate ongoing as to the advisability of using PEMs in military hardware. The concern is primarily the plastic encapsulation. Historically, comparable mil-spec parts have employed ceramic packages. PEMs can be qualified for military applications by special testing and evaluation including Highly Accelerated Life Tests (HALT) and Highly Accelerated Stress Screening (HASS). Extreme care must be exercised when upscreening or upgrading commercial parts for military applications. Knowledgeable parts experts need to be involved.

3.2.3 Logistics and Support

Sustaining USAF systems that include COTS products introduces new challenges, such as technology refresh and insertion.

Short commercial product life cycle: Obsolete electronic parts have been a major problem for existing military systems. The problem is exacerbated with the use of commercial parts since their product life cycle is generally much shorter. It is essential that a parts management program be put in place for component selection, obsolescence management, life cycle projection, inventory availability and design sustainment analysis.

Multiple configurations: Configuration management of a COTS-based systems is considerably more difficult due to a combination of short commercial product life cycles, planned technology refresh, and the spiral development process. Everything is in a constant state of flux.

Software upgrades: In order to maintain support, upgrades must be incorporated periodically.

Technology refresh: One of the major advantages of utilizing COTS products is the ability to modernize more frequently. For example, computers can be changed out periodically to provide more computational capability. Cost effective technology refresh requires an open system architecture that facilitates the integration of new and improved products.

Frequent procedure updates and training: The continuum of software upgrades, technology refresh opportunities, obsolete parts problems, etc. requires that procedures be updated more frequently and additional training be provided.

3.2.4 People

The lack of skills, knowledge and abilities in the use of COTS products by government and industry personnel at all levels has limited the effective utilization of COTS products in Air Force programs.

The existing culture in both government and industry does not enable successful implementation of the COTS philosophy in Air Force programs. Most personnel in each area have spent their careers dealing with the acquisition and application of hardware and software that is specifically designed, procured and built to military specifications. The skills and knowledge that they have acquired, although related, does not provide adequate understanding to address the additional complications in selecting, specifying, buying and using commercial products for military applications. Different abilities are required to develop a commercial performance specification than are required to impose and enforce an existing military specification. As a result, many people in both organizations recognize their shortcomings in applying COTS products in their respective jobs.

Government personnel involved in the development and maintenance of military specifications and the qualification and testing of military hardware feel that their role is threatened with the introduction of COTS products. Engineers, designers and production personnel within industry feel that work, which they routinely perform, will now be acquired from commercial suppliers. This has resulted in feelings of loss of job security and control within the leadership and the rank and file in government and industry.

While the senior leadership in both government and industry has proclaimed their support for the COTS initiative, this support has not been demonstrated by a well defined policy and understanding of COTS issues. Most people affected by this initiative at the program level in government and industry have not received sincere encouragement or adequate training to implement the goals. Since the implementation of COTS activities will require strong support and dramatically altered roles for these key players, the initiative is incurring resistance. All of these factors have limited the effective utilization and implementation of COTS products in Air Force programs.

3.2.5 Unique Cost Factors

There are several unique cost factors to consider when utilizing COTS components. For brevity sake, these factors are summarized in Table 2 and discussed below.

3.2.5.1 Assessment *Factors*

One of the challenges one faces when moving to COTS is figuring out which hardware/software to select. Things aren't always what they seem. Specific vendor literature may be deceptive as it may mix what is with what will be. Performance always seems to become an issue especially when one uses general-purpose products for specific applications. This is especially worry some when only one product is selected (and worse yet "extended") to meet a critical need. Furthermore, vendor stability and customer support become issues because resulting dependencies are established on others to sustain and maintain components that one may only have marginal influence over. Thus it is imperative that when using COTS, one considers and mitigates this dependency risk by employing "widely accepted standards" and avoiding "closing" the system through the use of proprietary/unique extensions, APIs or tools. Unless one is very familiar with the commercial marketplace, products, standards and vendor base, entire selection process can be very risk-filled.

To make sure that one gets the needed features and functions, one must conduct an assessment. Such assessments can be extensive and expensive unless it is already a normal process of the established "way of business." The best approach would be to have a broad knowledge base of various and alternative components before buying it. Having specific performance benchmarks that are measured against actual applications to include an assessment of vendor support on optional product provide significant visibility in the selection process. However, this approach costs money, takes time and consumes talent.

Finally, one must contract for the components. In this case, being a smart consumer helps a lot. When possible, one should negotiate terms and conditions for the initial order from a position of strength. Seek favorable quantity discounts for the future and push for maintenance and support agreements that "you can live with." Always keep other options open with alternative vendors.

3.2.5.2 Tailoring Factors

The next task is to tailor the generic products to do specific things. This may or may not be as easy as it seems. To tailor, one traditionally adjusts pre-defined parameters or rules within a set context. Problems often arise when the context or boundaries need to be expanded. For example, the Application Programming Interface (API) handles a variety of protocols, but not the protocol of interest. Again, this may take additional effort and time. Care must be taken not to violate the integrity or modify the code of the COTS application, and COTS extensions should be avoided.

Table 2. Unique Cost Factors Attributable to COTS

Unique Cost Factors	**Hardware Components**	**Software Components**
Assessment Factors	▪ Hardware understanding ▪ Performance benchmarking ▪ Component selection ▪ Contracting	▪ Software understanding ▪ Performance benchmarking ▪ Package selection ▪ Contracting
Tailoring Factors	▪ Hardware adaptation	▪ Package adaptation
Integration Factors	▪ Hardware staging, integration and checkout of COTS components ▪ Hardware/software integration and test using COTS components ▪ Vendor liaison and technical support	▪ API understanding ▪ Glue code development ▪ COTS software integration & test ▪ Vendor liaison and technical support
Production Factors	▪ Hardware production, packaging, distribution and quality control ▪ Vendor liaison for bug fixes	▪ Software production (from controlled libraries) ▪ Vendor liaison and support
Update/Maintenance Factors	▪ Synchronization of hardware updates/replacements ▪ Preventive maintenance ▪ Hardware obsolescence ▪ Vendor liaison and technical support	▪ Synchronization of package updates with release cycle ▪ Glue code maintenance ▪ Package obsolescence ▪ Vendor liaison & technical support

3.2.5.3 Integration Factors

Integrating COTS hardware and software components into the system and getting the vendor to fix problems can be a nightmare. Typically, the pressure is on to get a system to the field. Getting components to work together and performing to the user's satisfaction can take considerable time and effort. For example, one might have to develop glue software that was wrapped to support an API. One might be tempted to tailor COTS packages with new rules to compensate for hardware problems. Workarounds and functional tradeoffs must be part of the COTS solution. All sorts of situations arise that result in adding time and effort on this task, especially if one is not experienced in applying COTS

3.2.5.4 Production Factors

When initiating production, one must package the deliveries so that the "untouched" COTS components comply with vendor or government restrictions. For example, one may have to replace a 128-bit encryption algorithm with a 64-bit version because of export restrictions. This may require considerable vendor liaison so as not to violate the COTS software licensing agreements. In addition, multiple configurations of several versions of the products may have to be supported. Transition planning and multiple supported variants are part of doing business with COTS.

3.2.5.5 Update/Maintenance Factors

Finally, the ultimate challenge is keeping everything synchronized as vendor components are updated at different times and according to different maintenance cycles. As new and more powerful COTS hardware components are added, additional testing and liaison will be needed. As COTS software undergoes changes to fix bugs and add features, new releases of the system will have to be developed. Glue code will have to be changed and older versions that are no longer supported by the vendor may have to be maintained. However, "technology refresh" planning and migration are part of the solution and should minimize end of life buys and maintenance of dated systems or code.

3.3 Fact Finding Results

Interviews of USAF program personnel and contractors revealed a correlation between certain attributes of programs that successfully incorporated COTS products and the lessons learned from programs that were not so successful.

3.3.1 Exemplary Programs

Five of the 24 programs reviewed were considered to have been highly successful in incorporating COTS products.

> *"[Take] full advantage of the technologies and management lessons that have turned around American commerce and industry in the last decade."*
> Jacques Gansler,
> Under Secretary of Defense for Acquisition and Technology, Dec 1997

The panel offers the following five examples to foster a more complete understanding of the COTS program success factors by presenting them as they were observed in practice. These were selected to represent the best program attributes by both government and industry officials. While no specific success/failure statistics should be drawn from the relative numbers of total programs observed against those selected, the overall consensus of the panel was that very few programs could be categorized as "exemplary." Each program is presented with a description, background and key points section.

The following programs were rated "exemplary." First, was a program combination where the Navy effectively brought COTS and commonality efficiencies to its New Attack Submarine (NSSN) and Acoustic Rapid COTS Insertion (ARCI) programs. These two programs, grouped as one architectural implementation, were the most significant and extensive commercial application observed, as it dealt with commercial technology in both a large scale development program (NSSN) and an upgrade of legacy platforms (ARCI). The second program described in this section used commercially available components to reduce cost of an expendable weapon, the Joint Service's Joint Direct Attack Munition (JDAM). The panel found that the Air Force's Airborne Warning and Control System (AWACS) computer modernization program had practices that "opened" its architecture to commercial systems and cost savings. The Marines' Advanced Amphibious Assault Vehicle (AAAV) program had strong management and an exceptionally well integrated product team approach that exemplified the requirements-affordability-availability trade-off process required for leveraging available commercially products. Lastly, the Department of Defense's Manufacturing Resource Planning (MRP) II was a successful information system deployment, where the customer adjusted processes and procedures to match the available commercial system.

3.3.1.1 New Attack Submarine (NSSN) and Acoustic Rapid COTS Insertion (ARCI) Programs

Program description

The NSSN is the next generation attack submarine with a program start in 1996 and an Initial Operational Capability of 2006. The ARCI program "back-fits" the legacy submarine platforms with common non-propulsion systems (both within the current fleet and common with the NSSN). Lockheed Martin Undersea Systems company is the prime contractor. Both the Navy Program Office and the Lockheed Martin Undersea Systems company had established experience in deploying commercial technology prior to contract award.

Background

The Navy and the submarine industry commonly recognized that platform cost escalation combined with the budget reduction environment would "doom" submarines as unaffordable in the Post Cold War era and littoral warfare mission environment. This "life threatening" need drove innovation. Commercial technology solutions appeared to offer both cost and performance solutions. However, the Navy recognized that both the Navy program management and the supporting industry base required education in using commercial off the shelf (COTS) technology, if a new process was to succeed. Thus, the Navy established a study contract to accomplish this purpose.

Education. To help the Navy program office understand the commercial community and the potential of commercial technology the Navy prohibited all contractors that had ever done business on submarines from bidding on a commercial communications study contract. After award the existing prime submarine contractors were invited in to comment on the commercial solutions made by the companies. This two-year dialogue raised the experience base of both the Navy and the supporting contractor base to a level that a major proposal for a new submarine could be let with commercial technology objectives explicitly defined.

The resulting savings from the commercial approach have cut development costs by 75% and fixed price production costs by 80% over the then year prices of legacy systems – a dramatic result. Furthermore, the system doubled the processing capability over the existing systems, provided a life-cycle process to manage technology obsolescence, developed a sustainment warranty based on Total Ownership Costs, and defined common tools and processes by which both the Navy and the prime contractor could manage the commercial technology integration.

Cross-program Commonality. Because of the success of the NSSN commercial technology acquisition, the Navy decided to apply appropriate elements of the NSSN system to their existing submarine fleet. Objectives of this “back fit” program (ARCI) were 1) to reduce costs through fleet commonality – acquisition as well as supportability and training and 2) to reduce risk of the “forward fit” into the NSSN. By coupling the NSSN and ARCI programs, including the operation and support program elements, the Navy managed multiple funding and contracting issues through a cooperative integrated process team (IPT) in partnership with the prime contractor. This innovation in Acquisition Reform was recognized by the Vice President with *Hammer Awards* in both 1998 and 1999.

“It’s different.” Extensive commercial technology use required a fundamental change from “unit flyaway cost” to one that was founded on total ownership costs and a continual technology “refresh” process. This major shift required changes in every aspect of the acquisition and required different business arrangements, processes, tools and metrics. Risk management shifted to a balanced sharing with technology prediction as a weighted element of design decisions. Different processes and tools had to be acquired and validated.

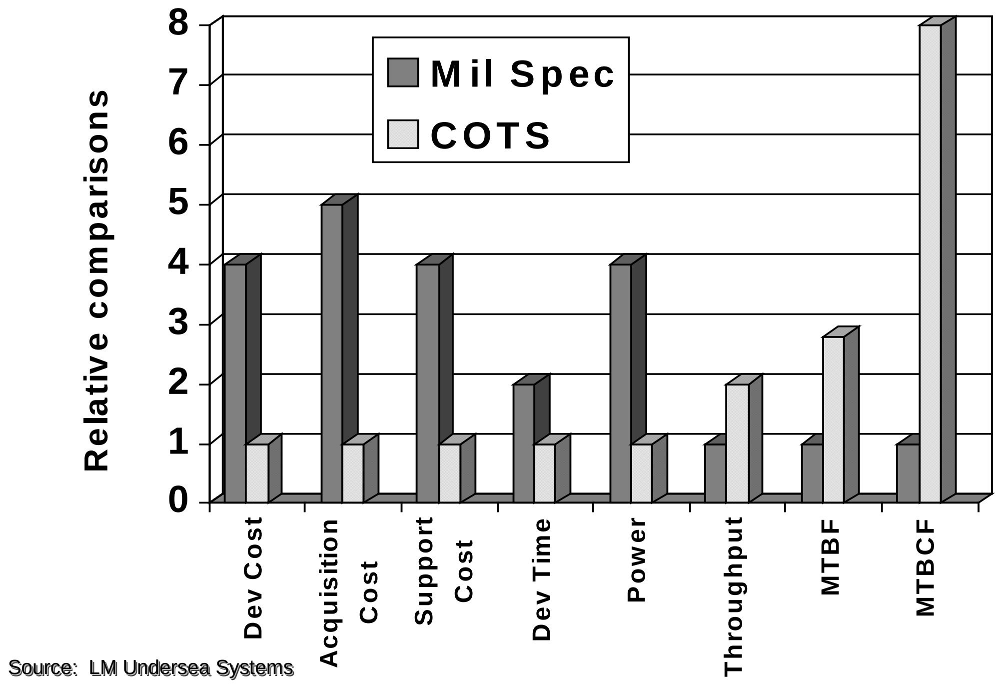

Figure 2. COTS Benefits of Acoustic Rapid COTS Insertion over Legacy Mil Spec system

Requirements, technology and affordability trade-offs. Requirements were not "flowed down" then matched to technology possibilities, in the traditional fashion. Instead, requirements were first examined and solutions partitioned to the available or planned commercial technologies. Furthermore, requirements that resulted in high costs were examined with alternative technology and procedural trade-offs. This process resulted in major design efficiencies on the NSSN such as shock isolating the submarine deck as opposed to the individual cabinets.

Process. Design decisions weighted affordability in an architecture that covered the entire system lifecycle, not just a product delivery where subsequent contract changes would be worked as independent and unconnected events. Thus, supportability became a major design element and the required sustainment engineering required for that activity was calculated from the start. A strong and comprehensive vendor base along with business rules (such as maximum percentage of particular vendor's business) were established and updated four times a year. This vendor tracking process produced a technology forecast, twice a year. In turn, this forecast was integrated with the Navy Program Office's technology modernization planning. Major new capabilities were prioritized and inserted synchronously with refresh events while incremental improvements from Navy Labs, universities and small businesses were more frequently accomplished. Throughout the program, production decisions were influenced by 1) technology refresh needs to manage obsolescence, 2) supportability engineering and 3) update and new capability insertion.

Tools. Several management tools were developed to support the "buy and integrate" process. Experience gained over the history of COTS use populated the data that drove the decision process. Two of the more important tools are described below.

> ***Technology Selection.*** To track value and to compare the numerous variables associated with technology and standards selection, a large database was constructed from previous lessons learned. These were the "questions that you wished you had asked, and those that you were glad you asked." After the questions were weighted by project need, the contractor used a commercially available

decision tool – Expert Choice™ – to accomplish a pair-wise comparison of the various factors in making each product choice.

Spares management. Spare parts were modeled across multiple platforms and configurations as technology was refreshed and legacy parts replenished the spares inventory. No level-sparing model, used by the Defense Department at the time was capable of modeling the new factors adequately. However, by validating a model that had traditionally supported the commercial oil and gas industry with the Navy, a new process was instituted that resulted in 40% less cost, not including other intangible savings in depot "shelf space," infrastructure and logistics personnel.

Life-cycle Support. To effectively employ commercial technology, a program must undergo a continual engineering element, maintain licenses and deliberately plan system updates. Or the program would soon have software and hardware components that would no longer be obtainable at competitive prices, supported by commercial tools or covered by vender warranties. However, this additional price when managed by experienced managers was very small in comparison to the savings attributed to leveraging commercial market investments. Warrantees and operational readiness guarantees were also included in the COTS based system. Although these were funded in one-year increments due to financial constraints, the intent of the program was to continue to let consecutive contracts based on satisfactory system performance, while retaining a minimal documentation package in case of failure. Coupling the maintenance and support package with the COTS design and production phases made the business deal attractive to industry, as well as incentivized it to cost sensitivities in the operations and support phase. With the shift away from production costs to lifecycle efficiency, COTS use drove the business model from "design and build" to "buy and integrate."

Standards. Standards selection, rigorous implementation processes and transition planning made system transportability across the 40-year system life possible. Avoiding the pitfalls of vendors that extend their products and provide proprietary shortcuts that lock-in customers to specific solutions, the program held firm to widely used commercial standards. By avoiding proprietary extensions and sticking to widely accepted, mainstream commercial standards, the program minimized perturbations of the ever-evolving commercial technology base and more easily accommodated new winners that immerged as mainstream products. Furthermore, technology forecasting not only supported product evolution but also supported standards evolution. By staying within the commercial marketplace state of the practice versus state of the art, the program positioned itself to use the commercial bridges that facilitate the general marketplace transitions. The program also realized that successful COTS implementation required more system engineering discipline and management focus as compared to a mil spec development.

Innovation Management. A strong system engineering environment (SEE) facilitated rapid incremental insertion of added functionality. In the case of the ARCI program, algorithm updates and improvements from the research base – labs, universities and small business – have been incorporated into the fleet in 12 to 18 months, as opposed to the previous five-year average. In this process the prime contractor passed the development interfaces, standards and implementation protocols to the research activities. Compliance, while not mandatory, facilitated fielding where services such as diagnostics and communications were already in place.

Suitability. COTS products were not suitable for every case on NSSN or ARCI. Government software was used for sensor and weapon launch interfaces. Nuclear propulsion was also controlled by government developed systems. However, these pieces comprised only 25% of the total weapon system's computational and communications activities as opposed to 90% on previous generation platforms.

Key Points

Leadership

- Both military and industry leadership recognition that business as usual practices would result in failure. Life threatening need, which was commonly recognized by both Navy and contracting industry made the risk of taking new approaches to the problem less of a risk than proceeding on the established course.
- Both Navy program management and contractors had experience with COTS. Through a series of incremental steps they had developed processes and tools to deal with commercial technology.
- The Navy leadership implemented an educational process, before implementing change. A study contract with non-traditional, commercial businesses trained the acquisition agency and conditioned the marketplace.
- Navy leadership extended its successful concepts across platforms to produce system-wide commonality and the resultant more efficient deployment of COTS
- Both contractor and Navy recognized need for different skills, organizations, models, design philosophy, and SEE required for the successful COTS deployment. The contractor realized that effective COTS implementation was not possible in a legacy organization based on the "design and produce" model and was committed to making the change to "buy and integrate."
- Program management made all requirements negotiable and adjusted them through a rigorous process that evolved end users and maintainers. Each need was subjected to a fit to commercial technology and affordability.
- Management synchronized the COTS implementation process with 1) technology refresh needs that managed obsolescence, 2) supportability factors and 3) new capability insertions.
- Program management developed and used new models and tools. These were especially valuable in making technology selection, performing supportability engineering and costing COTS options.
- Management based lifecycle support options on capability guarantees and funded them in yearly increments.
- Management oversaw an effective standards selection and implementation process, as it formed an essential element of the SEE. The standards promoted interoperability and transportability throughout the system life of the platform.
- Innovation management not only synchronized performance upgrades with technology refresh, but also used the strong SEE to rapidly integrate research-based software improvements.
- The program found that COTS could successfully meet many of the military "computational infrastructure" needs such as general purpose, signal processing, communications, and data handling.

Engineering

- Experience with COTS. Both Navy and contractor had processes and tools in place to deal with commercial technology.
- Cross-platform and system-wide commonality resulted in more efficient deployment of COTS.
- The requirements accommodated the available and projected commercial technology, all within a dramatically reduced cost structure.
- Commercial technology was the preferred solution. And only after rigorous evaluation was a custom solution implemented. The process was designed to leverage the large, non-recurring commercial investment. Software development, hosting and partitioning was designed leverage future capability updates.

- The program leveraged the commercial support infrastructure by establishing and executing a technology refreshment strategy through an architecturally driven process.
- Technology selection, supportability engineering and cost modeling were assisted with innovative management tools.
- Technology assessment was an essential part of a sustaining lab environment that included integration and testing.
- Standards selection and implementation procedures form essential elements of the SEE.

3.3.1.2 Joint Direct Attack Munitions (JDAM)

Program description

Joint Direct Attack Munitions (JDAM), with the Air Force as the lead service and produced by the Boeing Company, is a low cost guidance kit which converts existing unguided free-fall bombs into accurately guided "smart" weapons. By adding a new tail section containing an Inertial Navigation System (INS)/Global Positioning System (GPS) guidance to existing inventories of Mk-83 and BLU-110 1,000 pound (450 kg) bombs, and the Mk-84 and BLU-109 2,000 pound (900 kg) bombs, the cost effective JDAM provides highly accurate weapon delivery in any "flyable" weather. JDAM can be dropped up to 15 miles from a target with updates from GPS satellites to help guide the bomb to the target.

Background

This program came out of Desert Storm where our war fighters said they needed an affordable weapon that can attack accurately in weather. The Air Force and Navy plan to buy 87,496 low-cost JDAMs for approximately $18,000 per unit. Original cost projection for JDAM was $40,000 per unit, but through innovative management and acquisition reform measures DoD saved more than $2 billion. With the addition of foreign military sales, the JDAM contract is expected to be worth $4 billion to the Boeing Company. As a pioneer program in DoD for acquisition reform, JDAM has been carefully watched by program managers in all three services. The JDAM SPO looked at how successful commercial business operations were conducted and used a lot of commercial practices and many commercial components. All of those resulted in significant cost reductions in the development and production of the weapon. Currently, MK-83 1,000-pound and MK-84 2,000 pound blast and fragmentation bombs are being modified to become JDAMs. Hard Target penetrators being changed into low-cost JDAMs included the 2,000 pound BLU-109 and 1,000 pound BLU-110. Test launches have already been made from the Air Force B-1, B-2, B-52 and F-16, and the Navy's F/A-18 Plans are underway to also make JDAM compatible with the Air Force F-15, F-117 and F-22; the Marine A/V-8B; Navy F-14 and the Joint Strike Fighter (JSF). In a test last year over White Sands Test Range, N.M., a B-2 launched 16 JDAM on a single pass. The weapons attacked four targets on two widely separated complexes with devastating results. The unprecedented emphasis on cost drove the need for COTS.

Key Points

Leadership (Contractor and Government)

- Set clear cost goals and thresholds at the outset
- Setup a rigorous cost tracking and reporting process
- Ensured that all parties take responsibility for cost reduction
- Involved contractor before contract to help identify few critical performance requirements and perform trade studies to refine/validate cost goals.
- Streamlined data requirements

- Established integrated product teams
- Created environment of open, honest communications
- Established supplier partnerships
- Committed to production pricing during EMD phase of contract
- Negotiated a commercial-like 20 year warranty
- Utilized award fee to drive cost goals, i.e. employees incentive compensation funded by a percent of award fee

Engineering

- Concentrated on front end of program to optimize design for low cost
- Performed trade-offs on non-critical requirements via a requirements review board, i.e. reduced mission profile temperature
- Simplified product build-up eliminating special tools
- Simplified aircraft load eliminating special equipment
- Treated manufacturing as an integral part of the design team
- Utilized industrial grade parts including plastic encapsulated microcircuits (PEM)
- Performed Design for Manufacturing and Assembly (DFM/A) to eliminate parts and simplify design
- Performed factory simulation
- Analyzed and verified key manufacturing process capabilities
- Used industry best practices to maximum extent possible
- Periodically re-qualified to ensure product meets requirements

3.3.1.3 Airborne Warning and Control System (AWACS) Computer Modernization

The panel found that the Air Force's Airborne Warning and Control System (AWACS) modernization program had practices that opened its architecture to the use of commercial systems and cost savings.

