Software Maintenance, Sustaining Engineering, and Operational Support

Estimating Software Maintenance Costs for U.S. Army Weapons Systems

Office of the Deputy Assistant Secretary of the Army for Cost and Economics (ODASA-CE)
PSM User' Group Meetings and Workshops
29 July 2012

Objective

Provide the Department of the Army with the ability to accurately estimate, budget, allocate, and justify the software maintenance, sustaining engineering, and operational support resources required to meet evolving mission and service affordability requirements across the program life-cycle

Presentation Outline

- ***Current software maintenance estimation technology does not support objective resource decisions in the emerging Army systems sustainment environment***
- ***Defining a viable software maintenance cost estimation methodology for Army programs - project requirements, approach, and initial results***
- ***Integrated software maintenance life cycle cost estimation model - linking software maintenance resource requirements to program and functional domain sustainment profiles***

Estimating Software Maintenance Costs in the Current Army Sustainment Environment

Army Software Maintenance Environment

- ***Software is the "default component" for increasing system capability and performance***
- ***Operational requirements are dynamic and complex***
- ***Maintenance budgets are becoming more constrained and vulnerable***
- ***Difficult to defend program software life cycle cost estimates and annual maintenance budget requests***
- ***Limited understanding of the relationships between software investments, applied resources, product outputs, and mission capability***
- ***Multiple perspectives: enterprise - program - maintenance organization - user base***
- ***Existing software maintenance estimation methodologies are inadequate***

Software System Growth

107 - AH-64As

1620 - AH-64Ds

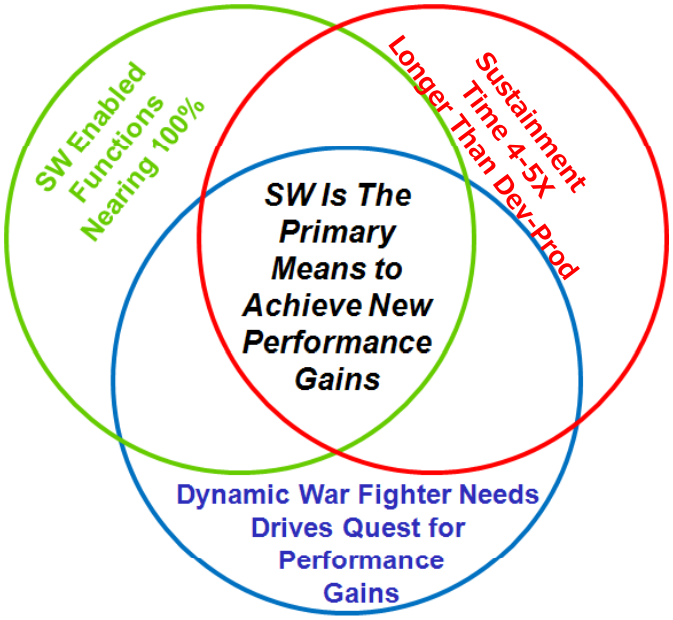
Apache Software Growth
300 KSLOC to Over 1.4 Million SLOC

Software Configuration Complexity

4,300 - M1A1 & variants
580 - M1A2 & variants
580 - M1A2 SEP & variants

- ***Multiple system variants drive multiple parallel software release baselines***
- ***Different "types" of concurrent software releases***
  - ***Correct defects***
  - ***Fault prevention - mandates***
  - ***Enhance functionality***
  - ***Adapt to new environments***
- ***Multiple change drivers***
  - ***End user requirements***
  - ***Mission evolution***
  - ***System interoperability***
  - ***Change mandates***
  - ***Technology***
  - ***Technical debt***

Significant Growth in Army Software Maintenance Resource Requirements ($)



- ***Reliance on software changes to meet evolving mission requirements***
- ***Proliferation of system software variants***
- ***Increasingly complex system to system functional interfaces***
- ***Proliferation of change drivers***
- ***Functional change backlogs***
- ***Budget realignments***

"We've lived in a rich man's world where there has been less emphasis on cost over the past 10 years."

