



# TRUE PROGRAM SUCCESS '06

## A CASE STUDY AND ASSESSMENT OF A COTS UPGRADE FOR A SATELLITE GROUND SYSTEM

PRICE Systems U.S. Symposium  
April 19-21  
Sheraton San Diego Hotel & Marina  
San Diego, CA

**Marilee Wheaton**  
Aerospace Corporation  
**Kevin McKeel**  
PRICE Systems



## *True COTS Solution : Case study*

- Introduction
- True COTS Solution
  - Why we need it
  - Solution Details
- Case Study Description
- Estimate Details
  - Assumptions
  - True COTS Solution
- COTS Integration Lessons Learned
- Wrap Up

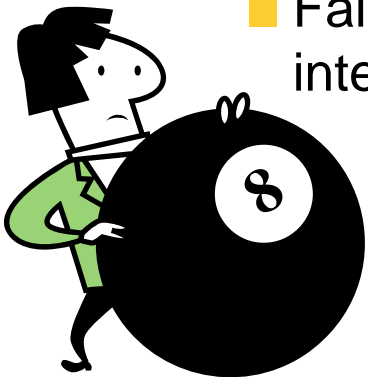


## ***Introduction***

- The integration of Commercial Off the Shelf Software is often misunderstood and underestimated
- Estimators and project planners continue to struggle with estimation of COTS projects
- PRICE True S introduced in 2004 with a comprehensive solution for estimating all of the activities associated with COTS Implementations
- Command and Control COTS Integration project with actual data available
- Opportunity to exercise COTS Solution in True S

## *True COTS Solution : Why we need it*

- COTS solutions can save time and money in the development and life-cycle phases of a software product
- The implementation of COTS Intensive or COTS Based Software deployments are often underestimated
  - Perception that COTS Integrations are plug and play exercises
  - Failure to account for all activities associated with successful COTS integrations
  - Failure to remain flexible in requirements
  - Failure to ask the right questions about a COTS based integration



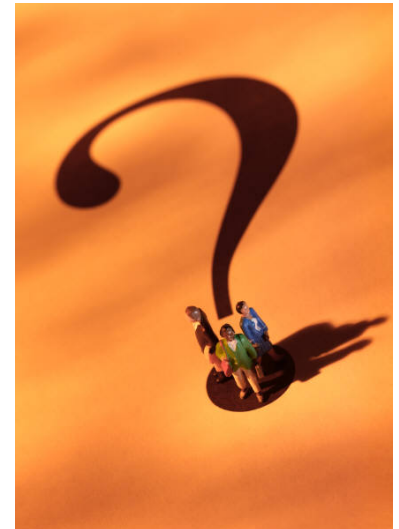
## ***COTS Solution : Solution Details***

- Comprehensive Coverage of COTS related activities
  - Analyze software requirements
  - Identify, evaluate and select COTS components
  - Purchase/Lease/License the COTS Components
  - Tailor COTS software
  - Design, code and test glue code and modifications
  - System level integration and test
  - Evaluate and integrate upgrades
  - Fix bugs



## ***COTS Solutions : Solution Details***

- Ask the right questions
  - Software Size in sensible terms
    - Functional Size
    - Function Points
    - SLOC
    - POPs
  - Glue code description
  - Tailoring scope and complexity
  - Evaluation scope and complexity
  - Vendor and product characteristics
  - Integration team familiarity with COTS integrations
  - Upgrade information



## ***Case Study : Background***

- COTS upgrade for a legacy satellite ground system
- Introduced new distributed client server-based architecture
- Maximized the use of COTS products
- This release of the system consisted primarily of the COTS upgrades
- Small amount of added functionality included as part of this release

## ***COTS Upgrade Development Environment***

- Difficulties with the COTS software created a cascading effect on the schedule
  - Resources from next release being pulled onto this version to fix problems
  - System did not stabilize from COTS upgrade which impacted follow on build
- Insufficient coordination between the multiple development organizations and COTS vendors
  - Complex contractor, subcontractor, vendor relationships
  - Misunderstandings between contractors and COTS vendors with respect to COTS functionality expected
- Limited visibility into the evolution of the NDI code
- Non standard requirements and software configuration management