Program Description

The AWACS program is an Air Force program whose prime contractor is the Boeing Company. The purpose of the modernization program is to incrementally update the mission computing architecture to provide the aircraft with improved offensive counter air capabilities. The first increment of the modernization program provides the warfighter with improved track continuity/maneuver response and eliminates bottlenecks caused by antiquated computer equipment.

The computer modernization is an upgrade from a custom Mil Spec central computer with unique interfaces to a scalable client/server architecture utilizing available commercial products to provide improved tracking, windows oriented full color displays and high resolution maps. The system is designed to ensure real-time capability employing COTS hardware and a COTS real-time POSIX operating system (Lynx).

Background

The AWACS role has evolved since the 1970's. New missions have required new mission support applications. The existing mission computing architecture is a major limitation. Tight fiscal and schedule constraints required a shorter development cycle and contained development, integration and maintenance costs. The modernization program emphasized the migration from the legacy system, not a wholesale replacement, and a design that would ease future upgrades (technology refresh).

The E-3 AWACS provides a mobile, survivable surveillance and C2 platform. In service since 1977, AWACS can separate airborne targets from ground and sea clutter using its sophisticated "look-down" radar. Its radar "eye" has a 360-degree view of the horizon, and at operating altitudes can track both air and sea targets simultaneously for a distance of 200 miles.

The aircraft is currently undergoing a multi-stage improvement program as avionics is moved from the 707 aircraft platform to a 767. In order to keep the costs of this avionics upgrade affordable, Boeing has adopted open mission computer architectural interfaces. These interfaces were selected to allow the architecture to grow and evolve.

The AWACS modernization strategy is incremental. During stage 1, new and improved COTS workstations, enhanced display capabilities, better track/maneuver response performance, and partial DII Common Operating Environment (COE) compliance will be provided. During stage 2, full migration to a DII COE compliant architecture is planned.

Implement a Fully Networked Client-Server Architecture that provides a Low Cost Growth Potential. The current legacy mission computing environment is a "closed" system that uses custom, expensive Mil Spec hardware to provide survivable capabilities. This hardware will be replaced with standard, low-cost commercial components (processors, buses, power supplies, cabinets, etc.) that provide the system with a plug and play upgrade capability that can be used if and when system performance becomes a problem (e.g., historically this occurs every five years as new mission requirements are added).

Improve the Display Capabilities. New displays that use COTS components to the maximum degree possible will be implemented as part of the hardware upgrade. These displays will be modern full color capable Windows-oriented peripherals that allow the operator to view the battlespace in 3D and color.

Unravel the Software Architecture. The legacy software will be rearchitected to ride on top of a middleware platform that will minimize the application software's dependencies on the hardware platform. To achieve these goals, a standards-based COTS POSIX operating system will be employed as the interface between the hardware and applications software. The middleware that acts as the bridge between the applications and the operating system will use a CORBA-compliant (Common Object Request Broker Architecture) interface to schedule communications in real-time between functions. Scheduling will be encapsulated within the communications requests and will be scheduled to completion using the popular rate monotonic scheduling algorithms developed by the Software Engineering Institute (SEI).

Reduce Risk to the Program by Putting a COTS Process Framework into Practice. Because of the numerous risks inherent in the use of COTS hardware and software, the program put a process framework into place that enabled it to cope with the downsides of the technology. For example, the framework emphasized careful selection of both COTS products and COTS vendors. It focused on cooling methods for hardware because these are critical to product availability (and performance). It implemented a "try before you buy" philosophy for software so adequate benchmark data on performance would be available for making decisions. It educated and trained all members of the team in the pluses and minuses of COTS. Team members included the customer and executives.

Key Points

Use of COTS is viewed as an enabler by the modernization program. AWACS plans to exploit both COTS hardware and software to keep costs down during engineering and maintenance. The key points of the program that are aligned with this goal are:

Leadership

- Early use of integrated product teams (IPTs)
- Rigorous cost tradeoffs relative to the environment and operational capabilities (Table 3)
- Contractor in support loop to handle DMS, technology refresh and software upgrade issues

Table 3. Environmental requirements substantially reduced to accommodate COTS products

Specification	Original Requirement	COTS Requirement
Operating temperature	-54 C to +55 C	0 C to +50 C
Non-operating temperature	-62 C to +85 C	-40 C to +71 C
Altitude 2.5 hours operating	Test of design changes	One time test
Humidity operating	100% with condensation	85%
Humidity non-operating	100% with condensation	95%
Rain non-operating	24 inches per hour	No requirement
Salt	Salt-sea atmosphere	No requirement
Ice, blowing snow, etc.	Various requirements	No requirement
Shock	15 +/- 2 g peak, 11 msec	6 +/- 1 g, 11 msec
Vibration	Based on Mil-Stds	Measured plus factor
Survivability	HEMP plus others	HEMP only

Engineering

- Open system architecture
- Spiral development
- Design for future upgrades or technology refresh
- Drive commonality across systems
- Modeling of COTS subsystems
- Avoid being first user/implementer
- Careful selection of COTS vendors
- Thorough COTS market research
- Plan adequate testing
- Shock mounted hardware

3.3.1.4 Advanced Amphibious Assault Vehicle (AAAV)

Program description

The Advanced Amphibious Assault Vehicle (AAAV) program has a Direct Reporting Program Manager (DRPM) to the Assistant Secretary of the Navy Research, Development & Acquisition (RDA). General Dynamics Land Systems has been prime contractor on the AAAV since award in June of 1996.

The majority of the engineering and design, along with prototype integration and assembly efforts are performed at a common government-contractor site. Located in Woodbridge, Virginia, the site was chosen for its proximity to the U.S. Marine Corps Combat Development Command and houses approximately 250 people from General Dynamics, the U.S. Marine Corps, and subcontractor companies.

Background

The Advanced Amphibious Assault Vehicle's role is to enable Marines of the surface assault echelon to quickly and securely hit the beach and sustain momentum ashore, fighting on to their “objective maneuver.” The AAAV is envisioned by USMC leaders as the third and newest element in an "amphibious triad" that will greatly enhance the speed, range, maneuverability, firepower and survivability of forces moving from ship to shore, and then into the enemy interior. The other two elements are the Landing Craft, Air Cushioned (LCAC) and the V-22 Osprey aircraft. All three programs are large technology and capability leaps beyond the systems they are replacing.

Highly integrated team. The strongest element of the AAAV procurement has been the highly integrated military-contractor integrated product team approach. Built on a foundation of trust and close working relationships, the program makes daily decisions and tradeoffs across the program in a manner that was impossible under “arms length,” formalized structure – saving time, increasing communication and reducing cost. The AAAV’s teamwork approach has evolved into an interactive, healthy, positive partnership. All of the services now use "integrated product teams," on which industry representatives sit side by side with government officials, but the AAAV program has developed the model to a “best practice” level, with the program office actually co-located with its prime contractor's factory.

Key Points

Leadership

- The AAAV program had strong program management and an exceptional integrated product team approach has exemplified the requirements-affordability-availability trade-off process required for advantaging available commercially available products.

Engineering

- Willingness to work trade-offs across the entire program against the available technology.
- Formal engineering processes established to accommodate the trade-off process. Process initiated from top to bottom and lateral at every decision element. Responsibility and authority for making trades pushed to the lowest level possible in the organization with top-level interface responsibilities.

3.3.1.5 Manufacturing Resource Planning (MRP II)

Program description

The typical business of the maintenance depots is to take a DoD asset, such as a helicopter or a tank, strip it down to its carcass, and then rebuild the asset. The parts used to rebuild the asset may be those that came with the asset if they need no repair. They could also be newly manufactured parts or repaired parts. The MRP II program is part of an overall philosophy to maximize the commonality of business processes across all depots for all of the services. The MRP II program automates a significant piece of the depot's business systems. In part, the MRP II program is the result of an earlier unsuccessful effort—the Depot Maintenance Management Information System (DMMIS), a program led by the Air Force that attempted to bring together the best components from each depot's suite of business systems and create a single depot management system. The DMMIS program, which was based on different custom software and modified COTS components, was unsuccessful, and the Joint Logistics Systems Command was tasked to provide a new system, part of which resulted in the MRP II program.

Background

Understanding the process. While the exact approach taken by MRP II is not intended as a formula for every program, it provides evidence that DoD programs can find success with COTS by adapting government processes to the supported processes of a commercial system. Part of this program's success can be found in the new way that the government approached the job of procuring, fielding, and sustaining the COTS-based system. As such, the system chosen embodied the MRP II (Manufacturing Resource Planning) philosophy, an extension of an earlier Material Requirements Planning (MRP) process. Thus, the MRP II philosophical foundation was established in the domain of commercial manufacturing practices, with established standards and professional groups. The MRP II program office selected the Compass Contract product from Western Data Systems (WDS) as its core application. In turn, WDS integrated COTS products and custom components into their product. As the development progressed, the development team determined which business processes could be realistically changed or adjusted to adapt to the accepted commercial practices.

Requirements and stakeholders. In MRP II the initial requirements were defined in a traditional manner. These were used in surveying the marketplace for available solution candidates. With a significant gap between available COTS products and government business practices, rather than attempt to modify any COTS products, the scope of the program was narrowed to provide automation for a useful subset. Stakeholders across the services were involved from early requirements definition, through the requirements "trades," and into source selection. After award, the program office built an effective partnership with its vendor. No government-specific feature requests were allowed. Participation in vendor and government user groups was promoted, as was participation in relevant professional groups.

Key Points

Leadership

- MRP II program leadership had the insight of a previously unsuccessful attempt (lessons learned) and furthermore, used an experienced COTS provider.
- MRP II leadership had both users and contractors involved at every step.
- Where "requirements" mismatched with COTS capability, the requirement/and or military process was modified to fit the available capability.
- MRP II program management and the contractor involved understood the commercial marketplace drivers from the start of the procurement and accommodated them at every step.

Engineering

- MRP II based their system on commercial practices and used the COTS code "untouched."
- MRP II was based on widely accepted commercial standards and practices.

3.3.2 Attributes of Successful Programs

The five exemplary programs have 12 attributes that contributed significantly to their success.

1) All operational requirements and procurement specifications should be negotiable

   Requirements were modified or in some cases deleted all together to accommodate an existing COTS product. For example, The AWACS program eliminated salt spray requirement to facilitate the use of COTS computers. Depots modified traditional processes to utilize an existing MRP II software package. The NSSN submarine structure was modified to provide environmental isolation. Generally, maximum utilization of COTS products requires a high degree of requirements flexibility. If requirements are fixed, then fewer COTS products can be utilized. The key is reasonable give and take.

2) Open system architecture and the spiral development process are essential

   Each exemplary program used Open Systems as an element of its system engineering environment to not only manage obsolescence and frequent changes driven by the marketplace, but also to synchronize capability updates with the inserted technologies. This was done in a continuously evolving spiral development and system support environment. Systems were kept open by adopting strict adherence to a "minimal set of widely accepted commercial standards," and planning for the eventuality of replacement and/or upgrade of both the hardware and software components. The evolutionary updates of technology insertion were wedded to functionality updates. To manage the Spiral Development a sustaining engineering element continuously evaluated the commercial technologies. The upgrade process was managed by a combined government-contractor team that included the end users and representatives that were charged with the system's support.

3) Incentivize the prime contractor to provide a credible estimate of support costs

   Support costs were incentivized by engaging the prime contractor in the support and sustainment phase of the system lifecycle. Use of fixed price guarantees and warranties with cost savings co-sharing opened the procurements to reducing total ownership costs by examining trades made in the upfront design and production phases. Although some of the systems had to deal with the "one year at a time" contracting and balance mandated government depot use, it was felt that multi-year, unrestricted contracting could potentially reduce costs even more.

4) Use Total Ownership Cost (TOC) as a source selection cost criterion

   Reducing costs over the entire lifecycle of a weapon system was an evaluated criterion of each acquisition. Although this activity lacked validated costing models and a consistent process across DoD, each program evaluated the criteria. One technique used to capture the costs in the sustainment phase was to have the contractor bid warranties with their products. The programs required explicit detail regarding the replanning and reengineering that would be ongoing throughout the life of the system to be balanced against potential savings.

5) Assess the contractors past experience employing COTS products

   Experience is a critical success factor. When both government and contractor were experienced in applying commercial products, the success rate was high and cost savings were dramatic. When both were inexperienced the success rate was very low and costly, many times resulting in substantial overruns and even program failure.

6) Evaluate contractor's COTS processes for assessing, selecting, integrating, supporting and refreshing

Using commercial technology effectively requires a different process for evaluating second tier companies and products, as the government no longer controls the development path of the underpinning vendor base. The traditional method of "down selecting" to a single vendor may lock a development along a proprietary path that could be very costly to break once taken. Commercial systems take their lead from the commercial marketplace and may not develop along lines expected in the marketing brochures. Successful programs had processes, and in one case a laboratory supported by models and business arrangements, to accomplish this assessment. Each program evaluated the contractors on this capability as an essential element of COTS competency.

7) Use TOC to determine suitability of COTS versus custom products

Total ownership costs considerations were required when assessing a COTS implementation. The program acquisition must deal with change and expense elements throughout a product lifecycle that may not be evident in a custom product. The successful programs assessed items such as product deviations caused by commercial product evolution cycles, extended military product support and licensing against the expected benefits of more affordable sparing, shorter development times and increased performance. Production costs as an exclusive selection criterion when using COTS was not considered a prudent choice.

8) Assess contractors understanding of the commercial marketplace and relevant COTS products

Factors such as a vendor's financial stability, strategic direction, and volatility of the technology on which the commercial item was based, or the frequency of commercial item releases must be assessed. Contractors that successfully used COTS in their designs understood this and had processes in place to manage it. COTS suppliers will generally not tolerate traditional military procurement relationships where the contractor may exercise substantial control.

9) Ensure the system application matches the COTS product functionality

Ideally, the COTS product functionality should closely match the intended use. However, this is not always possible. If the gap is too great then more effort will be expended developing adequate interfaces than developing the product from scratch.

10) Verify that the contractor proposes to use the COTS product without modification

If one line of code is changed or if one circuit board modification is made, it is no longer COTS. COTS must be used "unmodified" to retain the value of a commercial product. Otherwise, voided warranties, lack of support, and upgrade difficulties will result. Basically, the advantages of COTS are lost. In fact, the result may be less cost effective than a custom design. Realistically, there may be some unusual cases where modification is appropriate, but the government should take extraordinary measurers to assure that it is well justified and results in the lowest TOC.

11) Conduct trade-off analyses of all changes versus Total Ownership Cost

Trade-offs among the operational requirements, system context, the architecture and design, and the marketplace are essential to a successful COTS deployment. The exemplary programs made these trades against the total ownership costs of the system's life cycle. Each program documented trade-offs made with stakeholders. Unfortunately, evaluating commercial items in order to identify TOC trade-offs is an unfamiliar process for many program managers (and their users). It is equally unfamiliar for many contractors who are more comfortable with simply meeting a specified set of requirements. One program facilitated the TOC evaluation process by integrating a technology assessment test bed with cost models. This allowed end users to contribute to decisions regarding tradeoffs between commercial item function, cost, and other factors.

12) Enforce ongoing interaction between government personnel (ops and acquisition) and prime contractor in Integrated Product Teams

A strong Integrated Product Team (IPT) approach was evident in every exemplary program observed. Each program implemented business practices based on a team concept. As a result, management improvements included developing process that sought to maximize the use of appropriate commercial items. This process identified both strengths and shortfalls in the capabilities provided by commercial items. Then in turn, the program developed strategies to mitigate the shortfalls of commercial items, while at the same time identified opportunities for improved business practices within the DoD organization. In another successful program, the IPT was used as a mechanism for identifying and making trade-offs among system context, architecture and design, and the capabilities of commercial items. Requirements were collected and prioritized, and costs were estimated based on the commercial availability of the required capability. This information was used to rework program priorities. A third successful program developed an unusual incentive strategy that rewarded individual engineers for identifying commercial items that could be used in the system.

(This Page Intentionally Blank)

4 Product

People use processes to create products. In this and the following two sections we will address these three elements - - product, process, and people - - for end-item products (i.e., USAF systems) that include commercial-off-the-shelf (COTS) hardware and software as essential elements.

4.1 Commercial trends

When applying COTS to USAF systems, it is essential to be aware of, and plan for, changes that are likely to occur in COTS hardware and software products.

4.1.1 Software

Among the software trends over the next five years, application software will be built from common building blocks and vendor conformance to middleware interface standards will grow.

COTS-based systems fall into three categories:

1) COTS solution systems: systems where one COTS product provides all the required capabilities,

2) COTS aggregate systems: systems where a collection of COTS products is integrated together to provide all required capabilities.

3) COTS component systems: systems where a collection of COTS products are integrated with custom and perhaps legacy systems to provide all the required capabilities.

The complexity of the development and sustainment effort grows with the number of COTS components the make up the system. In order to control the complexity, system developers must leverage certain architectural styles, integration mechanisms, design methodologies, and SEE capabilities. The following sections describe these trends and how they impact COTS-based system development.

4.1.1.1 Software Components

Components are the basic building blocks of COTS-based systems.

Trends

- Niche components.
- Established vendors merging adding to volatility of vendor base.
- Established vendors expanding product capabilities means less to integrate.
- DCOM and EJB absorbing CORBA.
- Some vendors make more money supporting COTS not selling it.
- Current COTS products may contain a lot of dead code since infrastructure has advanced.
- DSP COTS software is emerging.
- Security issues not addressed.

4.1.1.2 Software Methodology

Object-oriented approaches to COTS-based system design are becoming predominant in the marketplace.

Trends

- UML is the design notation de jour.
- XML may provide portability of data and support data integration.

4.1.1.3 Software Architecture

Most COTS aggregate systems have a layered Client/Server/Broker architecture style.

Trends

- Web-centric architectures with Browser front ends dominant.
- Emerging "push" protocols using software agents.
- Run-time support for dynamic architecture update is immerging.

4.1.1.4 System Engineering Environment

Tools lag necessary support of COTS-based system development.

Trends

- Modeling software is lacking – there is no good way to predict performance of COTS systems.
- Test/Regression testing is needed but not adequately being addressed by marketplace.
- Configuration management tools are improving, but sometimes ad hoc OODBMS are used.
- Scripting languages (e.g., Tck/Tk, Perl, JavaScript, Visual Basic) are being used as "glue code".
- Some "Megaprogramming" foundation work is being done (e.g., Weidorhold at Stanford)

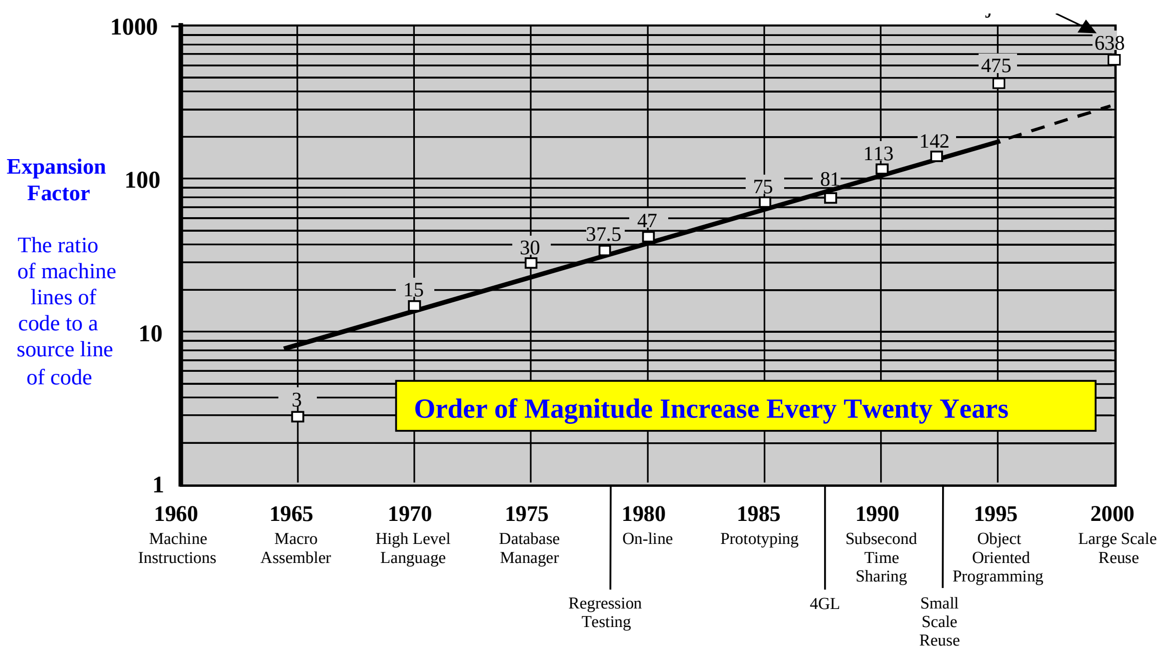

Figure 3. Trends in Software Expansion (Bernstein, 1997)

4.1.2 Hardware

Current trends in semiconductors (components), computing, and communications continue to the year 2012.

4.1.2.1 Hardware Components

<u>Semiconductors</u>

Semiconductor process advances underpin all advances in computers and in commercial, military, and consumer electronics. The semiconductor industry is unique because capability (semiconductor capacity) increases at the same time price decreases. The historic trend for the improvement in integrated circuits has been a doubling of performance every 18 months. This trend, which has been more or less consistent since Gordon Moore first observed it in 1964, is expected (according to the 1997 *National Technology Roadmap for Semiconductors*) to continue through at least 2012. In the semiconductor industry, this trend is known as Moore's Law. Moore's Law is not a law, but rather a business proposition.

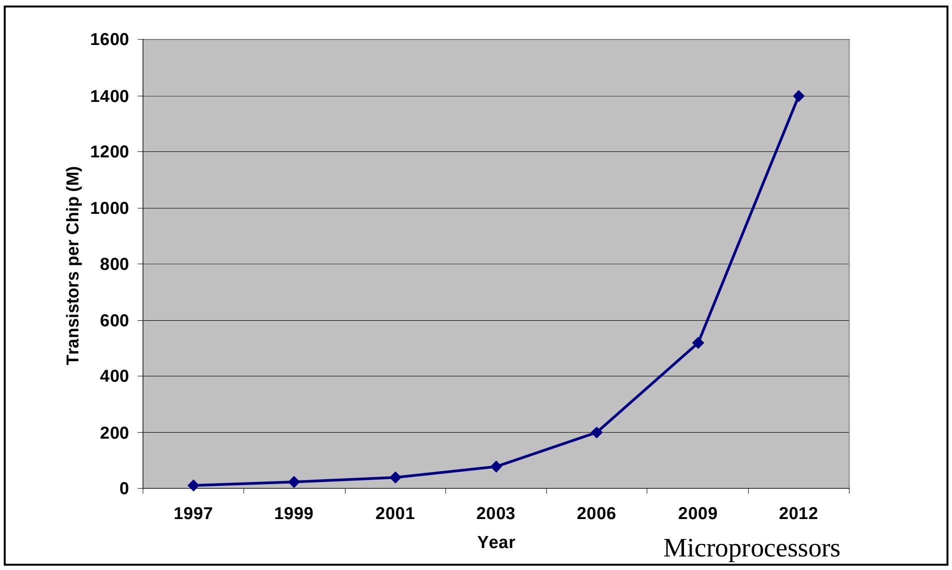

Figure 4. Semiconductor Capacity Forecast

Figure 4 plots the projected number of transistors on a microprocessor. The number of transistors on a microprocessor relates roughly to the capability to perform logic functions. This, then, is a measure of the rate at which computational capability will increase between the present and 2012.

The combination of increasing clock speed, increasing number of transistors on a microprocessor, and improvements in computer design cause the doubling of computer performance every 18 months. This trend should continue to 2012. Doubling every 18 months means we will get ten times the capability every five years. Five years from now, our computers will have ten times the computational capability they have today. In ten years, computers will have a hundred times the computational capability they have today. In fifteen years, computers will have a thousand times the computational capability they have today. Look at the curve above: the capability we have fielded over the history of the computer is <u>nothing</u> relative to what we will field in the next ten years!

Figure 5 shows the projected cost per transistor for a microprocessor. The interesting observation from this curve is the decrease in cost over time to implement a function. Suppose we need the capability that can be provided by a million-transistor microprocessor. According to the chart, that million-transistor microprocessor would cost us about $30 in 1997. The million-transistor microprocessor costs $10 in 2001 and only a few dollars by the year 2010.