Dr. Jacques Gansler

Current Software Maintenance Estimation Methods

- ***Parametric models***
  - ***Only include corrective, perfective, adaptive changes & enhancements***
  - ***Invalid assumptions about sustaining engineering tasks***
  - ***Historical data not visible in all models***
  - ***Not calibrated - validated***
- ***Past software maintenance funding - effort***
  - ***Limited access to historical FTE - Budget data***
  - ***Represent "rolled-up" costs***
  - ***Requirements or LOE funding?***
  - ***Availability of correlated program development - sustainment cost data***

Current Estimation Methods (continued)

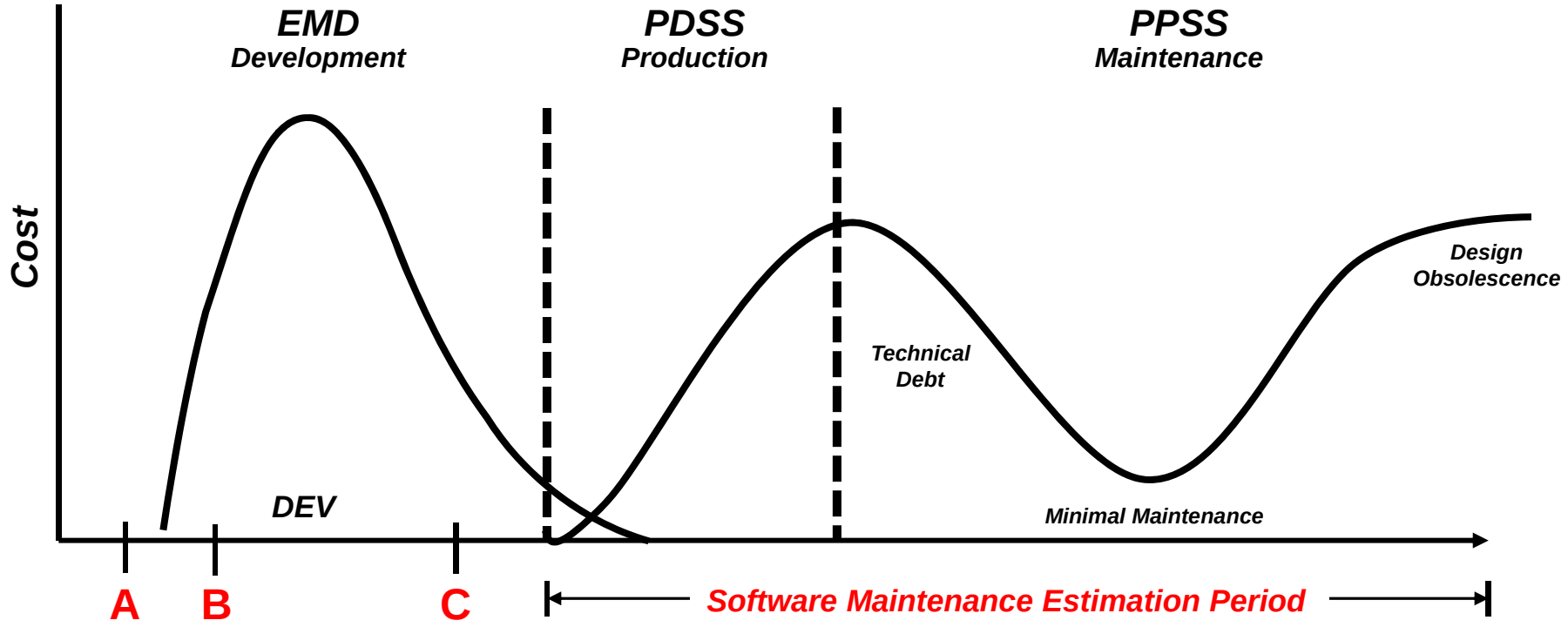
- ***Number of lines of code per software engineer***
  - ***Each engineer can maintain 20K-25K LOC/ESLOC***
  - ***Does not reflect the impact of software reuse or COTS***
- ***Software maintenance estimated as a percentage of development costs***
  - ***Rule(s) of thumb - development based:***
    - ***S/W maintenance costs - 2/3 of total S/W life cycle costs***
    - ***S/W maintenance costs - 60% to 75% of total S/W life cycle costs***
    - ***Annual S/W maintenance costs - 5% to 10% of total S/W life cycle costs***
  - ***Ignores total system life cycle software growth and maintenance requirements/strategy/tasks***