## Case Study Details

- 7 Existing home grown components
  - Legacy Baseline = 618.6K
  - Deletions and Modifications for Glue Code and new functionality – New Baseline = 658.6K
- One NDI (not developed internally) component
  - Legacy Baseline = 860K
  - Deletions for functionality replaced by COTS – New Baseline = 590K
- 7 COTS Components ranging from midsize to very large
  - Sun Solaris Operating System
  - COTS C2 Product (this replaced the deleted NDI functionality)
  - Oracle
  - Satellite Tool Kit
  - Tivoli Product (communication)
  - Veritas Product (network mgmt)
  - Gensym Product (network mgmt)

## *Case Study Details*

- Complexities
  - Integration team capability => above average
  - External Integration Complexity => High or very high
  - Project constraints => some constraints
  
- Project data
  - Project start date => Jan 2001
  - Project duration => 34 months
  - Project effort => 170K hours
  - Project cost => \$22M



## *Estimation Assumptions*

- Software and System Requirements Analysis set low
  - Existing functionality being replaced
  - Enterprise like components such as OS require less analysis than embedded components
- Documentation costs set low
  - Existing functionality being replaced
- Other overhead costs reduced for allocation to prime
  - Project management
  - Quality Assurance
  - Configuration Management
  - Plan and Oversee

## *Estimation Assumptions*

- Modeled Existing Components as Software Components with New Code for glue code and new functionality and Deleted Code for Deleted Functionality
- Modeled NDI Component as a COTS component
- Modeled COTS Products as COTS Components – using Functional Size Generator to determine size

## *Estimation Assumptions*

- Integration and Test costs require adjustment for several reasons
  - Existing components have already been successfully integrated
  - Enterprise type components have large functional size with a relatively limited percentage that requires integration
  - Operational software components integrate differently with each other than with the mission software components
- In order to accomplish reasonable integration and test values constructed sub-assemblies to represent:
  - Operational Software (software required to get the other components working together – more enterprise type software) – low integration
  - C2 Software – large NDI Component (low integration complexity) with C2 COTS component (high integration complexity)
  - Mission Software – existing homegrown components (average integration complexity) with mission specific COTS component high integration complexity
  - Spread integration and test activity equally between the system and the three sub-assemblies

# Product Breakdown Structure

PRICE TruePlanner - [COTS Paper 2]

File Edit View Tools Window Help

Product Breakdown Structure [PBS]

Estimate: Mission Software

Cost Objects Results Input Sheet Chart

Name: Mission Software Notes: ... Input: Esti

		Value	Unit
1	Start Date		
2	End Date		
3	Assembly Integration Multiplier	0.25	
4	Hardware Platform Stability	Very Stable - Hardware Exists and is F...	Select
5	Hardware Platform Availability	Available more than 95% of the time	Select
6	Number of Unique Hardware Platforms	1	Int
7	Assembly Size Unit	Source Lines of Code (SLOC)	Select
8	SLOC Conversion to Assembly Size Unit	1.0000	
9	Function Point Conversion to Assembly Size Unit	128.0000	
10	POPs Conversion to Assembly Size Unit	25.0000	
11	Use Case Conversion to Assembly Size Unit	80.00	
12	Development Process	Waterfall	Select
13	Number of Spirals or Increments	1	Int
14	Plan Software Development Schedule Multiplier	1.00	
15	Analyze System Requirements Schedule Multiplier	1.00	
16	Design System Schedule Multiplier	1.00	
17	Perform System Qualification Test Schedule Multiplier	1.00	
18	Perform HW/SW Integration and Test Schedule Multiplier	1.00	
19	Perform Assembly Integration and Test Schedule Multiplier	1.00	
20	Perform Software Product Evaluation Schedule Multiplier	1.00	
21	Plan and Oversee Schedule Effect	On	On

Product Breakdown Structure [PBS]

- 1 System
  - 2 System Software
    - 3 Mission Software
      - 4 Tivoli Product
      - 5 Satellite Tool Kit
      - 6 Component A
      - 7 Component B
      - 8 Component C
      - 9 Component D
      - 10 Component E
      - 11 Component F
      - 12 Component H
        - 13 C2 Software
          - 14 Component J (NDI)
          - 15 COTS C2 Product
          - 16 Operational Software
            - 17 Sun Solaris Operating System
            - 18 Oracle
            - 19 Veritas Product
            - 20 Gensym Product