Figure 5 plots how cost to build an integrated circuit declines over time. But cost in the semiconductor business is only weakly related to price. When a new high-end component is introduced, the manufacturer generally prices the component for very high margins. Leading-edge microprocessors (such as the latest Pentium, for example) and programmable logic devices (PLDs) may cost $30 to $65 to manufacture, test, and package, but manufacturers typically charge about $1000 for these components. From the time of introduction, however, the price of a component historically declines at 25 to 30% per quarter. A component introduced at $1000 would be priced at $100 two years later. At the end of two more years, the component would sell for $10.

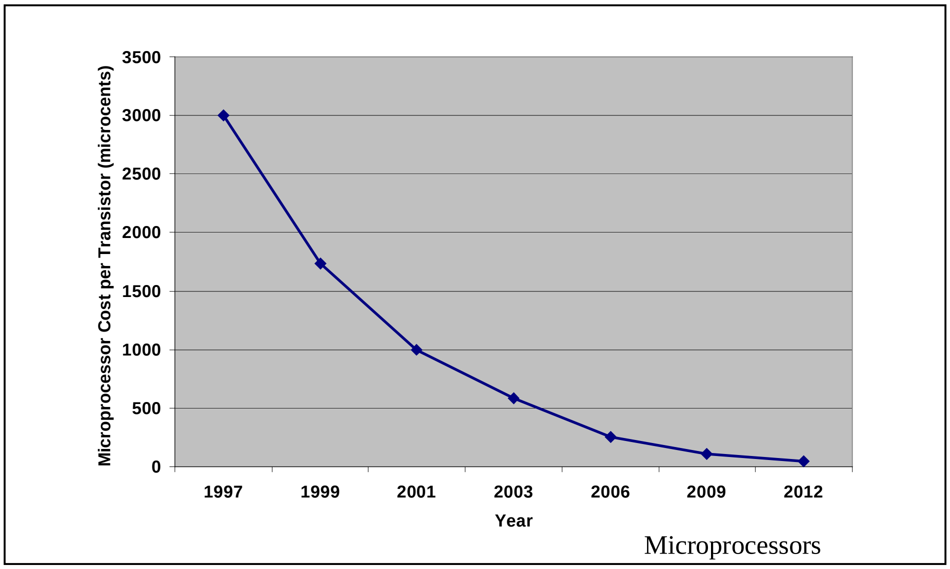

Figure 5. Semiconductor Cost Forecast

In 1999, worldwide sales of personal computers will be about 120 million units. In contrast, the 1999 sales of embedded microprocessors will be about 5 billion units. Personal computers use high-performance, high-margin parts, so they generate a lot of revenue and they get a lot of press attention. For all the attention they get, personal computers represent less than three percent of worldwide unit sales for microprocessors. Each year, microprocessor manufacturers ship about one microprocessor for every living person on the planet! Each of these microprocessors is embedded in an application and requires software. The average electric toothbrush, for example, contains about 3000 lines of code. An automatic transmission contains about 50,000 lines of code. A cell phone contains about 300,000 lines of code. Over time, the electronic content and the software content of the things around us is increasing rapidly.

The semiconductor industry is unique because it continues to improve as prices decrease. Figure 6 shows line width which is projected to continue to decrease at approximately its historic rate until 2012 (the limit of the projection). These trends were taken from the online edition of *The National Technology Roadmap for Semiconductors*. This document can be found at http://notes.sematech.org/97melec.htm. Figure 6 shows projections for six important trends: line width, DRAM bits/chip, microprocessor bits/chip, on-chip clock speed, DRAM cost/bit, and microprocessor cost/bit. Values are normalized to their maximum value over the period so the trends are clearly visible. Capabilities increase and costs decrease over the projected period.

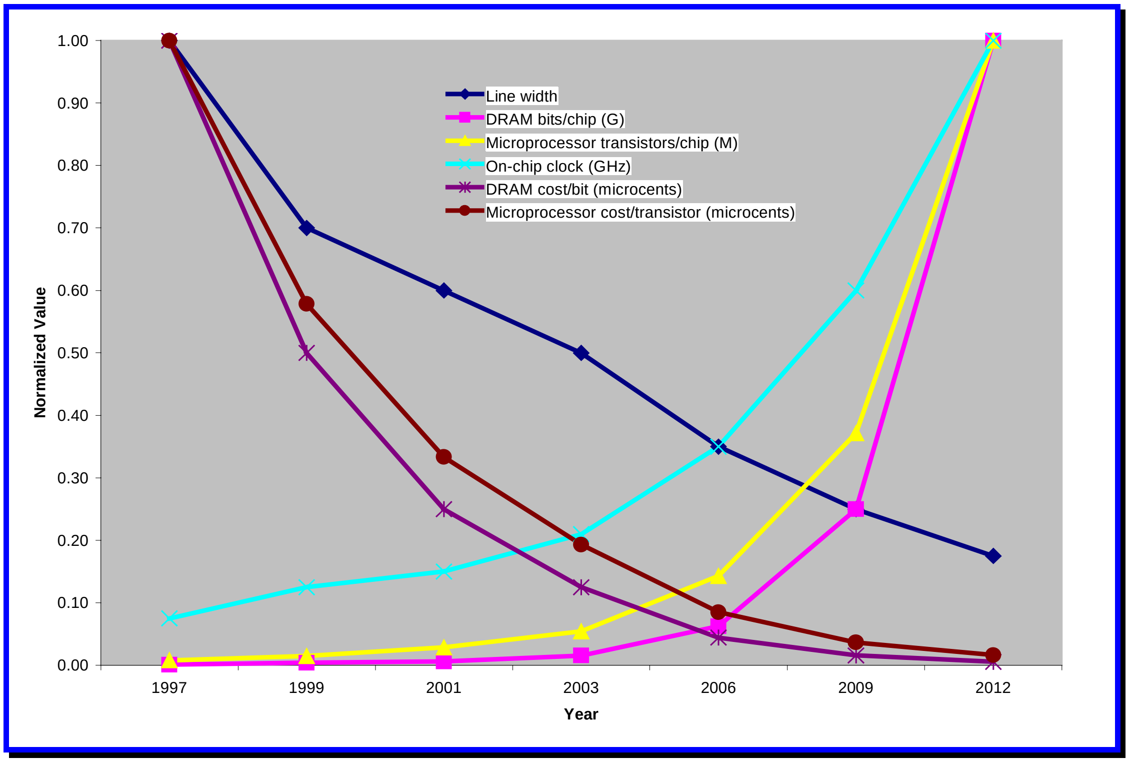

Figure 6. Projected Normalized Semiconductor Metrics

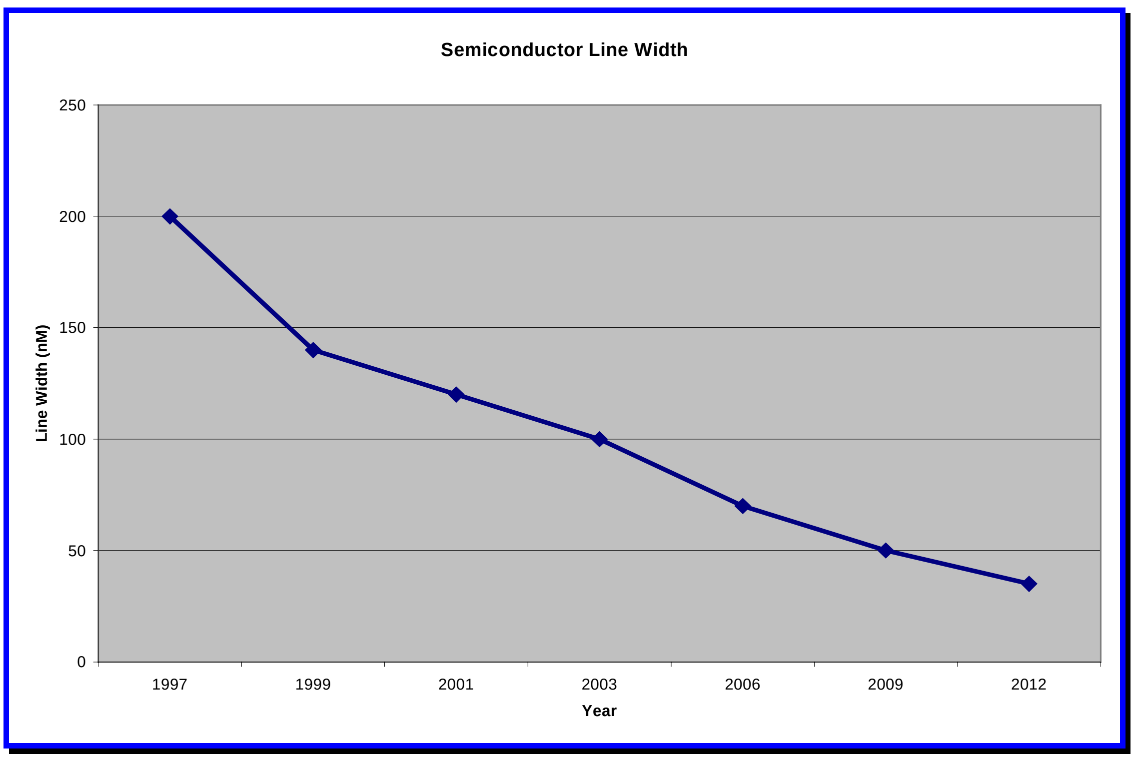

Figure 7. Semiconductor Line Width

Line width (Figure 7) is a measure of the scale at which the circuits are drawn on the chip. The narrower the line width, the more transistors will fit in a unit area. In fact, as the line width decreases, the number of transistors in a unit area increases with the square of the decrease in line width (since the decrease applies in both the x and y directions to build a two-dimensional transistor). The chart above projects semiconductor line width from 1997 to 2012. There are technical barriers to achieving this projection. The confidence among experts is very high that the industry will reach these projections. What is not clear is the exact path we will take to get there. The path will change as we make choices along the way. In general, near term problems will be solved to achieve near-term objectives. The number of options available to solve near-term problems is relatively small. More options are being considered long term. Solutions to near-term problems will reduce the number of options to work on for the longer-term problems.

Figure 8 plots projected clock speed to the year 2012. The on-chip clock speed is another indicator of increasing processing capability.

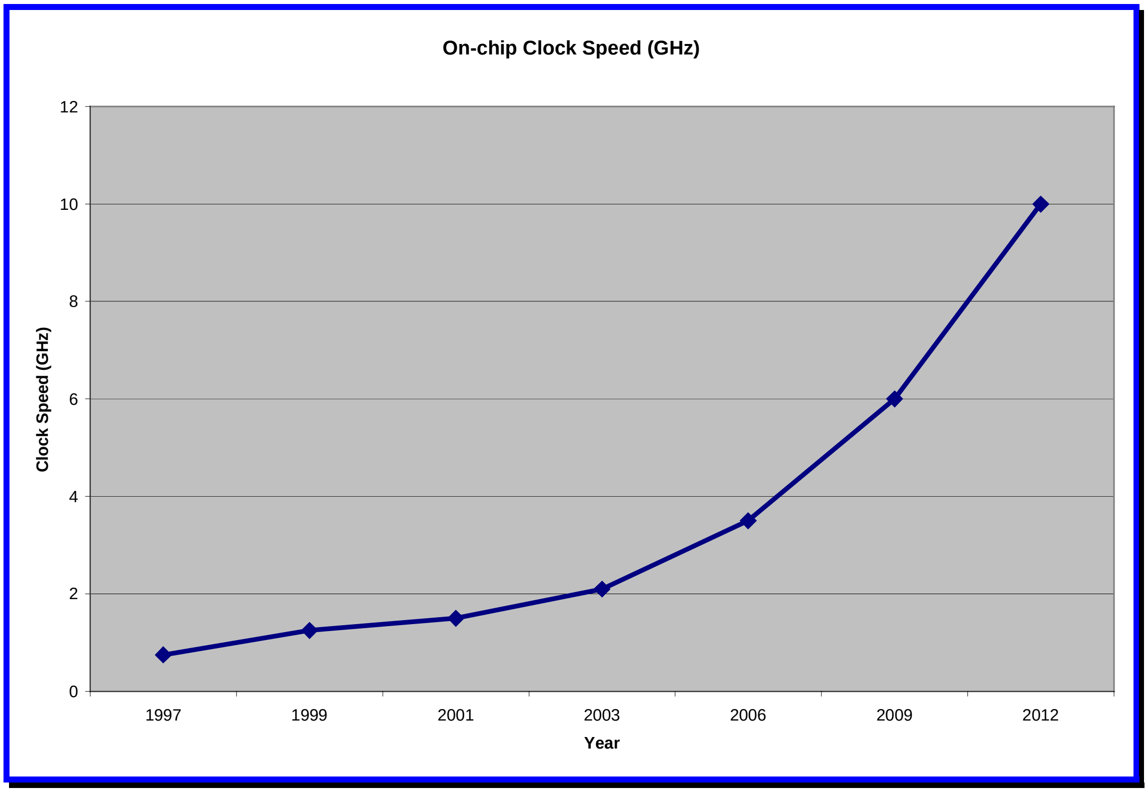

Figure 8. Projected Semiconductor Clock Speeds

Figure 9 projects transistors per chip for semiconductor DRAM (Dynamic Random Access Memory). Maximum single-chip memory size is another indicator of trends in available computational capability. These charts plot the size of the largest microprocessor and the largest DRAM over the period of the projection, but we have said nothing about the cost. The cost of the high-end part should remain relatively constant over the period from 1997 to 2012.

It is the combination of increasing clock speed, the increasing number of transistors on a microprocessor, and improvements in computer design that contribute to the doubling of computer performance every 18 months. This trend should continue to 2012.

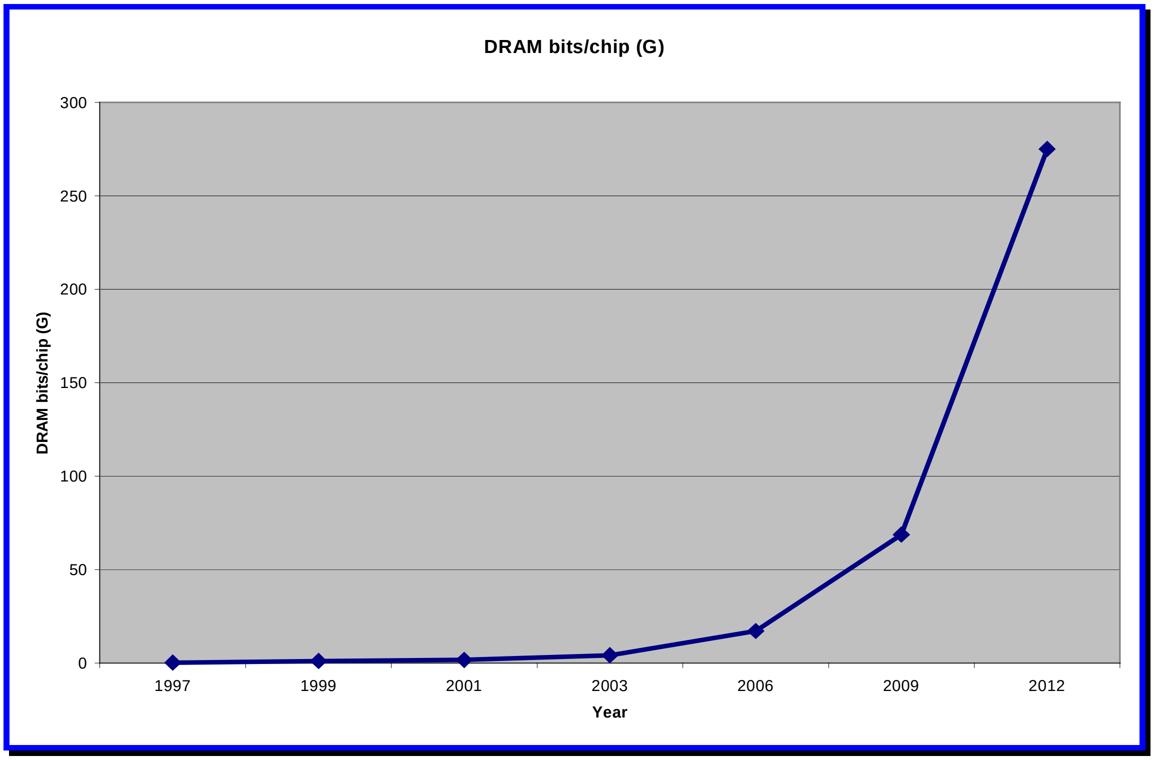

Figure 9. Transistors Per DRAM Chip

Figure 10 is a similar curve showing the projected trend in cost per transistor for DRAM. This curve is even steeper than the microprocessor curve. It appears to be dropping at about half every two years.

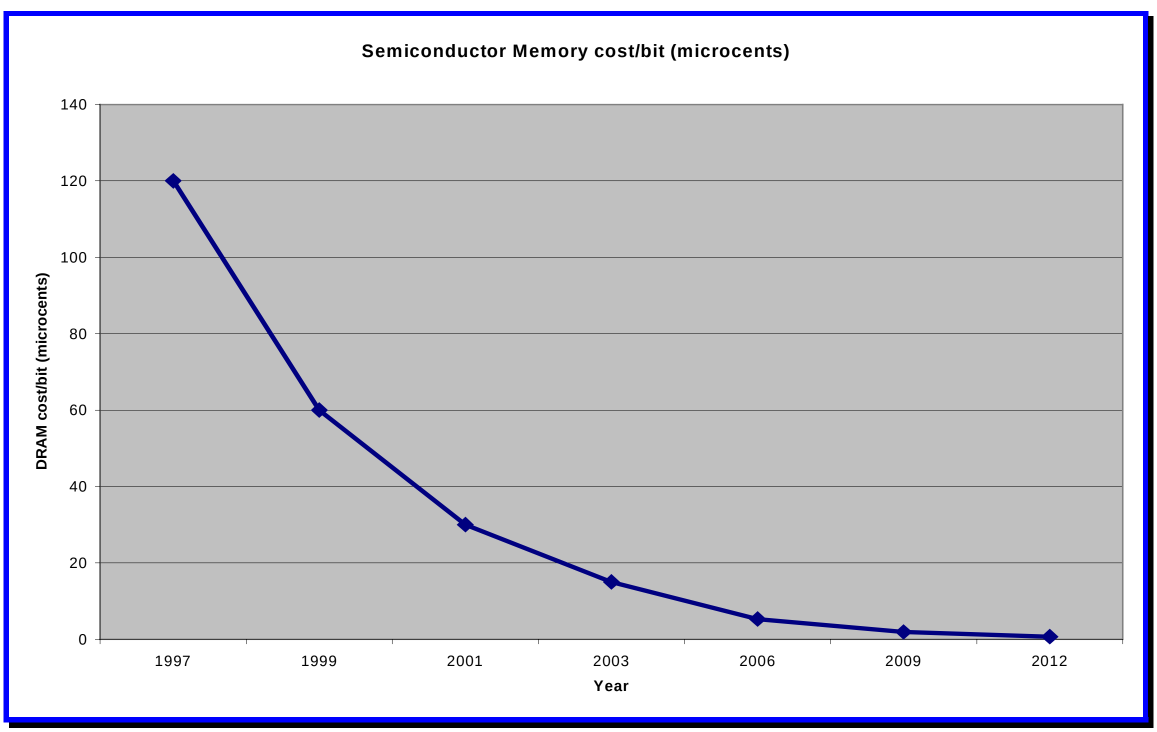

Figure 10. Cost Per Transistor

<u>MEMS and Sensors</u>

Below, we project general trends in component areas related to the integrated circuit. Advances in semiconductor processing underpin what is happening in these areas.

Since the invention of the integrated circuit, this combined trend of increasing capability and decreasing cost has been a unique hallmark of the semiconductor electronics industry. That is about to change with the introduction of MEMS. MEMS combine mechanical and electrical systems on a single chip. MEMS can take advantage of the same trends that have been driving the semiconductor industry because they use the same equipment. Progress in MEMS should be even faster than progress in semiconductors because MEMS, which don't require leading-edge process, can use already-amortized semiconductor processing equipment.

MEMS combine miniature mechanical components and electronic subsystems on a single substrate by exploiting semiconductor-processing techniques. The commercial market for MEMS, which was about $3 billion in 1997, is expected to grow to $14 billion by 2000.

The best current examples of commercial MEMS are the accelerometer used in automobile airbag sensors and the nozzles in ink jet printers. Further applications will soon include pressure, chemical (both for chemicals in fluids or airborne), and motion sensors, fluid pumps and valves, optical add/drop multiplexers, and mirror arrays. MEMS-based filter circuits can achieve a quality factor (Q) of 80,000. This compares with factors about 3000 for filters built of discrete components. In the not too distant future, as evidenced by already-existing prototypes, turbine engines occupying less than a quarter of a cubic inch and producing above 60 watts will be practical.

Because MEMS use semiconductor processes, hundreds or thousands of devices can be fabricated on a single wafer. Integrating the mechanical and electrical functions improves reliability, precision, and sensitivity and reduces susceptibility to vibration and electrical and mechanical noise. Batch fabrication in large numbers reduces cost.

The commercial market for video equipment (digital cameras and recorders) will drive improvements in integrated vision sensors. Games, personal computers, and cellular phone markets will drive improvements in audio and video sensors. Toy markets will drive improvements in integrated motion.

4.1.2.2 Hardware Computing

The x86's success in the microprocessor-based computer market drove most microprocessor manufacturers to seek embedded applications for their processors. The focus of development for microprocessors is shifting to embedded applications. These applications include personal digital assistants and cellular telephones. Performance improvement for these devices will follow Moore's Law, doubling every 18 months until 2012 (the end of the technology forecast in the *National Technology Roadmap for Semiconductors*). With the invention of the microprocessor, computing began to move from the computer to all electronic devices. This trend is accelerating now that most microprocessor manufacturers have been forced out of the market for computer applications. Consumer electronics will drive the industry, particularly digital cameras, which will drive sensor development and cellular communications, which drives computational power efficiency (considering the product of computational capability and power dissipation as a measure of efficiency).

<u>Computer Trends</u>

Since the commercial introduction of the microprocessor by Intel in 1971, the microprocessor industry has grown rapidly. Hobby computers based on microprocessors were on the market by 1974, but IBM's introduction of the Personal Computer in 1981 created the market for microprocessor-based computers. The market for microprocessors grew rapidly. Computers based on microprocessors overtook minicomputer implementations. Workstation companies based computers on proprietary microprocessor designs, but the Intel-based personal computer's volume grew faster than the workstation markets. With microprocessor development cost growing faster than the market, it soon became too expensive for most manufacturers to design custom microprocessors. In 1999, Intel-compatible (specifically, x86 which is also known as IA-32) microprocessors dominate the computer market. Intel-compatible microprocessors will continue to dominate computer applications for the next five years.

Increasing demand for Internet access and the consumer market for personal computers drive improvements in computing. These trends should continue at least for several years. Some time in 2000, Intel will introduce a new microprocessor architecture with a new instruction set. Intel calls the architecture IA-64 and the first processor will be Merced. This will not cause major disruption in the market because the IA-32 instruction set (x86) is thoroughly entrenched. Intel and others will continue to supply IA-32 products to service the giant and growing installed base. Intel's IA-64 microprocessor will have a x86 compatibility mode, so even the new microprocessors will execute legacy code.

Computers will continue to improve, but in the next few years the focus of development for microprocessors will be embedded applications for the consumer market.

Network Trends

Networks are proliferating. Networking companies are probably growing faster than the computer makers. Several factors drive these trends:

- Increasing popularity of the Internet and World Wide Web.
- Conversion of analog media (voice and video) to digital. This includes digital broadcasting.
- Conversion of the telephone networks from circuit switching to packet switching (which implies digital multiplexing on shared access lines).
- Convergence of voice, video, and data over common cable. These were separate industries with separate physical plants; they are converging to a single IP-based network.

4.1.2.3 Communications

Standards & Protocols Trends

ATM and Internet Protocol (IP) were battling for dominance in the network. IP is winning at the low end. In the network backbones, ATM and IP are mixed, with the telephone companies backing ATM. ATM has an advantage in being (theoretically, at least) able to guarantee quality of service (QoS) which is necessary for delay-sensitive communication, such as video and voice. ATM, however, has only a 48-byte payload in a 53-byte packet (a 10% overhead on the network), has never been able to achieve the low cost for network interface cards that is necessary for proliferation, and has been in a continuous state of being defined. This has helped hold back the proliferation of ATM to the desktop. Meanwhile, IP is adding features and improvements. In only a few years, for example, Ethernet has improved from 10 Mb/sec to 100 Mb/sec and now to 1 Gb/sec. Network Interface Cards for 100 Mb Ethernet are below $20, which ensures continued dominance of Ethernet to the desktop. In addition, switched Ethernet improves bandwidth availability for larger groups.

For wireline and fiber networks, IP and Ethernet will dominate the networks up to the backbone trunks. The backbone will carry a mix of Ethernet and ATM, with Ethernet dominating.