All of these methods have significant limitations in the current and future Army environment

Key Estimation Issues

- ***Current software maintenance estimation methods:***
  - ***Do not align with an individual program's unique life cycle sustainment profile***
  - ***Assume software maintenance resource requirements are consistent after system deployment***
  - ***Do not take into account specific software maintenance products and related activities for a given program (what's in and what's out)***
  - ***Do not encompass multiple types of cost relationships***
  - ***Do not address the differences across functional software domains***
  - ***Are high level constructs based on significant assumptions of commonality***

Notional Software Maintenance Life-Cycle Cost Model



Summary

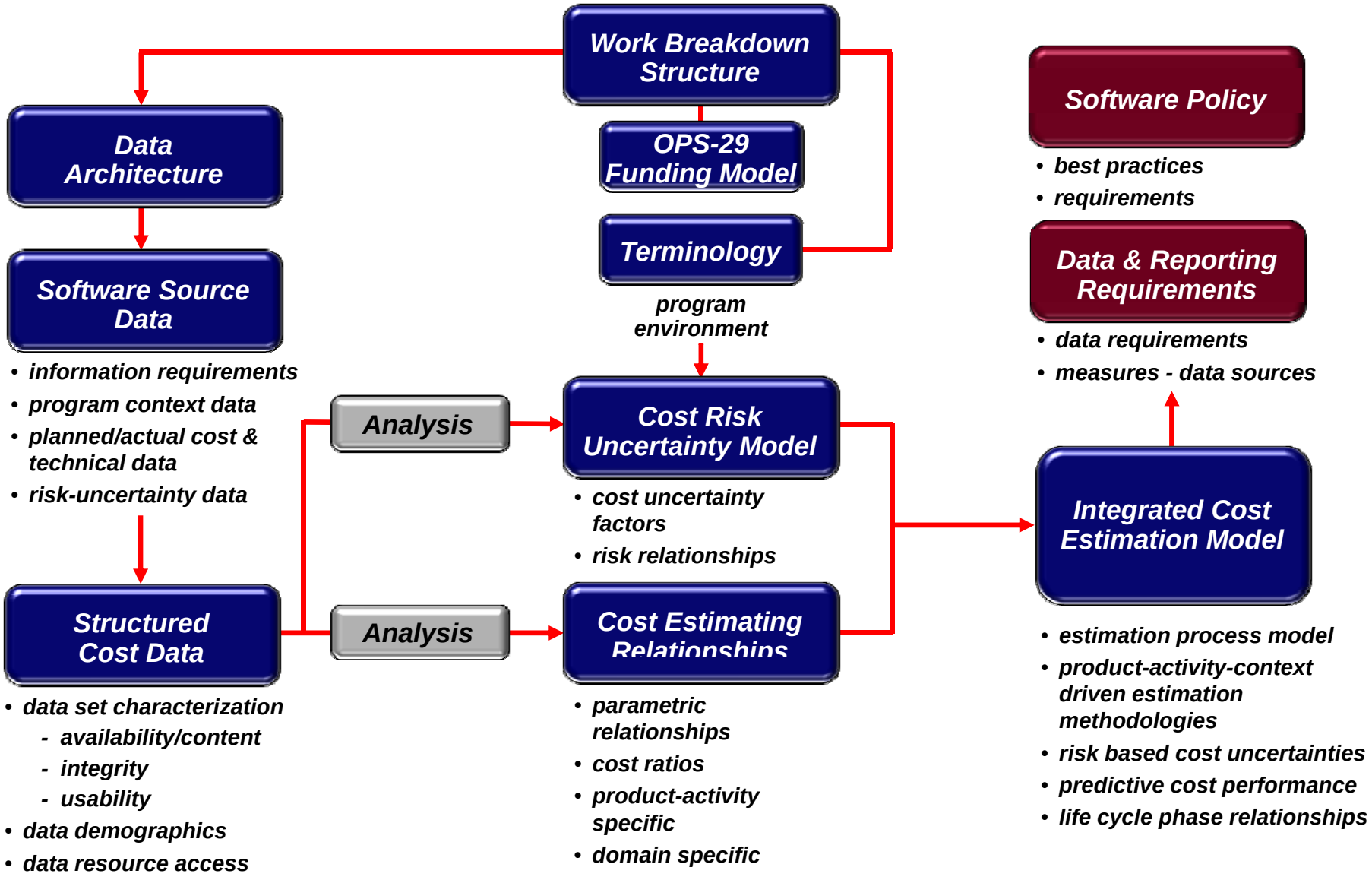
- ***Army system mission capability is dependent upon the adequate resourcing of program software maintenance requirements***
- ***To ensure this the Army needs a software maintenance estimation methodology that better reflects the software change profile and related employment characteristics of the system***