# Typical COTS Input Set

Name:  Notes:  Input:

		Value	U
1	Start Date	1/1/2001	
	End Date		
	Size Units	Functional Size	Se
	Functional Size	750	...
	Amount for Modification	0.00	
	Functional Complexity	6.70	...
	Target Operating Specification	1.20	...
	Off the Shelf Operating Specification	1.00	...
	Glue Code Size Units	Source Lines of Code (SLOC)	Se
	Glue Code Size	0	...
	Glue Code Language	C	Se
	Components under Evaluation	0.00	I
	Components under Detailed Evaluation	0.00	
	Evaluation Multiplier	2.37	
	Tailoring Complexity	1.00	...
	Tailoring Multiplier	9.00	
	Vendor and Product Complexity	1.00	...
	Project Constraints	0.55	...
	Integration Team Maturity	4.00	...
	External Integration Complexity	4.00	...
21	Amount for Purchased Software	0.00	Cu
22	Date of External Software Purchase		
23	Annual Support Fee	0.00	Cu
24	Upgrade Frequency	Annually	Se
25	Analyze Software Requirements Schedule Multiplier	1.00	

**Business Applications**  
e.g. MIS, Office Automation, Customer Relationships Management, Purchasing/Inventory Control, Human Resources, Database

**Decision Support**  
e.g. Expert or decision support, ERP Systems

**Computational/Graphics**  
e.g. Compilers, Imaging, Sensing and Mapping, Graphical

**Network Management**

**Communications/Controls**  
e.g. Telecommunications, Communications

**Controls and Displays**

**Radar/Satellite**  
e.g. Telemetry, Satellite Data Link

**Operating Systems**  
e.g. Text Based Operating Systems, Graphical Operating Systems

**Military Support**  
e.g. Weapons Management, Encryption, Weapon Control, Guidance Control

**Generator Outputs**

Functional Size Units

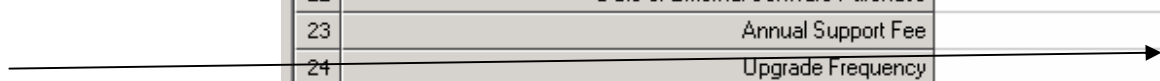
Functional Complexity

Evaluation Multiplier

Tailoring Multiplier

- Low mid**  
e.g. Crysti tracking/s
- Midsize**  
e.g. Advan analysis to
- High mi**  
e.g. Comp full compil
- Midsize**  
e.g. MS Ex
- Large**  
e.g. Orad environme
- Very lar**  
e.g. entire
- Extra la**  
e.g. comp enterprise

Update Frequency



# Typical COTS Input Set

**COTS Paper 2 (Tables and Generators)**

**PRICE True S Catalog**

**PROJECT CONSTRAINTS**

Some Additional Constraints ▾ Communication/Timing Constraints  
 Some Additional Constraints ▾ Performance Constraints  
 Some Additional Constraints ▾ Memory Constraints

**PROJECT CONSTRAINT INDEX**

0.55 Manageable constraints

Most software projects have or memory. In cases where o understand the impact on sometimes when one of these on productivity may be reduced used to compensate for areas a value between 0.5 (typical)

OK Cancel

Estimate: Satellite Tool Kit

Cost Objects | Results | **Input Sheet** | Chart

Name: Satellite Tool Kit Notes: ... Input: E

		Value	U
1	Start Date	1/1/2001	
2	End Date		
3	Size Units	Functional Size	Se
4	Functional Size	750	...
5	Amount for Modification	0.00	
6	Functional Complexity	6.70	...
7	Target Operating Specification	1.20	...
	Off the Shelf Operating Specification	1.00	...
	Glue Code Size Units	Source Lines of Code (SLOC)	Se
	Glue Code Size	0	...
	Glue Code Language	C	Se
	Components under Evaluation	0.00	
	Components under Detailed Evaluation	0.00	
	Evaluation Multiplier	2.37	
	Tailoring Complexity	1.00	...
	Tailoring Multiplier	9.00	
	Vendor and Product Complexity	1.00	...
	Project Constraints	0.55	...
	Integration Team Maturity	4.00	...
	External Integration Complexity	4.00	Se
	Amount for Purchased Software	0.00	Ct