For wireless networks, the situation is not clear. In the US, the legacy analog cellular system (AMPS) still dominates the installed base. There is a wide range of new digital standards competing to displace the analog system. TDMA, CDMA, and GSM are three mutually incompatible digital protocols. All of these protocols are being deployed in the US. It is unclear which, if any will dominate the market.

Communications Trends

Roaming is a major driver for the development of multi-band, multi-mode cellular telephones. Subscribers would like to be able to use the same phone anywhere in the country, so the incentive for the service providers is to have a cellular phone that will allow the user to connect to any digital network or even the analog network. It is difficult to do this in today's cellular designs because the protocols are implemented in application specific integrated circuits (ASICs), which means there is a separate set of chips for each protocol. Implementation of a multi-protocol cellular phone would be too bulky and it would consume too much power. The payoff to the providers is huge, however, so there is incentive to improve the cellular telephone. These improvements would spread to other consumer devices.

Despite the marketing difficulties Iridium has experienced, satellite communications is expected to increase rapidly in the next ten years. Middle earth orbit (MEO) constellations will provide global broadband access. Constellations will be a mix of MEO and geosynchronous (GEO) satellites. Low earth orbit (LEO) constellations may not be a viable solution.

The principal advantages of a LEO constellation are high bandwidth (there are lots of satellites) and low latency (the satellites are close to earth). However, there are significant disadvantages. LEO constellations require many satellites, which makes launch and maintenance expensive. Also, to achieve global coverage, the satellites must regularly traverse the South Atlantic Anomaly in the Van Allen Radiation Belt. This subjects them to intense radiation and shortens lifetime of the electronics.

The latency advantage of LEO constellations is due to the network protocols that interpret delay as congestion and slow down. In the future, it is likely that protocols will be smarter and interpret delay as delay and congestion as congestion, overcoming much of the disadvantage of being in a higher orbit.

4.2 Applicability Guidelines

Although there are no irrefutable rules for the use of COTS products in USAF systems, some guidelines give an indication of where success is more likely.

Based on the assessment of many on-going programs, some simple applicability guidelines are provided, but by no means are these guidelines complete. First, commercial market segments that have applicability in military systems and are leading technologically are clear candidates for COTS-based systems. Most notable examples are computers, communications, electronics, and electronic components. To field the most advanced system requires that commercial products from these industries be exploited. For example, a Pentium class microprocessor costs 250 to $400M to develop and development costs are increasing about 40% per year. The Air Force can't afford this type of development expense and must take advantage of their commercial availability.

Most weapon and C3I systems employing computers can use COTS infrastructure products including hardware, device drivers, operating systems and middleware. Generally, the application software is custom, but not always.

Management Information Systems (MIS) are clear candidates for COTS products. In fact one should be very cautious of a recommendation that custom software be developed. Most MIS needs can be met with available software. As stated before, avoid any customization of MIS software; there number of successful examples is limited.

A wide variety of commercial electronic products are COTS candidates such as GPS receivers. Organizations should be knowledgeable of the marketplace, particularly in relevant market segments, and constantly looking for applicable products.

5 Process

Successful COTS-based systems necessitate process tailoring to address COTS-related issues.

This section provides the reader with anecdotal evidence regarding the impact of COTS on the development of COTS-based systems. This information is then used to identify either new processes or changes to existing processes necessary to support COTS.

5.1 Acquisition of COTS-Based Systems

During the acquisition phase, the total ownership cost of the system must be addressed and tradeoffs firmly evaluated.

5.1.1 Anecdotal Findings

Myth #1: You buy a COTS product to fit your application.

Reality: You buy the right to use a version of a COTS product.

Discussion: COTS components provide immediate solutions at a fixed cost, but most applications have a life cycle that spans several releases of those components, which means that it is unrealistic (except in the case of hardware components) to expect the follow-on costs to be zero. In addition to the acquisition cost of the components, the customer and developer need to explore the cost and level of support services as well as opportunities for commodity purchases.

Myth #2: COTS products are cheap.

Reality: Sometimes it is cheaper to build it than to buy especially if considerable adaptation is involved.

Discussion: Often, considerable development is required to use a COTS product—especially when performance requirements are tight and open-interfaces are required. For example, hardware components may have to be hand-tuned when performance margins are unacceptable. In addition, glue code may have to be developed to wrap software components and to adapt them so that they conform to an open interface. Overall costs can be higher than anticipated when these costs are added to those associated with assessment, learning, adaptation, and integration.

Myth #3: The government can control COTS suppliers.

Reality: The market influences COTS suppliers.

Discussion: The size of the current customer and the potential customer base drives the COTS component supplier in determining their response to user needs. At one time government sales drove the market. Times have changed, however, as the commercial marketplace has opened up and DoD has become just another customer for components.

Myth #4: COTS products are specified/selected based on extensive evaluation and analysis.

Reality: COTS products are often selected based on slick demos, web searches, or trade journal articles and ads.

Discussion: Because component-based architecture development is a relatively new field, systems integrators and customers struggle with methods to keep abreast of technology advances and ways to determine which product best suites their needs. Oftentimes, in the rush to make a decision, the choice of COTS products is not based on a strong business case or on the TOC. This problem has been around in various forms for a long time (e.g., Garbage In, Garbage Out) and ways of dealing with it (e.g., trade studies, test labs, independent product certification agencies) can be discriminators used by the customer in reducing risks associated with the acquisition of a COTS-based system. Leaders in the field tried COTS products before they bought them. Others we visited had a large inventory of shelf ware.

Myth #5: You can configure COTS products to meet your requirements.

Reality: You must be prepared to configure your process to meet COTS product capabilities.

Discussion: The 80/20 rule applies to most COTS-based system efforts. The customer can satisfy 80% of their desired business process for 20% of the cost of a custom system (in 20% of the time). Most difficulties occur when a customer or developer believes that the additional 20% is achievable at traditional software development costs. The cost of modifying COTS products, or providing extra functionality is more difficult for the developer because they have little control or insight into how the COTS product was designed, documented, tested, or written/built. This information is usually proprietary and, furthermore, in the light of upgrades and new versions, maintaining compatibility becomes a challenge. Most successful system integrators: 1) <u>never modify COTS components</u> and 2) thoroughly understand the requirements (i.e., assuming an "All requirements are negotiable." approach). If the business case justifies modifying COTS components, then the developer should recommend that. However, this same developer should be concerned about the extra functionality that COTS products bring to the table. Users have to be isolated from this functionality because it may cause the system to do things that it shouldn’t. Testers will have to qualify the system with this added functionality. Acquisition agents will have to pay more for functionality that they don’t use.

Myth # 6: COTS products are a panacea.

Reality: COTS products exacerbate inadequacies in system development processes by compressing the development schedule.

Discussion: To some, COTS components may seem like a silver bullet because COTS components can provide faster, cheaper, and often better solutions for:

- Relatively simple applications
- Mature problem domains
- Using a small number of <u>mature, unmodified</u> components with proven integration mechanisms

But, not all applications fall into this category. Furthermore, the mere fact that applications can be developed so quickly facilitates the possibility that the wrong application will be developed, the wrong COTS components initially selected, and the perceived near-term success will pave the way to long-term disaster.

5.1.2 Process Impact

The following process changes are recommended to avoid common mistakes in the acquisition of COTS-based systems:

- Establish a review process to evaluate the necessity of explicit COTS product requirements in all offerings.
- Include a COTS product risk assessment and mitigation strategy as part of all system evaluations.
- Provide a requirement review process that allows COTS component tradeoffs.
- Establish a policy that stimulates a “try before you buy” philosophy for evaluation.
- Establish a COTS lessons learned data base for future acquisition agents.

- Establish Integrated Product Teams that include software vendors when possible.
- Stimulate improved software licensing practices as part of the process (e.g., enterprise-wide licensing and Air Force wide BOAs or ID/IQ contracts for common items).

5.2 Development of COTS-Based Systems

The development of COTS-Based Systems requires that careful review be placed not only on the architecture, but on the viability of the COTS products being used, vendor stability, licensing, and configuration management due to releases of updated versions of software that may occur during the development phase.

5.2.1 Anecdotal Findings

Myth #7: It's important to know what COTS components can do for you.

Reality: It's important to know what COTS components can do to you.

Discussion: A system architect must always evaluate the "build, buy, or modify" tradeoff when determining how to provide the functionality necessary to meet the customer's requirements. COTS components, in general, provide certain functional capabilities at an extremely attractive initial cost. But, experience shows that that functionality may come with certain limitations and implications. As the number of COTS components to be integrated increases, the dependencies and interplay among them becomes more complex and can lead to intractable problems or to difficult negotiations between different suppliers as to whose product is really at fault when things do go wrong. To complicate matters further, certain "non-functional" requirements, such as security, fault tolerance, or error handling, may not be uniformly supported by all components to the degree necessary to guarantee overall system performance. These potential "hot spots" typically form a list of risks that the architect must trade off in deciding the overall composition of the system under development.

Myth #8: COTS-based systems can be designed "top-down."

Reality: COTS-based systems are built "bottom-up."

Discussion: COTS components facilitate a spiral development model in the sense that functionality can be quickly demonstrated in most applications. In doing so, the customer benefits by early validation of requirements and the developer reduces risks by learning first hand about the capabilities and configuration and integration difficulties associated with the components. Most architects understand this point and do not limit their choice of components by making design decisions too early. They recognize the need to remain flexible in trading off functionality across components until the components are fully proven and the integration mechanisms identified.

Myth #9: An "Open System" architecture solves the COTS component interoperability problem.

Reality: There is no standard definition for "open system" and "plug and play" doesn't always work.

Discussion: Customers and developers clearly recognize the advantages of having plug-compatible components. They like not only having a choice of components, but they like knowing that if one component supplier goes out of business, then there is another source for compatible components. It is debatable as to how successful open system initiatives have been (such as the Defense Information Infrastructure Common Operating Environment (DII-COE) [3]). Most will agree that when it works, it is great, but the number of plug compatible components has yet to reach critical mass. Finally, successful COTS integrators have not confused "open systems" with "open source" systems. Open source software provides an attractive alternative to commercial COTS software packages, especially in the area of software development tools (e.g., operating systems, compilers, editors, and document formatting tools).

Unfortunately, most of these types of components fall into the category of ROTS (Research Off the Shelf) software and can be extremely volatile, thus introducing risk into the overall system that must be planned for. An exception to this rule is some of the freeware subroutine or class libraries that are from reputable sources on the Internet (e.g., the Public Ada Library).

Myth #10: You don't need to test COTS components.

Reality: You need to test COTS components more because you don't understand how they were built.

Discussion: It would be nice if all COTS components worked as advertised. But oftentimes there is a gap between what is advertised and what is delivered. Being that it is economically and oftentimes physically impossible for a COTS vendor to test all its products in combination with all other products under all operating conditions, subtle feature clashes can occur. Furthermore, when developers are trying to leverage emerging technology, oftentimes, marketing pressures force COTS vendors to deliver products with reduced capabilities along with the promise for increased functionality in future versions. Since the system integrator is usually responsible for the overall performance of the system, the system integrator should evaluate all components before they are selected for inclusion in the system.

Users must test COTS components to make sure that they don't perform any unintended functions and have not been tampered with. In particular:

- COTS hardware may contain circuitry that can be activated remotely for fault diagnostics or diagnosis purposes. Personnel working with classified systems are concerned that such circuitry could be activated during time of conflict to reduce system functionality and effectiveness. Worse than that, they are concerned that circuitry embedded at the micron level could render the system useless by an advisory.
- COTS software may contain embedded munitions (e.g., viruses, Trojan horses, backdoors) that could be activated by triggers (high traffic on a network, keywords like alert, etc.) to render software-intensive system useless during time of conflict. Such munitions could be easily hidden in areas that are outside of normally used functionality.

Given the magnitude of the task of testing for the absences of security loopholes, the use of COTS on critical systems needs to be carefully evaluated by the customer and developer.

Myth #11: COTS components come with adequate documentation.

Reality: Features sell COTS components, not documentation.

Discussion: This myth may be thought of as a continuation of the previous myth. The lack of documentation is a risk the system architect faces in determining the suitability of COTS components. Furthermore, in some instances, the customer, upon being exposed to certain component features demonstrated in a certain (sometimes contrived) context, may place unnecessary or unrealistic constraints on the developer's implementation (without adequate justification or flexibility in negotiating for different and possible "better" components).

5.2.2 Process Impact

The development process needs to give special consideration to the following:

- Set up COTS product testing labs with an environment that is close to the one that the product will be used in.
- Enhance the configuration control process to support tailoring of COTS components.
- Establish qualification tests for incoming COTS components.

- Establish a COTS product vendor evaluation and selection process.
- Capture past performance histories on both COTS vendors and components and make them widely available throughout the Air Force.

5.3 Maintenance/Sustainment of COTS-Based Systems

Version harmonization and configuration play a key role in maintaining COTS-based systems.

5.3.1 Anecdotal Findings

Myth #12: Vendors will fix problems in the current release of the product.

Reality: Vendors will fix problems in the next version of the product.

Discussion: The level of service one receives from the component supplier is negotiable. Therefore, unless the contract explicitly states it, the type of problem fixes one receives will be market driven. However, there is often a two-tiered system for repairs where those who pay a premium get their problems handled first. For components critical to system performance, decision makers should opt for the premium. The COTS product industry practices "spiral development," which is a meandering path based on adding features and patching software to correct bugs. Adding features invites problems with "feature clash" mentioned earlier. Correcting problems by spiral development is likely to introduce new bugs. The spiral development process may not converge on a stable product.

Myth #13: Software development is done in the acquisition phase.

Reality: Most software development is done in O&M.

Discussion: The following two observations support this myth.

- *"The maximum shelf life of a COTS software component is eighteen months to two years."* This rule of thumb factors into determining the TOC of an application in that all COTS components that have been configured and integrated together will probably have to be replaced two years after each was first introduced into the marketplace. To complicate matters, each new version of a component might have additional dependencies and possibly introduce new, conflicting functionality. Furthermore the updates may not be released at the same time or validated with the same versions of other components, thus further complicating matters. A new release of an application may, for example, require a newer version of the OS or other cooperating applications.
- *"The half-life of COTS product expertise is 3 to 6 months depending on the marketplace."* This observation is attributable to Kurt Wallnau, Software Engineering Institute who observed that with the fast-paced introduction of new product versions, as well as competing products, there is an unprecedented obsolescence associated with "current" technology. It has been validated by others who have observed that tools have a shorter half-life than systems software, which is about 6 months. The inverse of this ROT is that every 6 months you need to plan on evaluating a new version of a COTS product.

Myth #14: COTS components are free except for the purchase price.

Reality: COTS-based system sustainability issues overwhelm acquisition costs.

Discussion: This myth compliments Myth #2 (i.e., COTS are cheap). Instead of focusing on the purchase price, it looks at the long-term cost of COTS components. That is, annual maintenance and license costs and the expenses for run-time licenses for components that are part of fielded products can quickly increase the overall cost of the component by a factor of 10. This is in addition to the sizable cost of re-configuring and integrating the component into the system, then doing regression testing.

Myth #15: You can ignore vendor upgrades.

Reality: You lose support of back systems if you ignore vendor upgrades.

Discussion: Often you can't ignore vendor upgrades especially when the vendor elects no longer to support the version that you have frozen on.

5.3.2 Process Impact

Because systems are long-lived, sustainability is an important issue in determining the overall effectiveness of a system. Therefore both the government and the systems integrator need to address these issues by:

- Focusing on total ownership cost tradeoffs.
- Establishing a technology refresh process.
- Investigating a spares optimization strategy.
- Negotiating maintenance provisions as part of the initial acquisition (especially licenses).

6 People

All of the process and product changes suggest a new way of doing business when COTS components are involved. New processes must be inserted as the manner in which systems are acquired and sustained moves from a build to a buy orientation. Air Force and contractor personnel must therefore both be educated and trained to accommodate this changeover.

6.1 Issues

Because the roles and responsibilities change during the switchover, the Air Force needs to reeducate and train its workforce. The three primary issues that act as the forcing functions behind this education and training initiative are:

1) Roles and responsibilities of players change radically with the use of COTS components

   Traditional roles and responsibilities change as COTS components are used. For example, contractors may buy COTS components instead of developing them. As part of the procurement process, they must assess, tailor, integrate and determine how and when to refresh the component. These tasks require them to develop new skills, knowledge and abilities especially when they must be accomplished in an environment where they do not have access to the source code. In addition, the contractor will be held accountable for providing functionality that he may not have any control over the evolution. An architecture that considers this lifecycle planning is key to successful COTS use.

2) New risks occur when COTS components are used within Air Force systems

   Traditional risks change as COTS components are used more widely in Air Force systems. Instead of focusing on engineering issues, Air Force Program Managers must pay attention to vendor capabilities, business viability and capacity. They must make sure that products do what they are supposed to and that functionality that is not used does not interfere with overall need – not just a performance issue. Instead of providing contract oversight and direction, they must understand what is being provided and how it can be tailored and used within the context of their overall system architecture.

3) Acquisition, development and sustainment processes change when COTS is employed

   In addition, just about every process used to acquire, develop and sustain the product changes when COTS components are used to provide needed functionality. For example, a “make-buy” milestone must be inserted into the acquisition life cycle in order for the decision to be made in a timely manner. If “buy” is selected, then a procurement cycle starts in order to negotiate favorable license and support agreements with the vendor. Because COTS implies vendor-supplied capabilities, configuration management focuses on version control with traceability down to the piece part level. Quality assurance focuses on evaluation instead of test and standards compliance. Maintenance decisions are made during development as COTS products. The apparent randomness of the COTS revision cycle impacts engineering, maintenance and logistics processes greatly, especially when product updates need to be coordinated to take advantage of new COTS product releases.

6.2 Stakeholder Roles

Education and training issues and associated recommendations as a function of stakeholder involvement when COTS components are employed are summarized in Table 4.

Table 4. Education & Training Issues as a function of Stakeholder Involvement

Stakeholder Community	Training Issues	Recommendations
Customer ▪ Program managers ▪ PEO ▪ Buyer ▪ Program office personnel	▪ USAF personnel must become smart consumers ▪ Must follow systematic process that selects “best value” and emphasizes “try before you buy” ▪ Need to lever buying power via enterprise-wide licensing agreements whenever possible ▪ Must change the reward system to propagate sharing across programs	▪ Develop COTS policies and repeatable processes aimed at deploying them ▪ Capture experience in process use on pathfinder projects for use as case studies in courseware ▪ Train people in processes using case studies ▪ Make sure training becomes part of acquisition core certification requirements ▪ Make sure COTS risks are examined as part of the acquisition process ▪ Should use competitive market forces and “best of breed” solutions whenever possible to get best value for money ▪ Must use CAIV concepts when assessing COTS applicability
User ▪ Field operator	▪ Must increase the end-users awareness of the issues associated with using COTS ▪ Need to shield user from the extra functionality that COTS often provides ▪ Must be willing to participate in "trade-off" analysis for cost and performance.	▪ Educate end-users so they are aware of COTS policies, practices and most important, dangers associated with use ▪ Make sure user manuals and training address COTS usage
Support contractors ▪ FFRDC support ▪ SETA team	▪ Support contractors must provide in-depth technical support for COTS ▪ Must establish standardized evaluation frameworks for COTS selection ▪ Need to provide access to package information, vendor histories and test-beds ▪ Must establish a Air Force wide “past performance” database for COTS vendors	▪ Task support contractors to create the evaluation framework and knowledge bases the Air Force needs to exploit COTS to its fullest ▪ Stimulate support contractors to make this knowledge base available to the Air Force and its contractor community on-line via guidebooks, seminars and self-paced training courses
Maintainer ▪ In-house maintainer ▪ Third-party maintainer	▪ Must increase the maintainer’s awareness of the issues associated with sustaining COTS operationally ▪ Stimulate task them to synchronize COTS updates with their revision cycles ▪ Encourage partnerships with COTS vendors so Air Force problems will be worked as a first priority	▪ Educate maintainers so they are aware of COTS policies, practices and most important, maintenance issues ▪ Ensure that maintenance and support training is provided for COTS to both in-house and third party maintainers
Trainer ▪ In-house trainer ▪ External trainer	▪ Make sure that the trainers are trained relative to COTS use, maintenance and support	▪ Establish a train the trainer program with the COTS vendor

Stakeholder Community	Training Issues	Recommendations
Contractor ▪ Program manager ▪ Engineering team ▪ Integration team ▪ Manufacturing team ▪ Support personnel (CM/QA, contracts, legal, etc.)	▪ Executives must understand the issues relative to the use of COTS components ▪ Processes must be put into place to address the issues ▪ Engineers and support personnel must be trained in the process	▪ Ensure that COTS processes are in place and staff is trained in their effective use prior to awarding a contract ▪ Encourage contractor to have COTS skills, knowledge and abilities updated through periodic retraining
Suppliers ▪ Vendors ▪ Subcontractors ▪ Strategic partners ▪ Co-contractors	▪ Must increase suppliers awareness of Air Force needs relative to COTS ▪ Need to encourage supplier to provide education and training materials with COTS elements	▪ Periodically brief suppliers on Air Force needs ▪ Establish relationships with key COTS suppliers and use buying power to encourage them to provide training
Standards Organizations ▪ National and international groups	▪ Must influence the direction standards organizations take ▪ Must align these directions with Air Force needs ▪ Should stimulate standards organizations to educate and train the workforce as part of their roles and responsibilities	▪ Periodically brief standards organizations on Air Force needs ▪ Establish relationships with these organizations and use government influence to encourage their support
Air Force Research Lab	▪ Must influence the direction standards organizations take ▪ Must align these directions with Air Force needs ▪ Should stimulate standards organizations to educate and train the workforce as part of their roles and responsibilities ▪ Help "track" the industry ▪ Maintain models and simulations that help predict technology evolution – Manage Risk	
Testing Organizations ▪ In-house testers ▪ Third-party testers ▪ Independent V&V teams	▪ Need to establish testing frameworks for COTS ▪ Must get vendor to provide adequate support during system integration and test ▪ Must capture information during test that helps COTS vendor isolate/fix problems ▪ Need to ensure that added functionality does not interfere with system performance ▪ Should establish levels of test associated with degree of COTS integration	▪ Ensure that COTS test processes are in place and staff is trained in their effective use prior to the start of testing ▪ Encourage test organization to develop COTS evaluation and test skills, knowledge and abilities via training
Information Brokers ▪ Product evaluators	▪ Need to share information about COTS products, experiences and suppliers across the Air Force ▪ Stimulate use of this information via education and training programs	▪ Develop the on-line education and training material needed as the database is constructed ▪ Require organizations to become skilled in the use of the information

(This Page Intentionally Blank)

7 Findings and Recommendations

The Air Force should prepare and implement a policy for the acquisition and sustainment of COTS-based systems.

Based on an assessment of 34 programs and organizations, we found that only a few are realizing significant benefits from the utilization of COTS products. Generally, those with the most experience are realizing the biggest gains. Most are struggling with the technology, processes and issues. Everyone is on a steep learning curve. While the concept of a COTS-based system is easily understood, the implementation is not. **COTS is complicated**! The successful application of COTS products impacts virtually every aspect of the acquisition process including acquisition strategy, source selection, program management, system development, integration, and sustainment. COTS is not something you tell people to go do and expect successful results. It requires guidance, training, infrastructure, processes, tools, metrics, incentives, and leadership, otherwise progress will continue to be dismally slow. Therefore, we recommend that the Air Force prepare and promulgate an <u>implementation</u> policy for the acquisition and sustainment of COTS-based systems. The policy should drive the acquisition strategy, source selection, program management and, indirectly, industry as depicted in Fig. 7-1. These important success factors must form the basis of this policy. In the following sections, we identify the key elements of a proposed Air Force (and DoD) COTS implementation policy.

Key Success Factors
1. All operational requirements and procurement specifications are negotiable.
2. Open system architecture and the spiral development process are utilized.
3. The prime contractor is incentivized to provide a credible estimate of support costs.
4. Total ownership cost (TOC) is used as a source selection cost criterion.
5. The contractors past experience employing COTS products is assessed.
6. The contractor’s processes for assessing, selecting, integrating, supporting and refreshing of COTS products are adequate.
7. TOC is used to determine suitability of COTS versus custom products.
8. The contractor’s understanding of the commercial marketplace and relevant COTS products are evaluated.
9. The system application matches the COTS product functionality.
10. The contractor proposes to use the COTS product without modification.
11. Trade-off analyses of all changes versus total ownership cost are conducted.
12. There is continuous interaction between government personnel (operations and acquisition) and the prime contractor in Integrated Product Teams.