Software Maintenance Estimation Project Requirements, Approach, and Initial Results

Estimation Methodology Requirements

- ***Define a software maintenance cost estimation methodology that integrates the product release and software support costs over the life-cycle of a program***
  - ***Configurable to the specific program software maintenance improvement and release strategy***
  - ***Encompasses PDSS and PPSS efforts***
  - ***Applicable at Milestones A, B, and C***
  - ***Integrates component cost estimates from multiple CER based product and activity estimates (WBS defined)***
  - ***Directly related to development parameters and performance***
  - ***Can incorporate cost risk uncertainty at component and higher levels***
  - ***Practical - defensible - data derived - adaptable***

Software Maintenance, Sustaining Engineering, and Operational Support



Project Approach

- ***Establish software maintenance technical standards and baselines***
  - ***Software maintenance products and activities***
  - ***Configurable and adaptable WBS***
  - ***Army software maintenance and sustaining engineering process model***
  - ***Common terminology***
- ***Collect, evaluate, and structure relevant Army software maintenance data***
  - ***Data collection***
    - ***Army program cost data - all relevant parameters***
    - ***Army program, organizational, and enterprise context data***
    - ***Navy, Air Force, and contractor data***
  - ***Data evaluation - availability, integrity, usability***
  - ***Data schema and accessible data store***
    - ***Historical cost data***
    - ***Derived CERs***
  - ***Estimation data requirements***