**COTS Paper 2 (Tables and Generators)**

**PRICE True S Catalog**

**EXTERNAL INTEGRATION COMPLEXITY**

- No external Integration
- Very low - Experienced team with few integration points
- Low - Average team with few integration points
- Nominal - Average team with many integration points
- High - Inexperienced team with few integration points
- Very high - Inexperienced team with many integration points

**External Integration Complexity** 4

OK Cancel

## Results – Labor Hours

Labor Requirement Profile: System

Cost Objects | Results | Input Sheet | Chart

System Total Labor Requirements = 179,291.21 Hours View: Labor Requirement

Units: Hours Timescale: Monthly

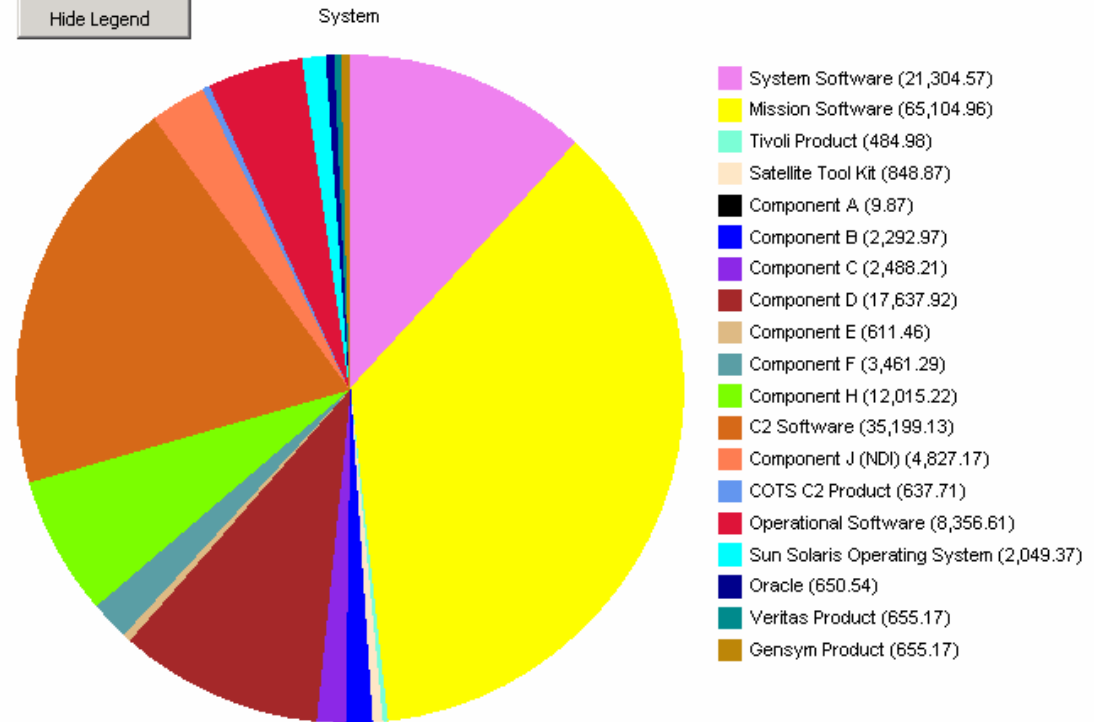
	Labor Requirement Profile : System - [System] Labor Requirements in Hours	Total	12/1/2000	1/1/2001	2/1
1	System Software	21,304.57	452.40	345.87	
2	Mission Software	65,104.96	5,039.46	1,999.79	
3	Tivoli Product	484.98		80.69	
4	Satellite Tool Kit	848.87		88.58	
5	Component A	9.87		2.53	
6	Component B	2,292.97		310.95	
7	Component C	2,488.21		341.29	
8	Component D	17,637.92		709.15	
9	Component E	611.46		133.53	
10	Component F	3,461.29		295.66	
11	Component H	12,015.22		612.87	
12	C2 Software	35,199.13	1,968.49	1,500.71	
13	Component J (NDI)	4,827.17		937.78	
14	COTS C2 Product	637.71		110.96	
15	Operational Software	8,356.61	644.59	326.87	
16	Sun Solaris Operating System	2,049.37		392.87	
17	Oracle	650.54		61.36	
18	Veritas Product	655.17		46.04	
19	Gensym Product	655.17		46.04	
20	<b>Total</b>	<b>179,291.21</b>	<b>8,104.94</b>	<b>8,343.55</b>	<b>1</b>