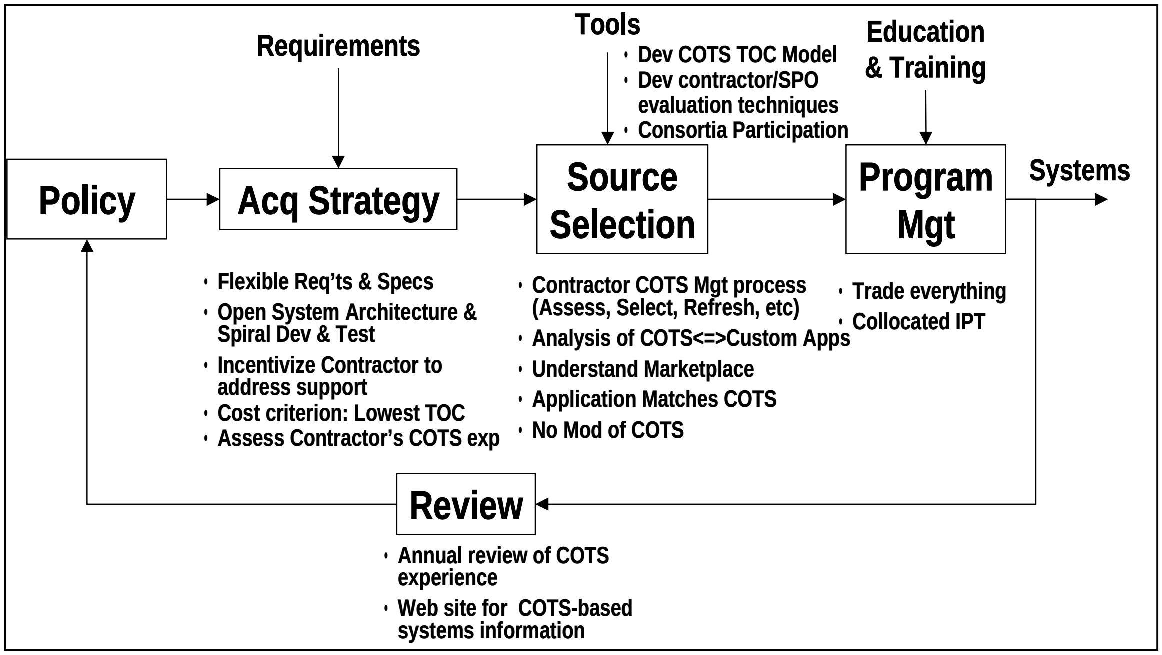

Figure 11. Key Elements of COTS Policy

7.1 Acquisition Strategy

The Service Acquisition Executive should assure that COTS unique aspects are addressed in Acquisition Strategy Review and the Single Acquisition Management Plan.

The Acquisition Strategy Review should assure that all operational requirements and associated contractual technical performance specifications are flexible. If not, then the program is not a candidate for a COTS-based solution. Industry should be involved before the Operations Requirements Document and Request for Proposal are finalized. The end user and the acquisition community need to fully understand what can be accomplished using COTS products before requirements and concepts are cast in concrete.

The acquisition strategy should assure an open system architecture (OSA) where interfaces and protocols are defined by industry standards. OSA protects hardware and software investments for longer periods of time. It provides interoperability with other like and unlike processing platforms. It also allows interfacing mission-specific components, legacy hardware and software, unique sensor systems and high performance specialized processors. New, advanced products that adhere to the OSA can be readily adapted for future system upgrades. And, of course, the supplier base is much more robust offering readily available products at a much lower cost.

The traditional waterfall development process is inappropriate for COTS intensive systems. The waterfall development evolves sequentially from system context to architecture and design followed by implementation. It is typically a build from scratch approach. COTS-based systems are critically dependent on simultaneity, as shown in Figure 12. The system context, architecture and design, and commercial marketplace must be considered concurrently. Rather than build from scratch, the COTS-based approach is to evaluate, buy, integrate and continuously refresh.

The strategy must emphasize total ownership cost (TOC). One of the many advantages of a COTS-intensive system is the ease of incorporating both hardware and software upgrades given an open system architecture. Software suppliers routinely offer upgrades and only support the more recent revisions.

Software upgrades need to be incorporated if for no other reason than to take advantage of the support offered. A COTS-based system must be architected from the outset to accommodate field support and upgrades. If initial acquisition cost is the only cost criteria, then it is distinctly possible that the total ownership cost may be higher than a traditional custom design. It is imperative that innovative contractual techniques be used to hold the contractor accountable for projected total ownership costs.

Finally, because COTS success is so highly dependent on contractor experience, the acquisition strategy must consider contractor's past experience employing COTS products.

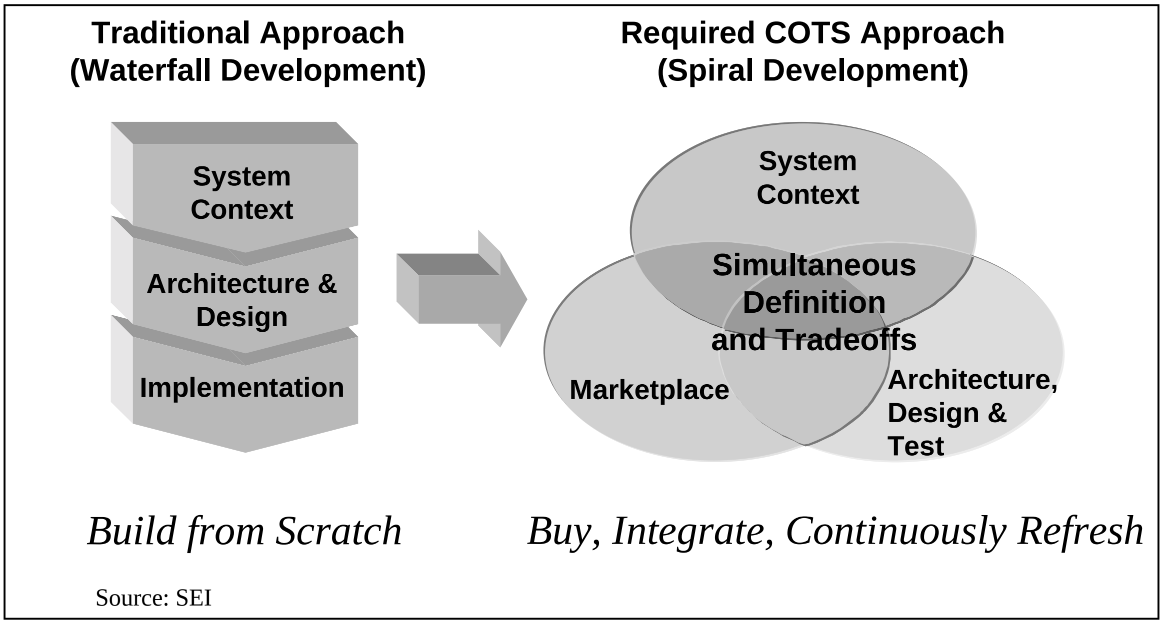

Figure 12. Waterfall versus Spiral Development Process

7.2 Source Selection

The System Program Director must assure that COTS unique aspects are addressed in the source selection process.

The source selection evaluation should take into consideration the bidders' COTS processes for assessing, selecting, integrating, supporting and refreshing commercial products. Because the development of COTS-based systems is so complex, good processes are essential. Processes need to be documented, continuously improved, incorporate lessons learned and used by the organization. An inexperienced team utilizing an ad hoc approach is a sure recipe for failure.

Total ownership cost not initial recurring cost should be used to determine the suitability of COTS versus a custom solution. One of the advantages of a COTS-based system is the ease of upgrades or technology refresh. Technology refresh needs to be an integral part of the development plan. Every design decision needs to take into consideration TOC. If the design is optimized for initial recurring or flyaway cost, it is likely that both TOC and ease of future upgrades will be adversely impacted.

Source selection should assess the contractor's understanding of the commercial marketplace. Contractors must be well aware of relevant COTS products. Since products don't always perform as advertised, a rigorous evaluation is necessary before program commitments are made. Many companies devote a portion of their R&D to such endeavors. The best companies attempt to predict where products that support open system architecture and standards are headed. Choosing the right architecture and anticipating future changes are an important part of the COTS process.

The government should ensure that the proposed COTS product functionality meets the intended system needs. This means that the government as well as industry must have a keen awareness of the commercial products that make up the system. For example, consider the process of building a custom home. A custom home is a COTS-based system. Very few people venture into custom components or building blocks because of the tremendous cost. Most custom homes are an assembly of standard commercial products. An architect and contractor work closely with the client to make individual selections that comprise the house. Very few people would leave these decisions up to the architect. After all the homeowner will live in the house, not the architect. Likewise, the government is the user of the weapon system, not the contractor. It is essential that the government understand fully the COTS products that are utilized.

The government should verify that the bidder proposes the use of COTS products without modification. The study team observed many cases where software or hardware products were modified with disastrous results. One should remember, one of the advantages of COTS products is a built in support system. Once modifications are made suppliers will no longer honor warranties or provide support. Furthermore, subsequent vendor releases will not be consistent with the modified product. If modifications are proposed, then they should be well justified and by exception only. A modified COTS product is no longer a COTS product!

7.3 Program Management

The System Program Director must assure that COTS unique aspects are considered in Trade Studies and integrated product teams.

All proposed changes should accompany a thorough trade study to determine the total ownership cost impact. If TOC is not emphasized during development, the sustainment costs of a COTS-based system can substantially exceed a traditional custom design.

Program management should enforce ongoing interaction between government (operations and acquisition) and prime contractor personnel through integrated product teams (IPTs), with collocation preferable. COTS-based system design requires a concurrent or simultaneous design process. The system context, commercial marketplace, system architecture, design, and test must evolve concurrently (see Figure 12). To maximize the practical use of COTS products, performance trades need to be made constantly. Collocated IPTs greatly facilitate this process.

7.4 COTS-Specific Decision Analysis Tools

The Service Acquisition Executive must assure that the tools are provided to support the successful acquisition of COTS based systems.

Evaluation techniques need to be developed that focus on contractor and program office capability to produce COTS-based systems.

TOC analysis models and tools are required that consider COTS cost factors. These tools will support various decisions including Cost as an Independent Variable (performance, schedule and cost trades), system cost estimating and proposal cost analysis.

As emphasized earlier, it is essential that the government become knowledgeable concerning relevant COTS products and the COTS marketplace. Therefore, they should participate in consortia that share COTS product experiences. The Electronic Products and Systems Consortium (CALCE) at the University of Maryland (http://www.calce.umd.edu) is a good candidate for electronics, particularly electronic components. Their mission is to provide a knowledge and resource base to support development of competitive electronic products and systems in a timely manner. The knowledge resource base includes design and manufacturing methods, simulations techniques, models, experimental methods, guidelines, and instructional information. For software, the Interoperability Clearinghouse's mission (http://www.e-interop.com/) as a public service consortium, is to provide information technology industry a neutral venue for identifying sets of interoperable architecture frameworks necessary to transition the enterprise into the new computing paradigm driven by e-Business, distributed object computing and the internet. Their objectives are to provide lessons learned from multiple implementations to help reduce the complexity, cost, and risk of integrating COTS-based solutions; to advocate the information needs of the enterprise architect; and to provide information technology implementers architecture guidance with validated product interoperability and compatibility data.

7.5 Education and Training

The Service Acquisition Executive must assure that the education and training are provided all levels of the organization.

COTS competency at all levels of the workforce is essential given its complexity and substantial departure from past practices. Audiences, required skills, knowledge and abilities need to be identified. Education and training courses need to be developed. More specifically:

- Require training at the front end of the PEO and DAC programs;
- Encourage DSMC, AFIT to incorporate COTS topics into their curriculum;
- Include COTS material in contracting officer and acquisition officer certification programs; and
- Make COTS guidelines available via the Defense Acquisition Deskbook (DAD).

7.6 Review and Feedback

The Service Acquisition Executive must assure that the lessons learned from COTS based system acquisitions are gathered and disseminated.

Sponsor an annual or periodic review of COTS-based systems experience within the Air Force. The focus should be on lessons learned. This information can be used to update policy, improve infrastructure and enhance training. In addition, a web site for COTS-based systems information including lessons learned, case studies, and references would be very helpful.

(This Page Intentionally Blank)

8 Conclusions

Pervasive changes are needed to exploit the benefits of COTS products.

The average aerospace engineer is between 45 and 50 years old with approximately 25 years of experience. Their job duties have evolved over the years driven to a large extent by the custom design needs of the military. Now we are asking them to radically change the way work gets done as depicted in Table 5.

Table 5. Mil Spec versus COTS

Design & Build (Mil Spec)	Buy & Integrate (COTS)
Requirements driven	Commercial market driven
Specification constrained	Tradeoff oriented
Rigid requirements	Flexible requirements
Unique architecture	Open system architecture
Owner controls product evolution	Market drives product evolution
Stable design	Constant changes
Ignore product evolution	Design for evolution (technology refresh)
Recurring cost emphasis	Total ownership cost emphasis
Make custom hardware	Buy from catalog
Develop software	License software
Unplanned obsolescence	Managed obsolescence
Waterfall-style development	Spiral development
Mil Standards	Widely accepted commercial standards

The same holds true for government personnel. The cultural impact is profound. Successfully fielding a COTS-based system and realizing the anticipated benefits is very difficult to do. Everyone has underestimated its complexity. Like many management edicts, the devil is in the detail and this is a particularly nasty devil.

COTS is <u>not a panacea</u> as many have been led to believe. Expecting major benefits such as lower cost and shorter development times without a major change in the way work is accomplished is a foolhardy notion. Every aspect of acquisition planning, system engineering processes, test planning, etc. must be explicitly crafted to account for COTS issues.

The mentality ought to be “how we can do it” as opposed to “why we cannot”, but not every new requirement can realistically be addressed with a COTS-based solution. The applications must be chosen carefully. The degree of implementation will depend on the specific application. Arbitrary mandates are dangerous. Leadership needs to drive the insertion of COTS products or the system will revert back to the old ways; however, leadership needs to be mindful of the pitfalls. If the experts say it is impractical then listen. Attempting a COTS-based solution where it is entirely inappropriate will end in failure.

COTS is <u>inevitable</u>. Competitive pressures will eventually push most companies to change their ways and adopt good COTS practices. It will be virtually impossible to successfully compete with a traditional custom design approach.

For the government and industry to be successful everyone needs training. New processes need to be established and old ones need to be modified. Roles and responsibilities need to be redefined. Hence, the study team strongly recommends that the Air Force prepare and promulgate an implementation policy for the acquisition and sustainment of COTS-based systems. This policy should drive the acquisition strategy, source selection, program management and, indirectly, industry. The most important success factors need to form the basis of this policy. Ideally, the policy should be adopted by DoD to assure uniformity across the services and in keeping with the Single Process Initiative.

Appendix A Terms of Reference

The SAB was requested to provide process guidelines to substantially improve the results of COTS products.

Ensuring Successful Implementation of Commercial Items In Air Force Systems

BACKGROUND: The Air Force is increasingly using commercial items or commercial-off-the-shelf (COTS) based systems in new systems as well as upgrades/modifications to existing systems. This is particularly true in systems and subsystems involving information technology. In some cases, this is done to lower cost while maintaining performance. In other cases, it is done to improve performance at equal or lower cost.

While the goals are good, there is currently no set of standards or processes that ensure that the Air Force (or DoD, for that matter) can maintain cost, performance, and reliability goals subsequent to an initial investment in or application of COTS in any form. There is much anecdotal evidence that the lack of standards and processes has led to failure and additional cost. The purpose of this effort is to develop a set of standards and processes that the Air Force can implement (and continue to improve) resulting in high assurance that the positive aspects of COTS implementation are realized. The goal is that this set of standards and processes will extend beyond the Air Force to all of DoD. What is needed is a "check list" of actions that need to take place to ensure the integration of COTS (hardware and software) into Air Force systems results in products that:

1) Perform as advertised initially and through subsequent upgrades,
2) Are affordable through their life cycle,
3) Are safe (i.e. safety is assured in the process), and
4) Are supportable (not made obsolete by a vanishing or changing industrial base).

STUDY PRODUCTS: Briefing to SAF/OS & AF/CC in October 1999. Publish report December 1999.

CHARTER: To develop case studies and standards which

1) Gather and analyze case studies as broad a set of experiences as possible from DoD and the commercial sector noting successes, failures, and reasons for each. Also, note where COTS was used "as is”, where it was “remanufactured”, to what extent, and why.
2) Identify, if any, standards and processes that were in place for each case study. Compare with standards and processes that would have been in place were COTS not used (instead, the system or subsystem was developed in DoD).
3) Analyze failures and successes in processes, standards, and implications (e.g. cost, supportability, etc.).
4) Identify information security/assurance issues and define guidelines, particularly for information systems.
5) Analyze issues related to training (operators and maintenance), documentation (for maintenance and system upgrades), spares management, and testing. Suggest guidelines.
6) Define a set of implementation measures (standards and processes) that include goals and metrics to determine what steps to take to ensure success using COTS as well as when the use of COTS is not advisable and why. Suggest methods for government/Air Force oversight.

(This Page Intentionally Blank)

Appendix B Panel members

Panel members have extensive experience in COTS products, defense acquisition, and commercial industry.

Mr. Jeffrey E. Grant, Chair
Corporate Vice President (Ret.)
Hughes Electronics

Dr. Robert Rankine Jr. (Maj Gen USAF Ret)
Vice President Government Requirements
Hughes Space and Communications

Mr. Kenneth M. Brown
Director, Sensors and Electronic Systems
Raytheon Systems

Mr. William R. Carter
Program Manager, Advanced Information Systems
Lockheed Martin

Mr. John Foreman
Manager, COTS Based Systems Program
Software Engineering Institute

Mr. Don Reifer
President
Reifer Consultants, Inc.

Dr. Will Tracz
Senior Software Engineer
Lockheed Martin Federal Systems Owego

Dr. Nick Tredennick
Consultant
Army Science Board Member

Mr. Frank Willis
Director Business Development
DY 4 Systems, Inc.

Lt Col Paul Schubert
Executive Officer
SAB Secretariat

Capt Rob Block
Tech Editor
USAF Academy

(This Page Intentionally Blank)

Appendix C Committee Meetings

COTS Kickoff
23 March 1999
Colorado Springs CO: SEI

Software Information Gathering Meeting
15 April 1999
Washington DC: ASD/C3I, SAF/AQX and Interoperability Clearinghouse

Information Gathering Meeting
5-6 May 1999
Salt Lake City, UT: AFOTEC, Boeing (Boldstroke, Open Systems, DCAC/MRM, PVS/EVS), DY 4 Systems, GPS, GTE and TRW (Large ADP Systems & Software Development Process)

COTSCON Conference
11-12 May 1999
McLean, VA: AWACS, Bradley Fighting Vehicle and JDAM

Information Gathering Meeting
13 May 1999
Wright-Patterson AFB, OH: ASC/EN, AFRL (F-22, F-15E, F-16, B-2, T-38, T-6, F-117/119 Engines, JASPO and Mobility SPO)

Information Gathering Meeting
19 May 1999
Lockheed Martin, Manassas, VA: New Attack Submarine and Rapid COTS Insertion

Information Gathering Meeting
8 June 1999
Hughes Washington Office, Rosslyn, VA: AAAV – General Dynamics, AFPEO/LI, CALCE and GBS - Raytheon

SAB Summer Session
14 – 25 June 1999
Irvine, CA: B-2 – Northrop Grumman, MRP II and TacTech

COTS Report Redline Session
26 October 1999
Hughes Washington Office, Roslyn VA

(This Page Intentionally Blank)

Appendix D Open Issues

D.1 Security

Security issues for COTS products are unsolved.

D.1.1 Software

COTS software is not secure. Windows and Unix are not secure. In fact, there is no trusted COTS operating system. A major concern is the inability to detect trap doors or "Trojan Horses." In addition software products may interact with each other in ways that create security holes. Security analysis of complex software systems has always been a serious challenge with many open research issues. Most COTS software is delivered in executable form with no source code, making traditional analyses impossible.

It has been suggested[4] that a third party arrangement offers the possibility of reducing the security risk while retaining much of the cost advantage associated with the use of COTS software. In this arrangement, a third party would procure limited rights from a COTS supplier to modify and maintain software for a specified military application only. The third party would primarily be responsible for assuring the integrity of the software by detecting and eliminating the presence of a Trojan horse or other deleterious acts. In addition, this arrangement would permit the user to determine the upgrade path and not have to react to the rapid pace of the commercial market place. Of course, these services would be at an additional cost. However, it is expected that the net cost while higher than a normal COTS product implementation would be substantially less than custom code. The compelling cost rationale for such an arrangement is directly related to the cost to develop code. A line of custom code costs approximately a 100 times more than a comparable line of COTS code. The difference is simply due to the fact that the development cost of a line of COTS code is amortized over many customers unlike custom code.

The business case is less clear. It is unlikely that private industry would be motivated to enter into this hypothetical third party COTS market. On the other hand, a non-profit special government contract service agency may be more than willing to participate. There are indications that at least some COTS suppliers would be willing to enter into a third party arrangement. Their concerns may be eased somewhat knowing that a responsible agency was involved.

D.1.2 Hardware

Simple, secure authentication is difficult. Components, boards, peripherals, or systems might be compromised. For example, consider the following hypothetical, perhaps extreme, scenario. KeyKing is a PC keyboard manufacturing company. Keyboards are its only product. KeyKing makes the best and cheapest keyboards in the world (it can afford to do both since it is subsidized). KeyKing specializes in USB keyboards but also make PS/2-style keyboards for legacy systems. KeyKing bids aggressively for OEM contracts with the PC manufacturers (e.g., Dell, Gateway, Compaq, and IBM).

[4] Dr. Edward A. Feigenbaum, former Chief Scientist of the Air Force

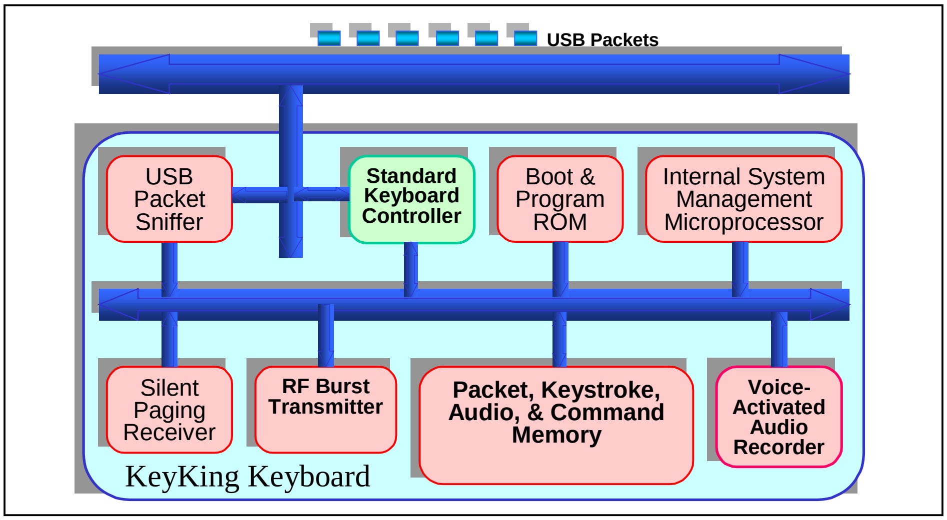

Figure D-1. KeyKing Keyboard Block Diagram

From the outside, the KeyKing product appears to be an ordinary, high-quality keyboard. It even has a tamper-proof molded housing. Inside, it includes the components shown in the block diagram above. Not every unit is built with this design, only those that are likely to accompany a military computer. A broadcast transmission keyed to the unique code for each Silent Paging Receiver wakes the unit and activates the Internal System Management Microprocessor. This microprocessor manages the USB Packet Sniffer, Voice-Activated Audio Recorder, and RF Burst Transmitter. The USB Packet Sniffer copies all of the traffic on the USB (this might, for example, include anything sent to the printer). The microprocessor includes programs for filtering the USB packets, capture, analysis, and filtering of keystrokes, and for data compression and assembly of burst transmission packets. Program control commands come through the Silent Paging Receiver to tell the unit whether to capture audio, for example, and to schedule transmission times.

This keyboard could unobtrusively capture account names and passwords. It would unobtrusively monitor all activity on the keyboard and on the USB. Its only real exposure is during RF burst transmit periods, but these might be scheduled for times when they would be masked by other office activity or when no one was around.

Normally, this keyboard operates passively with respect to the operation of the computer, but it could also be used to deny the user access to the computer (by ignoring key inputs) and it could actively send keystrokes to the computer. This feature could be activated by a broadcast command to single keyboards, or to various groups of keyboards through the broadcast identification code as interpreted by the paging receiver. Components for the keyboard are all commercially available and are estimated to cost less than $25.