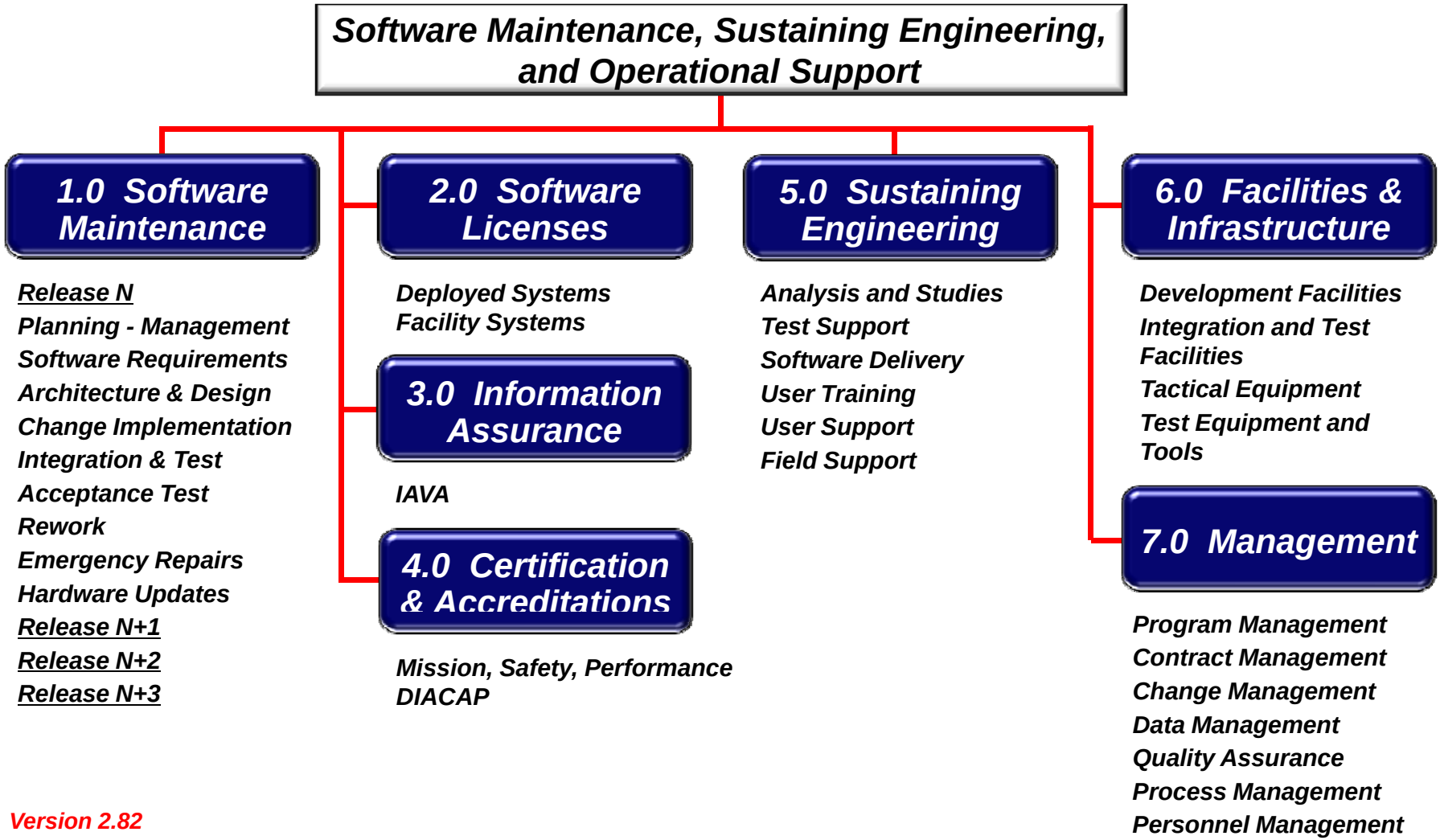
Project Approach (continued)

- ***Independent analysis and model development***
  - ***Data and information model***
  - ***Product sizing models***
  - ***Cost estimation relationships***
  - ***Cost risk - uncertainty model***
  - ***Life cycle integrated cost estimation model***
- ***Define Army software maintenance data collection and policy change requirements***
  - ***Information driven program and organization data requirements***
  - ***Post development policy requirements***
  - ***Required information infrastructure changes***
- ***Implement and improve***
  - ***Data stores - estimation assets***
  - ***Estimation and risk-uncertainty model***
  - ***Estimation processes***
  - ***Emerging stakeholder information requirements***

Work Breakdown Structure

- ***Common structure that includes all potential products and activities - "what's in - what's out"***
- ***Common definitions - terminology***
- ***Emphasis on DOD weapons systems***
- ***Basis for identifying specific cost elements attributable to a given program or system maintenance/sustaining engineering effort***
- ***Product based - driven by changes to the software baseline(s)***
- ***Release focused - primary software maintenance product***
- ***Intended to be tailored and adapted for each program or organization:***
  - ***Addition/deletion of lower level cost elements***
  - ***Re-binning of lower level cost elements***
- ***Foundation for cost estimation process/models***
- ***Basis for defining cost estimating relationships***

Work Breakdown Structure



Version 2.82

Work Breakdown Structure

1.0 Software Maintenance - products and activities associated with modifying an operational software product or system

2.0 Software Licenses - products and activities associated with the procurement and renewal of software licenses for operational software

3.0 Information Assurance - products and activities associated with ensuring that the software is compliant with externally defined information assurance requirements

4.0 Certifications and Accreditations - products and activities associated with verifying a software system against externally defined domain performance criteria

5.0 Sustaining Engineering - products and activities associated with supporting a deployed software product or system in its operational environment