# Results Labor Hours Per PBS Element

Product Breakdown Structure [PBS]	
1	System
2	System Software
3	Mission Software
4	Tivoli Product
5	Satellite Tool Kit
6	Component A
7	Component B
8	Component C
9	Component D
10	Component E
11	Component F
12	Component H
13	C2 Software
14	Component J (NDI)
15	COTS C2 Product
16	Operational Software
17	Sun Solaris Operating System
18	Oracle
19	Veritas Product
20	Gensym Product

Copy Chart  
Print Chart  
Hide Legend

Cost Object Requirements



Labor Requirements in Hours

## Results – Costs in \$

Cost Table: System Software

Cost Objects | Results | Input Sheet | Chart

System Software Cost = \$22,551,438 View: Cost Table

	Cost Table : System Software - [Software Assembly] Currency in USD (\$) (as spent)	Total	Programmer	System and Software Engineers
1	Perform Software Maintenance	0	0	0
2	Manage Project	4,004,146		746,128
3	Perform Configuration Management	1,540,962	39,694	26,447
4	Perform Joint Technical Reviews	0		0
5	Perform Quality Assurance	1,498,228	19,847	26,447
6	Plan and Oversee	573,398		494,452
7	Plan Software Development	880,374		363,685
8	Write Documentation	765,867	10,608	11,294
9	Analyze System Requirements	23,159		23,159
10	Design System	72,114	16,599	55,514
11	Analyze Software Requirements	1,330,452		1,330,452
12	Perform Assembly Integration and Test	6,023,883	757,740	2,381,382
13	Purchase External Software	0		
14	Evaluate and Select	0	0	0
15	Design Software	2,990,406	559,001	2,431,405
16	Tailor	269,922	186,678	83,243
17	Code and Unit Test	924,425	803,234	121,191
18	Perform Software Integration and Test	461,413	109,086	177,075
19	Perform Software Qualification Test	571,701	98,126	276,094
20	Perform HW/SW Integration and Test	380,080	112,474	120,248
21	Perform Software Product Evaluations	0		0
22	Perform System Qualification Test	240,908	59,370	130,840
23	<b>Total</b>	<b>22,551,438</b>	<b>2,772,458</b>	<b>8,799,057</b>

## *Lessons Learned*

- Steps taken to improve overall integration of processes between the contractors
- Common software development configuration management system enabled team to more readily share source code baselines
- NDI should be reclassified as mission unique code and its evolution managed as such
- Evolution of COTS and NDI components need to follow robust systems engineering processes
  - Especially in areas of requirements definition and risk management

## *Lessons Learned*

- Be aware when important decisions are being driven by schedule considerations at the expense of technical considerations
- More systematic process for specifying requirements for each new version of COTS and NDI software
  - Include provisions in development schedule for these activities
- Full appreciation of system and software engineering practices is required through the life cycle of COTS-based systems

## ***Conclusion***

- Planning and budgeting for COTS upgrades present unique challenges to a project manager
- Use the COTS solution capabilities of PRICE True S
  - Provides for comprehensive coverage of COTS related activities
  - Asks the right questions to get needed insight
  - Provides consistent and defensible estimates of costs
- Make COTS-based systems tradeoffs and realign budgets for COTS realities

## About PRICE Systems

- Leader in Program Affordability Management solutions
- Combine cost estimating, project control, and knowledge management – *ensuring project success at every decision gateway*
- Customers increase visibility, minimize risk and cost, accelerate project development, and improve the effectiveness of project selection, control and delivery

[www.PRICESystems.com](http://www.PRICESystems.com)



POWERED BY  
**PRICE** Research

**PRICE**