D.1.3 Recommendations

It is strongly recommended that COTS products, particularly software not be used for critical applications.

COTS Security is deserving of its own study. It is a very complex subject with many unanswered questions.

Appendix E Resources

E.1 COTS Links

Link	Comment	Rating
Using COTS Software in Systems Development	Good site with papers and ICSE 99 proceedings.	*****
ACQWeb Homepage (search for COTS)	All kinds of COTS projects and presentations.	*****
http://stsc.hill.af.mil/crosstalk/	Crosstalk – Search for COTS (includes some from SEI)	****
COTS Papers and Books	Indexes white papers on COTS technologies and issues surrounding procurement by the DOD.	***
COTS Resources – Information links	Indexes information links on COTS technologies and issues surrounding procurement by DOD (shows the SEI).	***
COTS: What it means to Mercury	Mostly hardware related, but several general COTS links, including COTS 95 and COTS in Canada conference presentations.	***
COCOTS – Other COTS Related Sites	COCOTS Information and links to some COTS projects, conferences and Y2K sites.	***
US ARMY questions for milestones (see Domain 2: COTS/GOTS Business Strategy)	Army suggestions for COTS strategy.	**
DISA Year 2000 COTS Product Compliance Catalog	COTS Product Y2K Compliance Information and links.	*

E.2 Conferences/Seminars

NRC Software Engineering Seminar - 1998	
OTS Worshop (SES 98)	
COTS in Canada 96	
COTS 95	
COTScon Conferences (Military & Aerospace Electronics) http://www.milaero.com/cotscon.htm	

E.3 COTS Organizations

Organization	Category	People	Projects and Focus
		More COTS	
USC Center for Software Engineering	Education	Dr. Barry Boehm	Constructive COTS (COCOTS) cost estimate model
IIT Software Engineering Group	Canada	Dr. Morven Gentleman, John C. Dean, Dr. Mark Vigder	Active work on COTS evaluation and integration.
MITRE	FFRDC		Management Guide to Software Maintenance, DISA Year 2000 COTS Product Compliance Catalog, "What Rots about COTS" , and other

Organization	**Category**	**People**	**Projects and Focus**
			relevant papers.
Institut f¨ur Informatik	Germany	Bernhard Deifel	Requirements Engineering of complex COTS, 3yr project
Reliable Software Technologies	Commercial	Jeffrey Voas	Consulting firm in VA, Java and software reliability, Funded by AFRL for Dynamic Security Analysis of COTS Applications. Also look at presentations and papers.
American Management Systems	Commercial		Air Force's Arnold Engineering Development Center (AEDC) work, COTS listed as a core competency , DoD financial management solution used by several government agencies, DoD standard procurement system
Lockheed Martin	Commercial		Claim they have an expertise in COTS and mention DMS on same page
NDIA – Information Technology Committee	Non-profit Association		Use of Commercial-Off-The-Shelf (COTS) in Major Programs
Johns Hopkins	Educational		Software Size and Cost Estimating class teaches COTS (COCOTS?)
www.calce.umd.edu/			CALCE, University of Maryland
www.e-interop.com			Interoperability Clearinghouse
http://www.sei.cmu.edu/cbs/	FFRDC		Carnegie Mellon Software Engineering Institute COTS-based systems initiative
http://www.bmpcoe.org/index.html/			Best manufacturing practices
http://members.tripod.com/~NavyPats/index3.htm			Market survey, COTS product selection, technology trending and product evaluation
Hitachi	Commercial		COTS-Based systems development paper at ICSE 98
Less COTS			
Lawrence Livermore National Laboratory	DOE, Education		Safety focus, COTS in mission critical systems
The RTC Group	Commercial		COTS Journal (more hardware then software)
School of Informatics City University London	Education	Dr Neil Maiden, Cornelius Ncube	Procurement Oriented Requirements Engineering Method (PORE) - process for COTS product selection. Process being evaluated by 1 or 2 banks in UK
Chalmers University of Technology	Education		Vulnerability analysis (1 COTS paper)
Hewlett Packard Laboratories	Commercial		Component sandbox to restrict COTS components (1 idea…several papers)
San Jose State University	Education		
CMU-Institute for Complex Engineered Systems	Education		Ballista - automatically test commercial off-the-shelf (COTS) software for and harden against robustness failures. (Industry sponsor in 1999-2000?)
Mercury Computer Systems	Commercial		Based in Chelmsford, Massachusetts, real-time imaging, Hosted COTS 95, COTS Page – mostly hardware, but some general COTS concepts.
Computer Systems Group (CSG) at the University of Waterloo	Education		

Organization	Category	People	Projects and Focus
DY4	Commercial		COTS hardware focus, but discusses product selection and software reliability issues. COTS Handbook , COTS newsletter, and examples of COTS programs.
Intermetrics (now AverStar)	Commercial	Ronald Kohl	V&V of COTS Dormant code, similar speech to NASA in WV
US Government			
Army	DoD		COTS Software Evaluation Workshop - Army Information Technology (IT)
Navy	DoD		Navy Product And Technology Surveillance (PATS) – mostly hardware
Defense Acquisition Deskbook, www.deskbook.osd.mil	DoD		*SD-2 Buying Commercial and Nondevelopmental Items: A Handbook*; and *Commandments of COTS: In Search of the Promise Land*
http://cmmr.crane.navy.mil/index.html			Commercial item military market research information center
http://pats.crane.navy.mil/defaulthome.com			Implementation of a disciplined engineering process addressing product surveillance, technology trends and solution evaluation

E.4 COTS Bibliography

Title **Automatically Detecting Mismatches during Component-Based and Model-Based**

Authors Alexander Egyed, Cristina Gacek
Published ASE'99
Source USC-Center for Software Engineering
Date 01-May-99
URL http://sunset.usc.edu/TechRpts/Papers/usccse99-518/usccse99-518.pdf
Abstract A major emphasis in software development is placed on identifying and reconciling architectural and design mismatches. Those mismatches happen during software development on two levels: while composing system components (e.g. COTS or in-house developed) and while reconciling view perspectives. Composing components into a system and 'composing' views (e.g. diagrams) into a system model are often seen as being somewhat distinct aspects of software development, however, as this work shows, their approaches in detecting mismatches complement each other very well. In both cases, the composition process may result in mismatches that are caused by clashes between development artifacts. Our component-based integration approach is more high-level and can be used early on for risk assessment while little information is available. Model-based integration, on the other hand needs more information to start with but is more precise and can handle large amounts of redundant information. This paper describes both integration approaches and discusses their commonalties and differences. Both integration approaches are automateable and some tools support is already available.

Title **Why COTS Software Increases Security Risks**

Authors Jeffrey Voas
Published ICSE Workshop on Testing Distributed Component-Based Systems
Source Reliable Software Technologies
Date 01-May-99
URL ftp://ftp.rstcorp.com/pub/papers/ses.ps

Abstract Understanding the risks inherent in using COTS software is important because information systems today are being built from ever greater amounts of reused and prepackaged code. Security analysis of complex software systems has always been a serious challenge with many open research issues. Unfortunately, COTS software serves only to complicate matters. Often, code that is acquired from a vendor is delivered in executable form with no source code, making some traditional analyses impossible. The upshot is that relying on today's COTS systems to ensure security is a risky proposition, especially when such systems are meant to work over the Internet. This short paper touches on the risks inherent some of today's more popular COTS systems, including Operating Systems and Java Virtual Machines.

Title **Depending on COTS**

Authors Kevin Gooder

Published Course work for COMP5990, Professor William C. Hoffman, Jr., Sponsored by the Office of Under Secretary of Defense for Acquisition and Technology (OUSD/A&T)

Source Webster University

Date 14 December 1998

URL

Abstract Commercial Off-The-Shelf (COTS) software components are being used by the military in increasing numbers. This report focuses on the Global Positioning System's (GPS) Operational Control Segment (OCS) conversion from a Legacy mainframe to a distributed, open systems, computer architecture highly dependent upon COTS software. Although the military is moving towards increased utilization of COTS products, this report details lessons learned and potential risks associated with the selection, integration, and use of COTS software products as experienced in the specific case of the GPS OCS. This report is intended to assist the Office of the Under Secretary of Defense for Acquisition and Technology (OUSD/A&T) in further refining Department of Defense (DoD) policies associated with the acquisition process as they pertain to the selection and integration of COTS software into mission critical systems.

Title **Why COTS Software Increases Security Risks**

Authors Jeffrey Voas

Published ICSE Workshop on Testing Distributed Component-Based Systems

Source Reliable Software Technologies

Date 01-May-99

URL ftp://ftp.rstcorp.com/pub/papers/ses.ps

Abstract Understanding the risks inherent in using COTS software is important because information systems today are being built from ever greater amounts of reused and prepackaged code. Security analysis of complex software systems has always been a serious challenge with many open research issues. Unfortunately, COTS software serves only to complicate matters. Often, code that is acquired from a vendor is delivered in executable form with no source code, making some traditional analyses impossible. The upshot is that relying on today's COTS systems to ensure security is a risky proposition, especially when such systems are meant to work over the Internet. This short paper touches on the risks inherent some of today's more popular COTS systems, including Operating Systems and Java Virtual Machines.

Title **Dependability Certification of Software Components**

Authors Jeffrey Voas

Published EASE 99

Source Reliable Software Technologies

Date 15-Apr-99

URL http://www.rstcorp.com/presentations/ease99/

Abstract Presentation

Title **DOD exec offers an off-the-shelf caveat**

Authors Christopher J. Dorobek

Published Government Computer News

Source Air Force

Date 12-Apr-99

URL http://www.gcn.com/gcn/1999/April12/50.htm

Abstract Air Force's Frye says commercial products are good for many uses, but sometimes carry hidden costs

Title **What Rots about COTS: Hidden Risks of Commercial Software**

Authors Vic Demarines

Published

Source MITRE

Date 01-Mar-99

URL

Abstract Presentation (available from Lorrain) The myth and reality of COTS, including discussion of cost, complexity and security.

Title **Commercial-Off-The-Shelf (COTS) Software: Five Key Implications for the**

Authors Kurt Wallnau
Published Software Tech News, Volume 2 Number 3 - Software Architecture
Source SEI
Date 01-Mar-99
URL http://www.dacs.dtic.mil/awareness/newsletters/technews2-3/cots.html
Abstract Some of the most significant changes that have confronted DoD software acquisition efforts in the past few years are the result of using Commercial Off-The-Shelf (COTS) software. However, these changes are not unique to the DoD, virtually all segments of US Government and industry have been forced to deal with the implications of COTS software. These changes are the inevitable and irreversible consequence of increasing industrial and social reliance on computing technology. And if this assertion is not convincing to the DoD program manager, there is a range of Government and DoD acquisition policies, guidelines, and directives that provide more than ample motivation for using COTS software. The implications of COTS software on DoD software acquisition are many and varied, as suggested by the SEI monograph series on COTS software. This short article is focused more narrowly on the topic of COTS software on software architecture. To side step the issue of what is meant by "architecture," this article examines how COTS software affects the strategies and tactics employed by the successful system architect or lead designer. Although this article focuses on the architect, DoD program managers and executives will find this information useful in understanding the issues faced by integration contractors, and in assessing how well integration contractors are responding to these issues.

Title **Architecture for Software Construction by Unrelated Developers**

Authors W.M. Gentleman
Published First Working IFIP Conference on Software Architecture 1999, San Antonio, Texas
Source IIT Software Engineering Group
Date 24-Feb-99
URL http://wwwsel.iit.nrc.ca/abstracts/NRC41613.abs
Abstract Suppose one COTS (Commercial Off the Shelf) software supplier provides an interpreter for a problem oriented language, another provides an application generator for producing numerical solvers for a class of partial differential equations, and a third produces a visualization package. A team of domain specialists writes scripts in the problem oriented language to define cases to be solved, uses the application generator to produce an appropriate solver, solves the generated PDE, and uses the visualization package to analyze the results and adjust the description of cases. Such examples illustrate that large and long lived software systems can result from the combined effort by various unrelated development organizations, organizations not even known to one another. No single design authority, to which the others report, has overall system responsibility. Such examples also illustrate the importance for software architecture to include relationships between entities that exist and are used during the construction process, instead of focusing only on relationships between entities that exist at runtime. The needs for software architecture for such systems are not well met by the existing literature.

Title **A Model for Version Planning of CCOTS**

Authors Deifel B.
Published Proceedings of SCE'99, Los Angeles
Source Institut f¨ur Informatik
Date 18-Feb-99
URL http://www.forsoft.de/teilprojekte/a4/publications/Sce99.zip
Abstract Complex commercial off the shelf software (CCOTS) is usually developed in separated products and in versions to be able to bring out new features fast and to react flexibly on changes on the market. This influences especially the requirements engineering since a range of future versions has to be planned in advance. Typical problems which have to be managed during version planning are the handling of parallel development of different products, changes of requirements and the distribution of requirements over products and versions. In this proposal paper we sketch a model for version planning of CCOTS. The model defines relationships between different products as well as relationships between requirements and versions. We show how the model allows for flexible changes of requirements, automatic generation of different views at the version plan and how

Title **Are COTS Products and Component Packaging Killing Software Malleability?**

Authors Jeffrey Voas
Published ISCM 98
Source Reliable Software Technologies
Date 20-Nov-98
URL http://www.rstcorp.com/presentations/icsm98/
Abstract Presentation

Title **Building Maintainable COTS Based Systems**

Authors Dr. M. R. Vigder and J. C. Dean
Published International Conference on Software Maintenance, Washington DC
Source IIT Software Engineering Group
Date 13-Nov-98
URL http://wwwsel.iit.nrc.ca/abstracts/NRC41611.abs
Abstract Maintaining large software systems based on Commercial Off-The-Shelf (COTS) components is a major cost driver for these systems. Maintenance includes activities from component replacement to trouble-shooting and configuration management. The maintenance costs for COTS based software systems can be reduced by building systems according to specific design criteria. This paper identifies the major activities of a system maintainer, describes the properties that can be designed into a system to facilitate these activities, and outlines a checklist of items that can be verified during a design or code review, or during the evaluation of a COTS components in order to guarantee these properties are built into the system. The verification is illustrated using a photo imaging system that is currently under development.

Title **A Management Guide to Software Maintenance in COTS-Based Systems**

Authors Judith Clapp, Audrey Taub
Published MITRE
Source MITRE
Date 01-Nov-98
URL http://www.mitre.org/resources/centers/sepo/sustainment/manage_guide_cots_base.html
Abstract The objective of this guidebook is to provide planning information that results in cost- effective strategies for maintaining Commercial Off-the-Shelf (COTS) software products in COTS-based systems. It considers the issues and risks in using COTS software over the life cycle and how to control them. It describes changes in the software maintenance process that are needed to manage a COTS-based system. It provides guidance in developing a COTS Software Life-Cycle Management Plan.

Title **An Architecture for COTS Based Software Systems**

Authors Mark Vigder
Published
Source IIT Software Engineering Group
Date 01-Nov-98
URL http://wwwsel.iit.nrc.ca/abstracts/NRC41603.abs
Abstract This document describes the software architecture issues from the perspective of those who are responsible for managing, acquiring, designing, building, and maintaining COTS-based software systems. Its purpose is to identify the activities and processes associated with integrating, maintaining, and managing COTS based software systems when minimal control is exercised over the individual COTS components. These activities cover the lifecycle of the system from initial integration and testing, through to replacing components, upgrading the system, and performing configuration management.

Title **Inspecting COTS Based Software Systems**

Authors Mark Vigder
Published
Source IIT Software Engineering Group
Date 01-Nov-98
URL http://wwwsel.iit.nrc.ca/abstracts/NRC41604.abs
Abstract This document provides a framework for developing an inspection checklist to determine whether a COTS based software system possesses the desired architectural properties that facilitate the ongoing management of the system. The framework is designed as a set of questions that can be used to develop a complete inspection checklist. The purpose of the checklist is to have a set of items that inspectors can be looking for as they inspect the design and the code. The questions are designed to be applied during the acquisition, construction, and maintenance processes in order that the properties are constructed with the system and preserved during maintenance.

Title **Measuring (Assessing) and Maintaining COTS-Based Software Systems**

Authors Jeffrey Voas
Published
Source Reliable Software Technologies
Date 08-Oct-98
URL http://www.rstcorp.com/presentations/orlando98/
Abstract Presentation

Title **Tolerant Software Interfaces: Can COTS-based Systems be Trusted Without**

Authors Jeffrey Voas, F. Charron, and K. Miller
Published SAFECOMP'96, Vienna
Source Reliable Software Technologies
Date 01-Oct-98
URL ftp://ftp.rstcorp.com/pub/papers/safecomp96.pdf
Abstract We have investigated an assessment technique for studying the failure tolerance of large-scale component-based information systems. Our technique assesses the tolerance of the interfaces between component objects in order to predict how the software will behave if anomalous failures exit certain components and enter others. (Note that we are not talking about graphical user interfaces, but rather the mechanisms that link software components together.) These failures can originate from incorrect code, bad input data from a failed hardware devices, or bad input data from human operators. Our approach is applicable to systems for which source code is available, as well as systems for which no source code is known (e.g., systems composed from executable Commercial Of-The-Shelf (COTS) components), and addresses several of the larger problems associated with software

Title **Proceedings of the California Software Symposium (October 1998) - "COCOTS**

Authors Barry W. Boehm, Christopher M. Abts, Elizabeth K. Bailey
Published
Source USC-Center for Software Engineering
Date 01-Oct-98
URL http://sunset.usc.edu/COCOTS/docs/CSS98/COCOTS_briefing.pdf
Abstract Software engineering in a fully connected world relies increasingly on the integration and tailoring of commercial-off-the-shelf (COTS) software components. This pre-existing software is from commercial vendors who supply self-contained off-the-shelf components that can be plugged into a larger software system to provide capability that would otherwise have to be custom built. The two primary distinguishing characteristics of this COTS software are 1) that its source code is not available to the application developer, and 2) that its evolution is not under the control of the application developer. .... The most significant factors driving COTS integration costs have been identified and mathematical forms for a set of four submodels incorporating those costs have been generated. In its current form COCOTS offers insight into the development costs of using COTS components. With extensions planned in the near future intended to address the entire system lifecycle, including acquisition and O&M costs, COCOTS is on its way to capturing all significant costs associated with using COTS software.

Title **Moving Toward Component-Based Software Development Approach**

Authors Gilda Pour
Published Proceedings of the Technology of Object-Oriented Languages and Systems
Source San Jose State University
Date 01-Sep-98
URL http://computer.org/conferen/proceed/Tools-27/9096/90960296abs.htm
Abstract The new trend is to move from the traditional software development approach, which focuses on building software systems from scratch, to component-based software development approach, which revolutionizes how software systems are built. The focus of this new approach is on development of new systems by selecting and assembling a set of off-the-shelf components within an appropriate software architecture. On one hand, the use of off-the-shelf components has led to a great potential for: (1) significantly reducing cost and time to market of large-scale and complex software systems, (2) improving system maintainability and flexibility by allowing new components to replace old ones, and (3) enhancing system quality by allowing components to be developed by those who are specialized in the application area, and systems to be built by software engineers who are specialized in component-based software development. On the other hand, the use of commercial off-the-shelf software--delivered as black box components--has raised a few major technical and non-technical issues. This paper explores those issues, and discusses several directions for future research that would help to expand the use of component-based software development approach.

Title **Have We Forgotten a Few Things in the Euphoria Over COTS?**

Authors Norman F. Schneidewind
Published OTS Workshop (SES 98)
Source Naval Postgraduate School
Date 11-Aug-98
URL http://www.rstcorp.com/ots/COTS.ppt
Abstract Presentation

Title **Road to an ISO Standard for 'Off-the-Shelf Software'**

Authors Stan Magee
Published OTS Workshop (SES 98)
Source

Date 11-Aug-98
URL http://www.rstcorp.com/ots/Monterey11.ppt
Abstract Presentation

Title **Measuring and Maintaining COTS-Based Software Systems**
Authors Jeffrey Voas
Published OTS Workshop (SES 98)
Source Reliable Software Technologies
Date 11-Aug-98
URL http://www.rstcorp.com/ots/orlando981.ppt
Abstract Presentation

Title **COTS Cartoon: What does (and does not) get talked about.**
Authors Ronald Kohl
Published OTS Workshop (SES 98)
Source Intermetrics
Date 11-Aug-98
URL http://www.rstcorp.com/ots/Cartoonv21.ppt
Abstract Picture only

Title **V&V of COTS Dormant Code: Challenges and Issues**
Authors Ronald Kohl
Published OTS Workshop (SES 98)
Source Intermetrics
Date 11-Aug-98
URL http://www.rstcorp.com/ots/DormantCodeSES981.ppt
Abstract Presentation

Title **Managing Long-Lived COTS Based Systems**
Authors M.R. Vigder and J.C. Dean
Published OTS Workshop (SES 98)
Source IIT Software Engineering Group
Date 11-Aug-98
URL http://www.rstcorp.com/ots/Position1008981.ppt
Abstract Presentation

Title **Off-the-Shelf Software Components in Systems Important to Safety**
Authors Thuy Nguyen
Published OTS Workshop (SES 98)
Source EDF - France
Date 11-Aug-98
URL http://www.rstcorp.com/ots/ots981.ppt
Abstract Presentation

Title **Why COTS Software Increases Security Risks**
Authors Gary McGraw
Published OTS Workshop (SES 98)
Source Reliable Software Technologies
Date 11-Aug-98
URL http://www.rstcorp.com/ots/COTS-security1.ppt
Abstract Presentation

Title **Security Control for COTS Components**
Authors Qun Zhong, Nigel Edwards
Published IEEE Computer, Vol. 31, No. 6, pp. 60-67
Source Hewlett Packard Laboratories
Date 01-Aug-98
URL http://computer.org/computer/co1998/r6067abs.htm

Abstract Using COTS components to build large-scale information systems can reduce costs, but it can also pose serious threats to system security. The authors analyze the risks and describe how their sandbox method can confine the damage potential of COTS components.

Title **A Map of Security Risks Associated with Using COTS**

Authors Ulf Lindqvist, Erland Jonsson
Published IEEE Computer, Vol. 31, No. 6, pp. 67-73
Source Chalmers University of Technology
Date 01-Jul-98
URL http://computer.org/computer/co1998/r6060abs.htm
Abstract Combining Internet connectivity and COTS-based systems results in increased threats from both external and internal sources. Traditionally, security design has been a matter of risk avoidance. Now more and more members of the security community realize the impracticality and insufficiency of this doctrine. It turns out that strict development procedures can only reduce the number of flaws in a complex system, not eliminate every single one. Vulnerabilities may also be introduced by changes in the system environment or the way the system operates. Therefore, both developers and system owners must anticipate security problems and have a strategy for dealing with them. This is particularly important with COTS-based systems, because system owners have no control over the development of the components. The authors present a taxonomy of potential problem areas. It can be used to aid the analysis of security risks when using systems that to some extent contain COTS components.

Title **Automated Robustness Testing of Off-the-Shelf Software Components**

Authors Nathan P. Kropp, Philip J. Koopman, Daniel P. Siewiorek
Published Proceedings of FTCS'98
Source CMU-Institute for Complex Engineered Systems
Date 23-Jun-98
URL http://www.cs.cmu.edu/afs/cs/project/edrc-ballista/www/ftcs98/index.html
Abstract Mission-critical system designers may have to use a Commercial Off-The-Shelf (COTS) approach to reduce costs and shorten development time, even though COTS software components may not specifically be designed for robust operation. Automated testing can assess component robustness without sacrificing the advantages of a COTS approach. This paper describes the Ballista methodology for scalable, portable, automated robustness testing of component interfaces. An object-oriented approach based on parameter data types rather than component functionality essentially eliminates the need f or function- specific test scaffolding. A full- scale implementation that automatically tests the robustness of 233 operating system software components has been ported to ten POSIX systems. Between 42% and 63% of components tested had robustness problems, with a normalized failure rate ranging from 10% to 23% of tests conducted. Robustness testing could be used by developers to measure and improve robustness, or by consumers to compare the robustness of competing COTS component libraries.

Title **Supporting Reuse and Flexibility in CCOTS Variation Development**

Authors Deifel B.
Published Proceedings of REFSQ'99, Heidelberg
Source Institut f¨ur Informatik
Date 08-Jun-98
URL http://www.forsoft.de/teilprojekte/a4/publications/Refsq99a.zip
Abstract Different requirements of different market segments force organizations to develop complex commercial of the shelf software (e.g. Microsoft Office, SAP R/3, Siemens SIMATIC, shortly CCOTS) in variations. Variations are adaptations of software to the specific needs of a group of customers. A lack of a systematic support of variation development, however, often leads to complex historically grown systems of variations. In this paper we present a description technique supporting a systematic development of variations. Especially the description technique assists requirements engineering to discover reuse potential and to identify parts of the CCOTS which have to be flexible in future.