6.0 Facilities & Infrastructure - products and activities associated with establishing and operating the facilities and processes required to modify, integrate, and test operational software products or systems

7.0 Management - products and activities associated with planning, organizing, funding, and controlling the resources required to support operational software products or systems

Program WBS Element Coverage

WBS	Title	#	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1.0	**Software Maintenance**	17	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■													
1.1	Software Release (N)	7	■	■	■	■	■	■	■																							
1.1.1	Planning and Management	19	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■											
1.1.2	Software Requirements	24	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■						
1.1.3	Architecture and Design	24	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■						
1.1.4	Change Implementation	23	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■							
1.1.4.1	Change Development	7	■	■	■	■	■	■	■																							
1.1.4.2	Unit Testing	7	■	■	■	■	■	■	■																							
1.1.5	Integration and Test	24	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■						
1.1.5.1	Test Planning	8	■	■	■	■	■	■	■	■																						
1.1.5.2	Test Scenario and Test Case Development	8	■	■	■	■	■	■	■	■																						
1.1.5.3	Test Tool Development	8	■	■	■	■	■	■	■	■																						
1.1.5.4	Test Conduct	8	■	■	■	■	■	■	■	■																						
1.1.6	Acceptance Testing	29	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	
1.1.6.1	Test Planning	7	■	■	■	■	■	■	■																							
1.1.6.2	Test Scenario and Test Case Development	7	■	■	■	■	■	■	■																							
1.1.6.3	Test Tool Development	7	■	■	■	■	■	■	■																							
1.1.6.4	Test Conduct	7	■	■	■	■	■	■	■																							
1.1.7	Rework	7	■	■	■	■	■	■	■																							
1.1.8	Emergency Repairs	25	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■					
1.1.9	Hardware Updates	15	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■															
1.2	Software Release (N+1)	0																														
1.3	Software Release (N+2)	0																														
1.4	Software Release (N+3)	0																														
2.0	**Licenses**	11	■	■	■	■	■	■	■	■	■	■	■																			
3.0	**Information Assurance**	27	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■			
3.1	IAVA	12	■	■	■	■	■	■	■	■	■	■	■	■																		
3.2	DIACAP	11	■	■	■	■	■	■	■	■	■	■	■																			
4.0	**Certifications & Accreditations**	12	■	■	■	■	■	■	■	■	■	■	■	■																		

Program WBS Element Coverage (cont.)

5.0	**Software Sustaining Engineering**	0
5.1	Analysis and Studies	18
5.2	Test Support	16
5.3	Delivery	15
5.4	User Training	15
5.5	User Support	16
5.6	Field Support	26
6.0	**Software Facilities and Infrastructure**	16
6.1	Development Facilities	11
6.1.1	Equipment	7
6.1.2	Licenses	11
6.1.3	Operations	7
6.2	Integration and Test Facilities	27
6.2.1	Equipment	7
6.2.2	Licenses	14
6.2.3	Operations	11
6.4	Tactical Equipment	11
6.5	Test Equipment and Tools	27
7.0	**Management**	11
7.1	Program Management	29
7.1.1	Project Release Management	12
7.1.2	Risk Management	11
7.1.3	Measurement/ Analysis	26
7.2	Contract Management	17
7.3	Change Management	27
7.4	Data Management	24
7.5	Quality Assurance	29
7.5.1	IV&V	26
7.6	Process Management	12
7.7	Personnel Management	0

Army Software Maintenance Data

		Number of Releases	WBS Task Performed	Interfaces	Effort	Cost ($)	Schedule	Software Change Requests (SCRs)	Requirements	Defects	Backlog	Source Lines of Code	OPS-29 Data
Org 1	Program 1	3											
	Program 2	1											
	Program 3	1											
	Program 4	4											
	Program 5	1											
	Program 6	3											
	Program 7	1											
	Program 8	1											
	Program 9	1											
	Program 10	4											
	Program 11	4											
	Program 12	1											
	Program 13	1											
Org 2	Program 1	2											
	Program 2	2											
	Program 3	5											
	Program 4	3											
	Program 5	1											
	Program 6	4											
	Program 7	2											
	Program 8	2											
	Program 9	1											
Org 3	Program 1	1											
Org 4	Program 1	6											
	Program 2	2											
	Program 3	3											
	Program 4	7											
Org 5	Program 1	Multiple											
	Program 2	Multiple											
	Program 3	Multiple											
Org 6	Program 1	3											
	Program 2	6											
	Program 3	3											
	Program 4	8											
	Program 5	1											
		88											