Title **Prioritization of complex COTS**

Authors Büyükekici B., Deifel B., Jacobi C., Sandner R
Published Proceedings of REFSQ'99, Heidelberg
Source Institut f¨ur Informatik
Date 08-Jun-98
URL http://www.forsoft.de/teilprojekte/a4/publications/Refsq99b.zip
Abstract We present a concept for a prioritization method for requirements of complex commercial off the shelf software (CCOTS). Based on a process model different roles are identified for the elicitation and the negotiation of requirements. To get a practical solution that is applicable in industry an instrument to prioritize was developed using the portfolio analysis.

Title **Requirements Engineering for complex COTS**

Authors Deifel B.
Published Proceedings of REFSQ'98, Pisa
Source Institut f¨ur Informatik
Date 08-Jun-98
URL http://www.forsoft.de/publikationen/Dei98a.zip
Abstract The development of commercial of the shelf software (COTS) is usually highly market-driven. That's why the requirements engineering process is affected by problems which are different from those of individual software, which are of main interest in current research. We describe the main problems during the early phases in the development of complex COTS (CCOTS) and present our approach to improve the current situation.

Title **The Cost of COTS**

Authors Nancy Talbert
Published IEEE Computer, Vol. 31, No. 6, pp. 46-52
Source University of York, UK
Date 01-Jun-98
URL http://computer.org/computer/co1998/r6046abs.htm
Abstract In this interview, safety-critical systems expert John McDermid explores the sources of risk and the extra analysis work they require. In some cases, he says, this extra effort may erode attractive initial development costs in critical applications. McDermid describes why an application's characteristics are the major influences on the choice of whether to choose COTS or custom. For stringent applications-those that demand high integrity, reliability, and availability- the cost of creating a suitable assurance or safety argument may be prohibitive, or even impossible if there is insufficient access to the COTS software's design rationale. On the other hand, applications that emphasize flexibility may find that real-time kernels, which change relatively little and have seen extensive use, may be more robust than bespoke solutions. Hard data that would clarify the trade-offs between custom versus COTS solutions is still not available. McDermid states that more experience is needed to determine the relative costs of each solution. The observations should be made of a long-term development cycle that includes multiple upgrades and maintenance problems. Meanwhile, he states, the best strategies for those contemplating COTS use are to identify and plan for both project and COTS-specific risks and look beyond the initial development cost to the lifetime support of the product. Those who fail to do so may end up paying more than the COTS solution is worth.

Title **Certifying Off-the-Shelf Software Components**

Authors Jeffrey Voas
Published IEEE Computer, Vol. 31, No. 6, pp. 53-59
Source Reliable Software Technologies
Date 01-Jun-98
URL http://dlib.computer.org/co/books/co1998/pdf/r6053.pdf
Abstract Off-the-shelf components could save the software industry considerable time and money. However, the industry first needs a set of black-box processes to certify the suitability of COTS components.

Title **The Challenges of Using COTS Software in Component-Based Development**

Authors Jeffrey Voas
Published IEEE Computer, Vol. 31, No. 6, pp. 44-45
Source Reliable Software Technologies
Date 01-Jun-98
URL http://computer.org/computer/co1998/r6044abs.htm
Abstract An increasing number of organizations are using software applications of larger applications. In this new role, acquired software must integrate with other software functionality. In the introduction to the cover features, the author describes why the industry is moving toward a software design paradigm in which many of the needed software functions already exist. The developer's task, then, becomes one of accurately selecting functions and integrating them into a system. The problem is that commercial, off-the- shelf (COTS) software is almost always delivered in a black box with restrictions that keep developers from looking inside. Therefore, most forms of software analysis that would help developers decide if the software is going to perform safely, securely, and reliably are not available. Developers are thus at the mercy of the software vendor in many ways. The author argues that to achieve the goal of widespread component-based engineering, the industry must overcome challenges related to safety, reliability, and security. If the industry cannot adequately address these problems, the goal may remain unmet.

Title **Defensive Approaches to Testing Systems that Contain COTS and Third-Party**

Authors Jeffrey Voas
Published Proc. of 15th Int'l. Conference and Exposition on Testing Computer Software
Source Reliable Software Technologies

Date 01-Jun-98

URL ftp://ftp.rstcorp.com/pub/papers/ictcs98.ps

Abstract Most systems today are composed of hardware components, COTS software, and custom software. When a system fails, a confusing and complex liability problem ensues for all parties that have contributed software and hardware functionality to the composite system. This paper presents a consumer-oriented methodology for predicting what impact on system quality a particular Commercial-Off-The-Shelf (COTS) software component will have. When the result computed by the custom software causes a system failure, it becomes necessary to track down why that result occurred. If it is because of a logical defect in the custom software, then the vendor of the custom software is liable. If the result occurred because of a failure of a COTS software component (upon which the custom software was dependent for information), then the COTS vendor should be liable. Regardless of how these events might get argued in a court case and who would prevail, those persons responsible for integrating custom and COTS software together should take proactive steps to ensure that all safeguards against COTS software failures have been taken. That is clearly their best legal defense strategy. This paper presents methods that provide those safeguards.

Title **Wrapping Legacy Components**

Authors Robert Seacord

Published Software Engineering Seminar

Source SEI

Date 06-May-98

URL http://wwwsel.iit.nrc.ca/projects/cots/seminar/feb98slides/Robert/

Abstract Presentation

Title **CHANGING THE CULTURE (COTS VS DEVELOPMENT)**

Authors Colonel M.E. Hanrahan

Published Software Engineering Seminar

Source Canada Department of National Defence

Date 06-May-98

URL http://wwwsel.iit.nrc.ca/projects/cots/seminar/feb98slides/Mike/

Abstract Presentation

Title **(C)OTS From Models to Implementations**

Authors Don Cowan

Published Software Engineering Seminar

Source Computer Systems Group

Date 06-May-98

URL http://wwwsel.iit.nrc.ca/projects/cots/seminar/feb98slides/Don/

Abstract Presentation

Title **Maintenance, Support, ... Insurance, What Your 20% Buys**

Authors Barry Sullivan

Published Software Engineering Seminar

Source Gallium Software

Date 06-May-98

URL http://wwwsel.iit.nrc.ca/projects/cots/seminar/feb98slides/Barry/

Abstract Presentation

Title **COTS Software Evaluation Issues**

Authors John C. Dean

Published Software Engineering Seminar

Source IIT Software Engineering Group

Date 06-May-98

URL http://wwwsel.iit.nrc.ca/projects/cots/seminar/feb98slides/John/

Abstract Presentation

Title **Architecture of COTS Based Systems**

Authors Mark Vigder

Published Software Engineering Seminar

Source IIT Software Engineering Group

Date 06-May-98

URL http://wwwsel.iit.nrc.ca/projects/cots/seminar/feb98slides/Mark/

Abstract Presentation

Title **Building Long-Lived Systems from COTS Components**

Authors W. Morven Gentleman
Published Software Engineering Seminar
Source IIT Software Engineering Group
Date 06-May-98
URL http://wwwsel.iit.nrc.ca/projects/cots/seminar/feb98slides/Morven/
Abstract Presentation

Title **Middleware and the Integrated Diagnostic System (IDS)**

Authors Bob Orchard
Published Software Engineering Seminar
Source National Research Council Canada
Date 06-May-98
URL http://wwwsel.iit.nrc.ca/projects/cots/seminar/feb98slides/Bob/
Abstract Presentation

Title **Key Paradigms Shifts of COTS**

Authors Patricia Oberndorf
Published Software Engineering Seminar
Source SEI
Date 06-May-98
URL http://wwwsel.iit.nrc.ca/projects/cots/seminar/feb98slides/Tricia/
Abstract Presentation

Title **A proposal of an Internet-based software development process model for**

Authors Chiaki Hirai, Nobuo Saeki, Toshihiko Nakano
Published ICSE 98 Workshop on Software Engineering over the Internet
Source Hitachi
Date 25-Apr-98
URL http://sern.cpsc.ucalgary.ca/~maurer/ICSE98WS/Submissions/Hirai/hirai.html
Abstract Developing software systems from Commercial Off-The Shelf (COTS) components is becoming a mainstream method to achieve cost-effective software development. However, in the actual software projects it is often seen that a project is delayed by bugs in COTS products or otherwise beset by problems because of unsuitable conventional process models. We propose a new process model that places an emphasis on gathering bug information from Internet information sources. A software development environment based on this model is also proposed. This environment has sensors to observe COTS vendors, control rules to decide how a software process should be controlled and a process controller to control the software process based on the observation and the rules. We are now constructing this new environment, with some part of it having been used by developers. In this paper, we describe the current status of our approach.

Title **Army COTS Evaluation Workshop**

Authors
Published
Source Army
Date 13-Apr-98
URL http://doim.army.mil/cots/
Abstract On 13-17 April 1998, ODISC4 conducted a COTS Evaluation Workshop in Fairfax, Virginia. The purposes were to validate newly developed processes and procedures for evaluating COTS software products and to evaluate COTS imaging/capture software for recommendation to Army MAJCOMs and installations. Capture software was selected based upon a stated need at the Central Issue Facilities (CIF). The Directorates of Information Management (DOIM) at installations and MAJCOMs were surveyed, and the majority of the respondents agreed with this selection.

Title **A Defensive Approach to Testing Systems that Contain COTS and Third-Party**

Authors Jeffrey Voas
Published Proceedings AQUIS '98
Source Reliable Software Technologies
Date 01-Apr-98
URL ftp://ftp.rstcorp.com/pub/papers/ven.ps

Abstract The adage, ``if you want something done right, do it yourself'' is less of an option for software developers today than it was years ago. Today's software systems are complex ``systems of systems'', and developers must accept the fact that substantial portions of these composite systems will be provided by other developers. Losing control over every aspect of a system's functionality may worry the parties that are legally liable for the quality of the complete system. Those parties need assurance that each component will tolerate each other. 1 Software reuse has the potential to massively increase the rate at which information systems are built while reducing the costs of building these systems. Software reuse generally occurs in one of two ways: (1) purchasing Commercial-Off-The-Shelf (COTS) ``generic'' software, and (2) reusing one's own software modules from project to project through shared libraries. 2 But each of these methods run the risk that the complete system will suffer from problems caused by the reused or acquired software. This paper presents a methodology for predicting whether this is likely to occur (as well as presenting approaches for reducing this likelihood).

Title **COTS: The Economical Choice?**

Authors Jeffrey Voas
Published IEEE Software (Manager Column)
Source Reliable Software Technologies
Date 01-Mar-98
URL ftp://ftp.rstcorp.com/pub/papers/ieee_sw_manager.ps
Abstract Thirty years ago, there was not much interest in software from ``Joe Citizen.'' The personal computer did not exist, and most people had never even heard the term. Today, software is given as gifts, just as neckties and stereos. The amount of software that can be purchased Off-The-Shelf (OTS) is growing daily. Our appetite for it appears unbounded. There is a different side to software commerce that is just emerging---the offering of generic software components that contain fixed functionality. These software components can be leveraged by other systems still under development, i.e., the developing system will be bundled with the generic components as a single functional entity. This emerging marketplace is a trading forum between software developers and is similar in nature to a baseball card trading show. These generic software packages are termed Commercial-Off-The-Shelf (COTS) components. Their role is to enable new software systems to reach consumers faster and cheaper. Being last-to-market spells sudden death in the software industry, and any gimmick that carves days or weeks from the development schedule decreases this possibility. So today's systems are mainly hybrid architectures, where part of the complete system is bespoke (custom-made) and part is COTS. What portion is bespoke and what portion is COTS is application specific, however it is more than likely that the system is not 100% COTS. Communication occurs between the bespoke and COTS parts, and, as we will discuss later, the information in the messages transferred between the two sides will ultimately decide the quality of the composite.

Title **Acquiring COTS Software Selection Requirements**

Authors Neil A. M. Maiden, Cornelius Ncube
Published IEEE Software, Vol. 15, No. 2, pp. 46-56
Source School of Informatics City University London
Date 01-Mar-98
URL http://computer.org/software/so1998/s2046abs.htm
Abstract The authors describe their design of Procurement Oriented Requirements Engineering, using knowledge gained from past studies of real-world requirements acquisition for complex product selection. PORE is built from existing requirements engineering and knowledge engineering techniques, feature analysis, multicriteria decision making, argumentation, and template-based approaches. The authors recount their experiences applying PORE to help a UK Ministry of Defense team devise requirements for a new naval platform. They report on 11 problems encountered during the project and how their solutions to them will be incorporated in future versions of PORE.

Title **Calibration Results of COCOMOII.1997**

Authors Barry Boehm, Brad Clark, Sunita Devnani-Chulani
Published SEPG-98
Source USC-Center for Software Engineering
Date 01-Mar-98
URL http://sunset.usc.edu/TechRpts/Papers/usccse97-507/usccse97-507.pdf
Abstract COCOMO II is an effort to update software cost estimation models, such as the 1981 COnstructive COst MOdel and its 1987 Ada COCOMO update. Both these and other 1980's cost models have experienced difficulties in estimating software projects of the 90s due to new practices such as non-sequential and rapid-development process models; reuse-driven approaches involving commercial-off-the-shelf (COTS) packages, reengineering, applications composition, and application generation capabilities; object-oriented approaches supported by distributed middleware; software process maturity effects and process-driven quality estimation. The COCOMO II research effort has developed new functional forms reflecting these practices, and is concentrated on developing a model well-suited for the 1990s and then annually updating it for the forthcoming years of the 21st Century. The current COCOMO II.1997 has been calibrated to a dataset of 83 projects from a mix of Commercial, Aerospace, Government, and FFRDC organizations. The estimates of the 1997 calibrated model are within 30% of the actuals 52% of the times before stratification by organization; and within 30% of the actuals 64% of the times after stratification by organization. The 1997 calibration results indicated that the following changes from COCOMO '81 to COCOMO II were successfully explaining sources of variation in the project data : Replacing the COCOMO '81 Development Modes by the 5 exponent drivers Precedentedness, Development Flexibility, Architecture/Risk Resolution, Team Cohesiveness, and CMM-based Process Maturity. Adding multiplicative cost drivers for Amount of Documentation

and Multisite Development.

Title **USE OF COTS TECHNOLOGY IN C2 INFORMATION SYSTEMS: BALANCING**

Authors Iain Macleod

Published Journal of Battlefield Technology, Volume 1 Number 1

Source

Date 01-Mar-98

URL

Abstract The overlap in requirements of military and commercial information systems is steadily growing. Wider use of Commercial-Off-The-Shelf (COTS) information technology in military systems offers the prospect of reduced development and support costs, improved interoperability, reduced technological risk, accelerated deployment, and incremental system evolution. On the other hand, COTS products are effectively "black boxes" and are usually not of military grade, raising significant security and reliability concerns if they are used in critical Command and Control (C2) information systems. Management difficulties can also arise as a consequence of frequent product revisions, immaturity of released products and vendor "lock in". In the search for affordable leading-edge capability, military forces are seeking to take advantage of commercial technology wherever possible. This paper examines potential benefits and risks associated with use of COTS technology in C2 information systems and outlines a number of risk mitigation strategies.

Title **Evaluating and Sharing Year 2000 COTS Compliance Information**

Authors Robert Martin

Published ITW/AA NETWORK - SNDC2 SPO Year 2000 Working Group Meeting

Source MITRE

Date 01-Jan-98

URL http://www.mitre.org/research/y2k/briefings/evaluating.pdf

Abstract This presentation focuses on dealing with commercial software for the Year 2000 and the various issues it presents and some suggestions on how to handle them. An earlier version of this presentation was given at the Joint Staff Year 2000 Working Group meeting in November 1997.

Title **Composing Components: How Does One Detect Potential Architectural**

Authors Cristina Gacek, Barry Boehm

Published Proceedings of the OMG-DARPA-MCC Workshop on Compositional Software Architectures

Source USC-Center for Software Engineering

Date 01-Jan-98

URL http://sunset.usc.edu/TechRpts/Papers/usccse98-505.html

Abstract Nowadays, in order to be competitive, a developer's usage of Commercial off the Shelf (COTS), or Government off the Shelf (GOTS), packages has become a sine qua non, at times being an explicit requirement from the customer. The idea of simply plugging together various COTS packages and/or other existing parts results from the megaprogramming principles [Boehm and Scherlis 1992]. What people tend to trivialize is the side effects resulting from the plugging or composition of these subsystems. Some COTS vendors tend to preach that because their tool follows a specific standard, say CORBA, all composition problems disappear. Well, it actually is not that simple. Side effects resulting from the composition of subsystems are not just the result of different assumptions in communication methods by various subsystems, but the result from differences in various sorts of assumptions, such as the number of threads that are to execute concurrently, or even on the load imposed on certain resources. This problem is referred to as architectural mismatches [Garlan et al. 1995] [Abd-Allah 1996]. Some but not all of these architectural mismatches can be detected via domain architecture characteristics, such as mismatches in additional domain interface types (units, coordinate systems, frequencies), going beyond the general interface types in standards such as CORBA. Other researchers have successfully approached reuse at the architectural level by limiting their assets not by domain, but rather by dealing with a specific architectural style. I.e., they support reuse based on limitations on the architectural characteristics of the various parts and resulting systems [Medvidovic et al. 1997] [Magee and Kramer 1996] [Allan and Garlan 1996]. This approach can be successful because it simply avoids the occurrence of architectural mismatches. Our work addresses the importance of underlying architectural features in determining potential architectural mismatches while composing arbitrary components. We have devised a set of those features, which we call conceptual features [Abd-Allah 1996][Gacek 1997], and are building a model that uses them for detecting potential architectural mismatches. This underlying model has been built using Z [Spivey 1992].

Title **Lessons Learned During Requirements Acquisition for COTS Systems**

Authors Neil A. M. Maiden, Cornelius Ncube, Andrew Moore

Published Communications Of the ACM, vol. 40, No 12, pp 21-25

Source School of Informatics City University London

Date 01-Dec-97

URL

Abstract

Title **An Architectural Approach to Building Systems from COTS Software Components**

Authors Dr. Mark R. Vigder and John Dean
Published Proceedings of the 1997 Center for Advanced Studies Conference (CASCON 97)
Source IIT Software Engineering Group
Date 13-Nov-97
URL http://wwwsel.iit.nrc.ca/abstracts/NRC40221.abs
Abstract As software systems become increasingly complex to build developers are turning more and more to integrating pre-built components from third party developers into their systems. This use of Commercial Off-The-Shelf (COTS) software components in system construction presents new challenges to system architects and designers. This paper is an experience report that describes issues raised when integrating COTS components, outlines strategies for integration, and presents some informal rules we have developed that ease the development and maintenance of such systems.

Title **Comparing Operating Systems using Robustness Benchmarks**

Authors Nathan P. Kropp, Philip J. Koopman, Daniel P. Siewiorek, Christopher Dingman, Ted Marz
Published 16th IEEE Symposium on Reliable Distributed Systems
Source CMU-Institute for Complex Engineered Systems, SEI
Date 22-Oct-97
URL http://www.cs.cmu.edu/afs/cs/project/edrc-ballista/www/srds97/index.html
Abstract When creating mission-critical distributed systems using off-the-shelf components, it is important to assess the dependability of not only the hardware, but the software as well. This paper proposes a way to test operating system dependability. The concept of response regions is presented as a way to visualize erroneous system behavior and gain insight into failure mechanisms. A 5-point CRASH scale is defined for grading the severity of robustness vulnerabilities encountered. Test results from five operating systems are analyzed for robustness vulnerabilities, and exhibit a range of dependability. Robustness benchmarking comparisons of this type may provide important information to both users and designers of off-the-shelf software

Title **Maintenance of COTS-Intensive Software Systems**

Authors Duane Hybertson, Anh Ta, William Thomas
Published Software Maintenance: Research and Practice, Vol. 9, pp. 203-216.
Source MITRE
Date 01-Aug-97
URL
Abstract The software industry has made extensive use of commercial software tools such as compilers and editors in development environments of computer-based systems for several decades. However, in recent years an emerging trend is the extensive usage of commercial off-the-shelf (COTS) software products as a major part of delivered software systems. It is generally recognized that this trend introduces a significant change to the development of computer-based systems. The thesis of this paper is that this trend also introduces a significant change to the software maintenance process. The paper first provides a context for maintenance of COTS software, by describing the traditional software maintenance process and the development of COTS-intensive systems. Some of the issues involved in the maintenance of COTS-intensive software systems and reasons why the COTS factor constitutes a significant change are then presented. Finally, some suggestions are made for addressing the issues in the maintenance of COTS-intensive systems.

Title **Evolutionary Rapid Development**

Authors
Published
Source Software Productivity Consortium
Date 01-Jun-97
URL http://www.software.org/pub/darpa/erd/erdpv010004.html
Abstract The Consortium has published a report describing the Evolutionary Rapid Development (ERD) process. ERD is an architectural and development approach which leverages commercial off-the-shelf (COTS) software components, Internet and Web-based information artifacts, and flexible architectures to manage the development of complex information systems in an environment of rapidly evolving components and architectures. As a COTS-based approach to rapid development, featuring small teams of highly experienced developers and significant user participation, the Evolutionary Rapid Development process can help members meet the increasing demands for COTS-based information systems.

Title **A Software Development Process for COTS-based Information System**

Authors Greg Fox from TRW, Karen Lantner from EDS, Steven Marcom from TRW
Published 5th International Symposium on Assessment of Software Tools
Source TRW
Date 01-Jun-97
URL http://stsc.hill.af.mil/crosstalk/1998/mar/development.asp

Abstract Modern software developers are guided by a variety of formal and informal processes that organize and control development activities across large groups of developers or multiple organizations and supply discipline and order lacking in many early development efforts. The available inventory of documented process methods is limited: Most process methods assume the system being built will be coded largely from scratch. The processes do not address many of the challenges associated with building systems that contain large amounts of commercial off-the-shelf (COTS) software. The Infrastructure Incremental Development Approach (IIDA) is a combination of the classical development model and the spiral process model to accommodate the needs of COTS-based technical infrastructure development.

Title **COTS Software Integration Cost Modeling Study**

Authors Christopher Abts, Barry W. Boehm
Published USAF Electronic Systems Center, Hanscom AFB
Source USC-Center for Software Engineering
Date 01-Jun-97
URL http://sunset.usc.edu/COCOTS/docs/USAFReport.pdf
Abstract This study represents a first effort towards the goal of developing a comprehensive COTS integration cost modeling tool. The approach taken was to first examine a wide variety of sources in an attempt to identify the most significant factors driving COTS integration costs, and to develop a mathematical form for such a model. These sources ranged from already existing cost models to information gathered in a preliminary high level data collection survey. Once the form and candidate drivers had been identified, the next step was to gather project level COTS integration effort data in a second round data collection exercise. This project level data was then used to calibrate and validate the proposed model. Data from both a graduate level software engineering class and from industrial sources were used in calibration attempts. The industrial data proved problematic, however, so for the purposes of this study, the final calibration of the model was based upon the student projects. The final result was a cost model following the general form of the well-known COCOMO software cost estimation model, but with an alternate set of cost drivers. The scope of the model is also narrow, addressing only initial integration coding costs. The predictive power of the model at this stage is only fair, but it was demonstrated that with appropriate data, the accuracy of the model could be greatly improved. Finally, the richness to the problem of capturing all significant costs associated with using COTS software offers many worth-while directions in which to expand the scope of this model.

Title **Effective Use of COTS (Commercial-Off-the-Shelf) Software Components in Long**

Authors W. Morven Gentleman
Published ICSE 97 Tutorial 2C
Source IIT Software Engineering Group
Date 17-May-97
URL
Abstract This tutorial looks at kinds of COTS software components that can be used in long lived systems, and the technology available for building around them. The potential benefits and risks of this approach to systems are examined. Modifications of conventional development processes are required to focus on where time and cost expenditures occur, and where risks arise

Title **System Implementation Using Commercial Off-The-Shelf (COTS) Software**

Authors John C. Dean, CD and Dr. Mark R. Vigder
Published Proceedings of the 1997 Software Technology Conference(STC '97)
Source IIT Software Engineering Group
Date 03-May-97
URL http://wwwsel.iit.nrc.ca/abstracts/NRC40173.abs
Abstract In an attempt to reduce cost and delivery time there is an increasing effort to build effective software systems from commercial off-the-shelf (COTS) software components. Typically the source code for these components is not available to the system developer nor does the system developer control the specification, release schedule and evolution of the components. In order to better understand the issues involved in system implementation using COTS software, we are undertaking a series of experiments concerning systems which use COTS software components. Our purpose is to experiment with architectures, technologies, and processes in order to better understand the issues relative to system users and developers (as opposed to developers of the COTS software components).This paper describes preliminary results on building a distributed Photographic Document Transfer system. This system represents the need to integrate a significant number of COTS software products under one umbrella system, including data acquisition, data conversion, data manipulation, communication, database, and messaging. Many of these functions are provided by COTS software components from different vendors. A prototype incorporating some of these components is being developed.