Data Analysis Summary

- ***Availability of data varied significantly by organization and program***
- ***Execution data (actuals) was very limited - most data was planning data aligned with the OPS-29 funding model***
- ***Multiple funding streams exist for similar work***
- ***Data was cost based - not product/activity based***
- ***Software engineering data derived from different source artifacts for a given release were sometimes inconsistent***
- ***The aggregation levels of provided data were inconsistent (details vs. summary)***
- ***The data that was available provided significant insight into potential cost estimating relationships and maintenance products and activity allocations***

Operating Environments - Domains

Count of Operating Environment

	Helicopter	UAV	Ground	Missile	Grand Total
Mission Processing			5		5
Real Time Embedded - Other	3		5		8
System			4		4
Training			6		6
Vehicle Control		2		2	4
Vehicle Payload			2		2
Grand Total	3	2	22	2	29

Count of Super-Domain

	Helicopter	UAV	Ground	Missile	Grand Total
Mission Critical - Embedded	3	2	7	2	14
Mission Critical - Non-Embedded			9		9
Mission Support			6		6
Grand Total	3	2	22	2	29

Application Domain Types

Embedded

- *Sensor Control and Signal Processing*
- *Vehicle Control*
- *Vehicle Payload*
- *Real Time Embedded*
- *Mission Processing*

Non Embedded

- *Systems Software*
- *Automation and Process Control*
- *Simulation & Modeling*