Title **COTS Inclusion in the DII COE**

Authors
Published
Source DISA
Date 15-Jan-97

URL http://coeeng.ncr.disa.mil/REFERENCE_PAGES/JCSCOT/JCSCOT.HTM

Abstract Purpose. This paper discusses considerations surrounding the inclusion of commercial, off-the-shelf (COTS) products in the Defense Information Infrastructure (DII) Common Operating Environment (COE). Executive Summary. The DII COE will make maximum use of COTS, particularly in those areas of the COE most widely used across the DII subscriber community. COTS bring their own set of integration and interoperability issues with them, however, the evaluation process associated with inclusion of COTS products should either minimize interoperability issues or identify up-front the costs associated with achieving interoperability to the Department based on inclusion of a particular product.

Title **Portability and Supportability of COTS Applications**

Authors Bruce Eidsvik
Published COTS IN CANADA Conference
Source Array Systems Computing
Date 14-Nov-96
URL http://www.mc.com/COTS_folder/COTS-Canada/Eidsvik.pdf
Abstract Presentation on COTS software in Canada.

Title **COTS AND THE WARFIGHTER**

Authors Paul D. Manson
Published COTS IN CANADA Conference
Source Lockheed Martin Canada
Date 14-Nov-96
URL http://www.mc.com/COTS_folder/COTS-Canada/Manson.html
Abstract Keynote Address at the COTS IN CANADA Conference. Here in Canada the emergence of the COTS phenomenon was particularly iconoclastic, given the fact that our armed forces, being rather small and specialized, had grown accustomed over the years to the rigid application of carefully derived operational requirements, to the extent that these could in most cases be satisfied only by specially designed equipment or by the extensive “canadianization” of systems that had been designed for other armed forces.

Title **TEAM SUBMARINE STRATEGY 2000: COTS Acquisition Primer**

Authors
Published
Source
Date 12-Nov-96
URL http://pats.crane.navy.mil/pubdoc/cotsacqu.doc
Abstract The purpose of this document is to describe the processes that will be used to develop and execute a comprehensive and cost-effective Commercial Off The Shelf (COTS)-based submarine Non-Propulsion Electronics (NPE) systems acquisition and support strategy. It should be noted, however, that many of the techniques described can also be applied to the selection of hull, mechanical and electrical products. This document will address options available to the Program Manager (PM) for cost effective selection and application of COTS products based on DoD policy, industry experience, and program lessons learned. Team Submarine, comprised of all United States Navy submarine electronic system acquisition program offices, recognizes that the acquisition and insertion of COTS technology is vital to lowering total ownership costs while improving submarine performance and fully supports this strategy for COTS implementation and support.

Title **USC-CSE Focused Workshop #7: System Integration with Commercial Software**

Authors Christopher Abts, Barry Boehm
Published USAF Electronic Systems Center, Hanscom AFB
Source USC-Center for Software Engineering
Date 01-Nov-96
URL http://sunset.usc.edu/COCOTS/docs/Nov96_COTS/Nov96_COTS.html
Abstract Although the workshop concentrated on COTS software integration risks and ways to improve COTS integration, this was done in a context that COTS usage is generally a good thing. For most applications, deciding not to use COTS is simply unrealistic. But the road to successful COTS integration has many risks and pitfalls. In this context, I'd like to summarize the workshop results in terms of: 1) Four characteristics of COTS integration which I found helpful in explaining the various pitfalls and recommendations highlighted by the workshop participants (several of the points are adapted from a particularly good Lockheed Martin briefing included in appendix A of these proceedings, "COTS Integration: Application Lessons Learned," which is well worth your study). 2) The primary research priorities identified by the participants, and USC-CSE's plans for addressing them. 3) A set of corporate-level COTS integration issues emerging from the workshop which we plan to address at the upcoming USC-CSE Executive Workshop for Affiliates on March 12, 1997.

Title **Business Reengineering for Information Technology: From Business Process to**

Authors Shawn A. Bohner, Clement L. McGowan, and Mary M. Harlow
Published
Source Mitre
Date 01-Jun-96
URL http://www.srv.net/~hheydt/ncose/busreeng.htm
Abstract The bridge between business process and information systems reengineering is all too often missing from the roadmap of reengineering efforts. When process and system engineers get to this transition, they discover a rickety old bridge with steep terrain on either side of a wide chasm. Recognizing this dilemma, we developed the Business Reengineering for Information Technology (BRIT) approach that systematically transitions from business process to information systems engineering. BRIT is designed to handle a wide range of reengineering factors including: "best practices," COTS applications, non- standard business processes, and change situations ranging from continuous improvement to radical restructuring. This proven approach is described here with relevant examples of its applications.

Title **Integrating Through User Interface: A Flexible Integration Framework for**

Authors Yimin Bao, Ellis Horowitz
Published Proceedings of COMPSAC '96
Source USC-Center for Software Engineering
Date 01-Apr-96
URL
Abstract Current trend of constructing new systems from collections of pre-existing third-party software and the commercial off-the-shelf (COTS) products presents serious challenges to existing integration technology. In this paper we present a flexible integration framework which has general applicability for pre-existing third-party and COTS software (often highly interactive, with graphical user interface, and without source code access), supports users to easily change the way software interact with each other (thus supporting system evolution and component reusability), and is easily programmable by the end-users. Specifically we describe Tool Integration Language (TIL) and Tool Integration Server System (TISS) which provide flexible integration mechanisms for our framework and show how they can be used to integrated a set of existing applications and COTS together.

Title **Software Engineering and ICSE Futures**

Authors Barry Boehm
Published 18th ICSE
Source USC-Center for Software Engineering
Date 27-Mar-96
URL http://www.cs.tu-berlin.de/cs/events/1996/ICSE-18/v2n3/v2n3-11.html
Abstract The next century may well be called "the software century". Organizations competing in product lines, services, or national defense will find that the excellence of their software engineering efforts will be one of their most critical success factors. Meanwhile, the twin paradigm shifts of COTS (commercial-off-the-shelf) software and cyberspace are shaking traditional software engineering methods to their roots. COTS software is causing a 180 degree shift in the traditional software development cycle: from requirements-determining-capabilities to capabilities-determining-requirements. Cyberspace is changing the nature of software applications, and their development, from individual-oriented activities to networked-group

Title **COTS Software Integration: State of the Art**

Authors Mark R. Vigder, W. Morven Gentleman, John C. Dean
Published
Source IIT Software Engineering Group
Date 15-Jan-96
URL http://wwwsel.iit.nrc.ca/abstracts/NRC39198.abs
Abstract This paper outlines the current state of the art in using Commercial Off-the-shelf (COTS) software to build systems, and identifies the issues which must be resolved in order to make the use of COTS software components a cost-effective solution to system development and support. The paper is based on interviews and discussions with organizations that are users of COTS components (rather than organizations that are builders of COTS components). The technologies and methods for integrating COTS software are described; and the problems encountered during development and maintenance are identified. The main issues in COTS usage are identified and provide a direction for further research.

Title **Integrated Monitoring, Analysis, and Control COTS System (IMACCS)**

Authors
Published
Source NASA/GSFC
Date 15-Jan-96

URL http://joy.gsfc.nasa.gov/RenTeam/Doc_in_PDF/IMACCS_Report.pdf

Abstract This report describes a successfully functioning, commercial off-the-shelf (COTS)-based ground support system called the Integrated Monitoring, Analysis, and Control COTS System (IMACCS). IMACCS was implemented as a prototype by Goddard Space Flight Center's Mission Operations and Data Systems Directorate (MO&DSD) to operate NASA's Solar Anomalous and Magnetospheric Particle Explorer. IMACCS was conceived specifically to build on previous experience in test bed evaluation of COTS products. The IMACCS project was to integrate a typical set of such tools, connect them to live tracking and telemetry data from a real on-orbit satellite, and perform shadow mission operations. The IMACCS project was to assess the completeness, robustness, and performance of a COTS-based ground system. As an additional constraint, IMACCS had to be implemented within 90 days of project approval. This report discusses the challenges that led to the IMACCS project, the processes used for implementing IMACCS, how these processes fit within MO&DSD's reengineered ground systems development processes, and the results obtained by comparing IMACCS requirements, operations, costs, and implementation process against the currently operating ground system.

Title **Using Commercial-Off-the-Shelf (COTS) Software in High-Consequence Safety**

Authors J. A. Scott, G. G. Preckshot, J. M. Gallagher

Published

Source Lawrence Livermore National Laboratory

Date 10-Nov-95

URL http://nssc.llnl.gov/FESSP/CSRC/122246.pdf

Abstract This paper is based on work performed by Lawrence Livermore National Laboratory 1 to assist the U.S. Nuclear Regulatory Commission in understanding the state of the art with respect to applying commercial off-the-shelf (COTS) software to high-consequence safety systems. These systems, for which the consequences of failure can be severe or catastrophic, must be developed, implemented, and maintained in ways that provide assurance that catastrophic consequences will be prevented. This paper discusses various aspects of the question of using commercially available software in these systems. Risk, grading, and system assessment are discussed, and relevant standards are summarized. A recommendation for addressing key issues regarding the use of commercial software in high-consequence safety systems is given.

Title **Architecture and Design of Storage and Data Management for the NASA Earth**

Authors Ben Kobler, John Berbert, Parris Caulk, P.C. Hariharan

Published 14th IEEE Symposium on Mass Storage Systems

Source NASA/GSFC

Date 01-Jul-95

URL http://computer.org/conferen/MSS95/KOBLER/KOBLER.HTM

Abstract Mission to Planet Earth (MTPE) is a long-term NASA research mission to study the processes leading to global climate change. The EOS Data and Information System (EOSDIS) is the component within MTPE that will provide the Earth science community with easy, affordable, and reliable access to Earth science data. EOSDIS is a distributed system, with major facilities at eight Distributed Active Archive Centers (DAACs) located throughout the United States. At the DAACs the Science Data Processing Segment (SDPS) will receive, process, archive, and manage all data. It is estimated that several hundred gigaflops of processing power will be required to process and archive the several terabytes of new data that will be generated and distributed daily. Thousands of science users and perhaps several hundred thousand nonscience users will access the system.

Title **Cost Models for Future Software Life Cycle Processes: COCOMO 2.0**

Authors Barry Boehm, Bradford Clark, USC-Center for Software Engineering ; Ellis Horowitz , Chris Westland,

Published Annals of Software Engineering Special Volume on Software Process and Product Measurement, J.D.

Source USC-Center for Software Engineering

Date 01-Jun-95

URL http://sunset.usc.edu/TechRpts/Papers/usccse95-508/usccse95-508.pdf

Abstract Current software cost estimation models, such as the 1981 Constructive Cost Model (COCOMO) for software cost estimation and its 1987 Ada COCOMO update, have been experiencing increasing difficulties in estimating the costs of software developed to new life cycle processes and capabilities. These include non-sequential and rapid-development process models; reuse-driven approaches involving commercial off the shelf (COTS) packages, reengineering, applications composition, and applications generation capabilities; object-oriented approaches supported by distributed middleware; and software process maturity initiatives. This paper summarizes research in deriving a baseline COCOMO 2.0 model tailored to these new forms of software development, including rationales for the model decisions. The major new modeling capabilities of COCOMO 2.0 are a tailorable family of software sizing models, involving Object Points, Function Points, and Source Lines of Code; nonlinear models for software reuse and reengineering; an exponent-driver approach for modeling relative software diseconomies of scale; and several additions, deletions, and updates to previous COCOMO effort-multiplier cost drivers. This model is serving as a framework for an extensive current data collection and analysis effort to further refine and calibrate the model's estimation capabilities.

Title **Implications of COTS for the Defense Industry**

Authors Dr. John Kreick
Published COTS 95
Source Sanders, a Lockheed Martin Company
Date 01-Jun-95
URL http://www.mc.com/COTS_folder/cots95/cots95_kreick.html
Abstract Speech at COTS 95. I think it's been an interesting session today. I've enjoyed the opportunity to hear from our senior DOD officials. I've been asked to provide a perspective from an industry viewpoint. Thought it might be of interest to look at COTS as it relates to the various aspects of systems developments as we see them. In doing this, I want to use some brief and specific examples from our experience at Sanders, since Bill Perry set the course of this new way of doing business. I'll discuss COTS from our perspective of system development touching on five areas: system engineering, detail design, fabrication, integration and test, and product support.

Title **COTS 95 Conference Summary**

Authors Robert Costello
Published COTS 95
Source DoD
Date 01-Jun-95
URL http://www.mc.com/COTS_folder/cots95/cots95_summary.html
Abstract Speech at COTS 95. What I heard first of all, this is a unique time for change. Everybody understands that there's a unique time for change, and we better take advantage of it. But there is problem. I can't live with COTS and I'm going to die without COTS, and we need some very sharp people looking at the critical issues that have been brought up today. What do we do with R&D? How do we ensure that the money we save over here goes to the R&D that we need to get the job done? How do we get competitive forces working together with these few R&D dollars to compound our capability to solve the problems we're working on? And I'll go back and I'll use a different program.

Title **Will the Adoption of COTS Help the Military Stay Abreast of Changes in**

Authors Panel Discussion
Published COTS 95
Source DoD
Date 01-Jun-95
URL http://www.mc.com/COTS_folder/cots95/cots95_panel1.html
Abstract Panel discussion at COTS 95. Includes some discussion of COTS software.

Title **COTS in the Air Force - Success Story**

Authors Lloyd Mosemann - Deputy Assistant Secretary of the Air Force
Published COTS 95
Source DoD
Date 01-Jun-95
URL http://www.mc.com/COTS_folder/cots95/cots95_mosemann.html
Abstract Speech at COTS 95. It's a privilege to be here and to tell you a little bit about what the Air Force experience has been into &emdash; and probably will be in the future &emdash; with respect to the use of COTS. And then I'd like to close my remarks by telling you that I think there are some problems ahead as well that we foresee as we become more dependent on COTS

Title **Why COTS is Vital to the Modern Military**

Authors ADMIRAL WILLIAM OWENS - Vice Chairman of the Joint Chiefs of Staff
Published COTS 95
Source DoD
Date 01-Jun-95
URL http://www.mc.com/COTS_folder/cots95/cots95_owens.html
Abstract Speech at COTS 95. I've been giving some speeches, talking about the four revolutions that face our military today. And I'd like to just briefly run through those revolutions and why I think they're so enormously important, and why this is such a watershed time period for us in the U.S. military. And why I think COTS as an element of that is such an extremely important

Title **Advanced Technology for the Military at Lower Cost**

Authors Dwight Williams - Deputy Director, Defense Airborne Reconnaissance Office
Published COTS 95
Source DoD

Date 01-Jun-95

URL http://www.mc.com/COTS_folder/cots95/cots95_williams.html

Abstract Speech at COTS 95. Anyway, what I'd like to do today is talk to you about the Air Reconnaissance Office and the importance of commercial off-the-shelf technology in our concept for improving and modernizing the Air Reconnaissance environment.

Title **USC-CSE Focused Workshop #4: COCOMO 2.0 - COTS Software Integration**

Authors Ellis Horowitz, Christopher M. Abts

Published

Source USC-Center for Software Engineering

Date 01-May-95

URL http://sunset.usc.edu/COCOTS/docs/FW4_COTS_breakout_group_discussion.pdf

Abstract The discussion focused on five areas related to COTS issues. Initial efforts were devoted to reaching a consensus on what is actually meant by the term "COTS" software. This was followed by an exploration of scenarios in which COTS software might be employed. The group then examined how existing COTS products might be evaluated and selected for use. Discussion of issues related to the testing of COTS software followed next, with the final efforts of the group focusing on the challenge of reflecting COTS software in overall system development cost estimations.

Title **Applying COTS Products and Services to Major Defense Programs**

Authors

Published

Source Mercury Computer Systems

Date

URL http://www.mc.com/COTS_folder/cots_mtb/cots_mtb.html

Abstract The requirements for computer systems within those DoD programs in which Mercury Computer Systems is typically involved cannot can not be satisfied by Commercial Off-The-Shelf (COTS) computer technology alone. There are various requirements which are simply beyond the capability of the systems designed, built and sold in the commercial markets. DoD, in an attempt to meet the military cost reduction requirements of the Clinton administration, have begun stipulating COTS technology be used in all applicable applications. It is our contention that requiring COTS components by itself will not produce the desired magnitude of cost reductions nor improvement in "time-to-deployment." This paper sets forth an approach which begins with COTS, but goes further to address the issues of "best value" where COTS alone is insufficient. The motivations for requiring COTS is discussed along with examples of where COTS technology fails to meet DoD requirements. The paper articulates the "COTS+" approach - a technology architecture and design philosophy which should be required of any world class technology vendor committed to selling their products and services to DoD. By recognizing COTS+ as meriting high on the "best value" curve, DoD can affect change in the way commercial vendors and prime contractors respond to military system requirements in the future.

Title **The market for COTS or "Commercial-off-the-shelf" electronics for harsh**

Authors

Published

Source DY4

Date

URL http://www.dy4.com/cots/exec_sm.htm

Abstract The market for COTS or "Commercial-off-the-shelf" electronics for harsh environments is growing and systems integrators want to know how they can take advantage of the products offered by vendors. To address this need, the COTS Handbook brings together information on the subject of rugged COTS electronics products, from hardware to software, from design to deployment.

Title **From Avionics to Vetronics: Considerations for Application of COTS VME to**

Authors Duncan Young

Published

Source DY4

Date

URL http://www.dy4.com/cots/position.htm

Abstract Position paper that discusses mostly hardware (and some software) COTS issues. This paper defines Commercial Off-The-Shelf (COTS), distinguishes between technology and product, and identifies a three-stage process for evaluating COTS VME for deployed defense programs.

(This Page Intentionally Blank)

Initial Distribution

Headquarters Air Force

SAF/OS	Secretary of the Air Force
AF/CC	Chief of Staff
AF/CV	Vice Chief of Staff
AF/CVA	Assistant Vice Chief of Staff
AF/HO	Historian
AF/ST	Chief Scientist
AF/SC	Communications and Information
AF/SG	Surgeon General
AF/SF	Security Forces
AF/TE	Test and Evaluation

Assistant Secretary of the Air Force

SAF/AQ	Assistant Secretary for Acquisition
SAF/AQ	Military Director, USAF Scientific Advisory Board
SAF/AQ	Principal Deputy (Acquisition & Management)
AFPEO/AT	Airlift & Trainers
AFPEO/FB	Fighters & Bomber Program
AFPEO/LI	Logistics Information Systems
AFPEO/SP	Space
AFPEO/WP	Weapons
AFPEO/C2	Command and Control
SAF/AQC	Contracting
SAF/AQI	Information Dominance
SAF/AQL	Special Programs
SAF/AQP	Global Power
SAF/AQQ	Global Reach
SAF/AQR	Science, Technology and Engineering
SAF/AQS	Space and Nuclear Deterrence
SAF/AQX	Management Policy and Program Integration
SAF/MI	Assistant Secretary (Manpower, Reserve Affairs, Installations & Environment)
SAF/SN	Assistant Secretary (Space)
SAF/SX	Deputy Assistant Secretary (Space Plans and Policy)

Deputy Chief of Staff, Air and Space Operations

AF/XO	DCS, Air and Space Operations
AF/XOC	Command and Control
AF/XOI	Intelligence, Surveillance and Reconnaissance
AF/XOJ	Joint Matters
AF/XOO	Operations and Training
AF/XOR	Operational Requirements

Deputy Chief of Staff, Installations and Logistics

AF/IL	DCS, Installations and Logistics
AF/ILX	Plans and Integration

Initial Distribution (continued)

Deputy Chief of Staff, Plans and Programs

AF/XP	DCS, Plans and Programs
AF/XPI	Information and Systems
AF/XPM	Manpower, Organization and Quality
AF/XPP	Programs
AF/XPX	Strategic Planning
AF/XPY	Analysis

Deputy Chief of Staff, Personnel

AF/DP	DCS, Personnel

Office of the Secretary of Defense

USD (A&T)	Under Secretary for Acquisition and Technology
USD (A&T)/DSB	Defense Science Board
DAU	Defense Acquisition University
– DSMC	- Defense Systems Management College
DARPA	Defense Advanced Research Projects Agency
DISA	Defense Information Systems Agency
DIA	Defense Intelligence Agency
BMDO	Ballistic Missile Defense Office

Other Air Force Organizations

AFMC	Air Force Materiel Command
– CC	- Commander, Air Force Materiel Command
– EN	- Directorate of Engineering and Technical Management
– AFRL	- Air Force Research Laboratory
– SMC	- Space and Missile Systems Center
– ESC	- Electronic Systems Center
– ASC	- Aeronautics Systems Center
– HSC	- Human Systems Center
– AFOSR	- Air Force Office of Scientific Research
ACC	Air Combat Command
– CC	- Commander, Air Combat Command
– ASC2A	- Air and Space Command and Control Agency
AMC	Air Mobility Command
AFSPC	Air Force Space Command
PACAF	Pacific Air Forces
USAFE	U.S. Air Forces Europe
AETC	Air Education and Training Command
– AU	- Air University
AFOTEC	Air Force Test and Evaluation Center
AFSOC	Air Force Special Operations Command
AIA	Air Intelligence Agency
NAIC	National Air Intelligence Center
USAFA	U.S. Air Force Academy
NGB/CF	National Guard Bureau
AFSAA	Air Force Studies and Analysis Agency

Initial Distribution (continued)

U.S. Army

ASB	Army Science Board

U.S. Navy

NRAC	Naval Research Advisory Committee
Naval Studies Board	

U.S. Marine Corps

DC/S (A)	Deputy Chief of Staff for Aviation

Joint Staff

JCS	Office of the Vice Chairman
J2	Intelligence
J3	Operations
J4	Logistics
J5	Strategic Plans and Policies
J6	Command, Control, Communications & Computer Systems
J7	Operational Plans and Interoperability
J8	Force Structure, Resources and Assessment

Other

Study Participants
Aerospace Corporation
ANSER
MITRE
RAND
SEI
Study Interviewees

(This Page Intentionally Blank)

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave Blank*)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	April 2000	**Final, January 1999 - November 1999**

4. TITLE AND SUBTITLE

Ensuring Successful Implementation of Commercial Items in Air Force Systems

5. FUNDING NUMBERS

6. AUTHOR(S)

Mr. Jeffrey E. Grant, Chair, Dr. Robert Rankine Jr., Mr. Kenneth M. Brown, Mr. William R. Carter, Mr. John Foreman, Mr. Don Reifer, Dr. Will Tracz, Dr. Nick Tredennick

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(ES)

AF/SB
Pentagon
Washington, DC 20330-1180

8. PERFORMING ORGANIZATION REPORT NUMBER

SAB-TR-99-03

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

SAF/OS
AF/CC
Pentagon
Washington, DC 20330-1670

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Cleared for Open Publication

12b. DISTRIBUTION CODE

ABSTRACT (Maximum 200 Words)

Taking advantage of commercial off-the-shelf (COTS) products seems like a logical way to achieve significant cost savings. However, based on an assessment of 34 programs and organizations, only a few are realizing significant benefits. Most are struggling with its complexity and a few have failed miserably. The complexities are numerous and less than obvious. Arguably the biggest pitfall of all is inflexible requirements. A balance must be achieved between desired performance and what can be reasonably attained by integrating available and projected commercial products. In total, the COTS Study Panel observed about 25 common pitfalls that programs are experiencing. Most could be avoided or mitigated if appropriate processes or procedures were in place that people understood and followed. Requirements must flow into an architecture that can truly exploit the advantages of COTS. Contractors must shift from “design and build” unique products to “buy and integrate” standard products. Everyone coping with a COTS development is on a very steep learning curve and those that seem to do it well have been at it for many years. They freely admit that they have made every mistake imaginable along the way. Unfortunately, others can’t imagine the mistakes they are about to make.

14. SUBJECT TERMS

AAAV, Advanced Amphibious Assault Vehicle; CI, Commercial Item; GOTS, Government Off-The-Shelf; JDAM, Joint Direct Attack Munitions; LCC, Life Cycle Cost; MOTS, Modified Off-The-Shelf; NSSN, New Attack Submarine; NDI, Non-Developmental Item; PEM, Plastic Encapsulated Microcircuits; RCOTS, Ruggedized COTS; ROTS, Research Off-The-Shelf; SEE, Software Engineering Environment; TOC, Total Ownership Costs, Spiral Development, Technology Refresh

15. NUMBER OF PAGES

102

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	None

(This Page Intentionally Blank)