Mission Support

- *Training*
- *Test*
- *Data Processing*

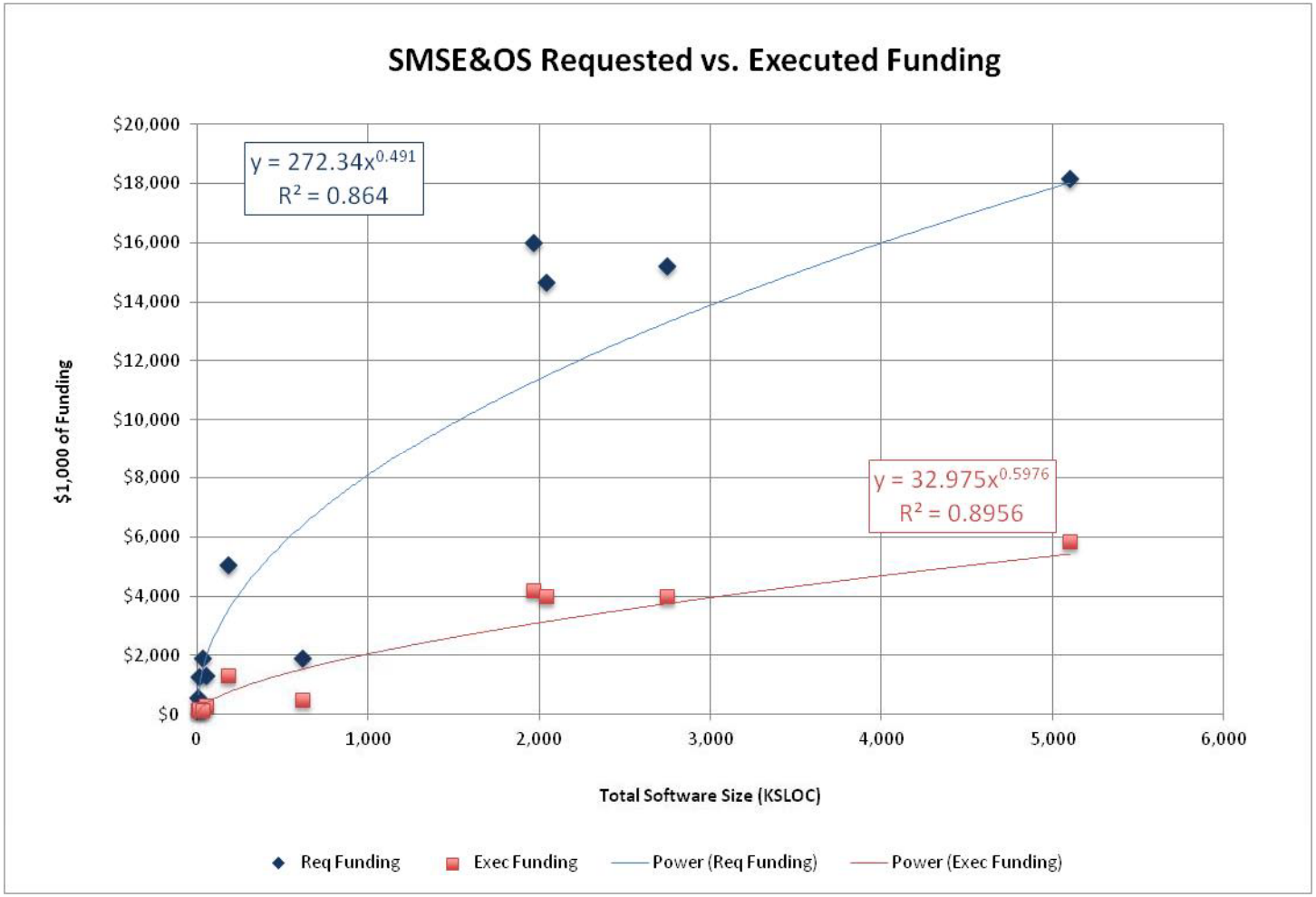
Cost Estimating Relationships

- ***Types of cost estimating relationships:***
  - ***Systemic ratios***
  - ***Parametric models***
  - ***Simple relationships***
  - ***Historical cost trends***
- ***Approach:***
  - ***Ratios useful to identify performance relationships***
  - ***Product costs structured into software releases - estimated parametrically based on functional change content***
  - ***Activity costs estimated using simple CER relationships - unique variables and drivers***
  - ***Fixed costs based on historical resource expenditures***
  - ***Related factors influence and/or modify the outcomes in all cases***

Correlation Coefficients Matrix

Correlation Coefficients Matrix						
Sample size	8	*Critical value (10%*	1.94			
		Interfaces	***Age***	***Est_Total_KSLOC***	***Req_Fund_FY11***	***Exec_Fund_FY11***
Interfaces	**Pearson Correlation Coefficie**	*1.00*				
	R Standard Error					
	t					
	p-value					
	H0 (10%)					
Age	**Pearson Correlation Coefficie**	*0.47*	*1.00*			
	R Standard Error	*0.13*				
	t	*1.31*				
	p-value	*0.24*				
	H0 (10%)	*accepted*				
Est_Total_KSLOC	**Pearson Correlation Coefficie**	*0.70*	*0.23*	*1.00*		
	R Standard Error	*0.08*	*0.16*			
	t	*2.43*	*0.58*			
	p-value	*0.05*	*0.58*			
	H0 (10%)	*rejected*	*accepted*			
Req_Fund_FY11	**Pearson Correlation Coefficie**	*0.39*	*0.30*	*0.87*	*1.00*	
	R Standard Error	*0.14*	*0.15*	*0.04*		
	t	*1.04*	*0.77*	*4.41*		
	p-value	*0.34*	*0.47*	*0.00*		
	H0 (10%)	*accepted*	*accepted*	*rejected*		
Exec_Fund_FY11	**Pearson Correlation Coefficie**	*0.50*	*0.59*	*0.81*	*0.94*	*1.00*
	R Standard Error	*0.13*	*0.11*	*0.06*	*0.02*	
	t	*1.41*	*1.78*	*3.40*	*6.87*	
	p-value	*0.21*	*0.13*	*0.01*	*0.00*	
	H0 (10%)	*accepted*	*accepted*	*rejected*	*rejected*	

Funding vs. Software Size



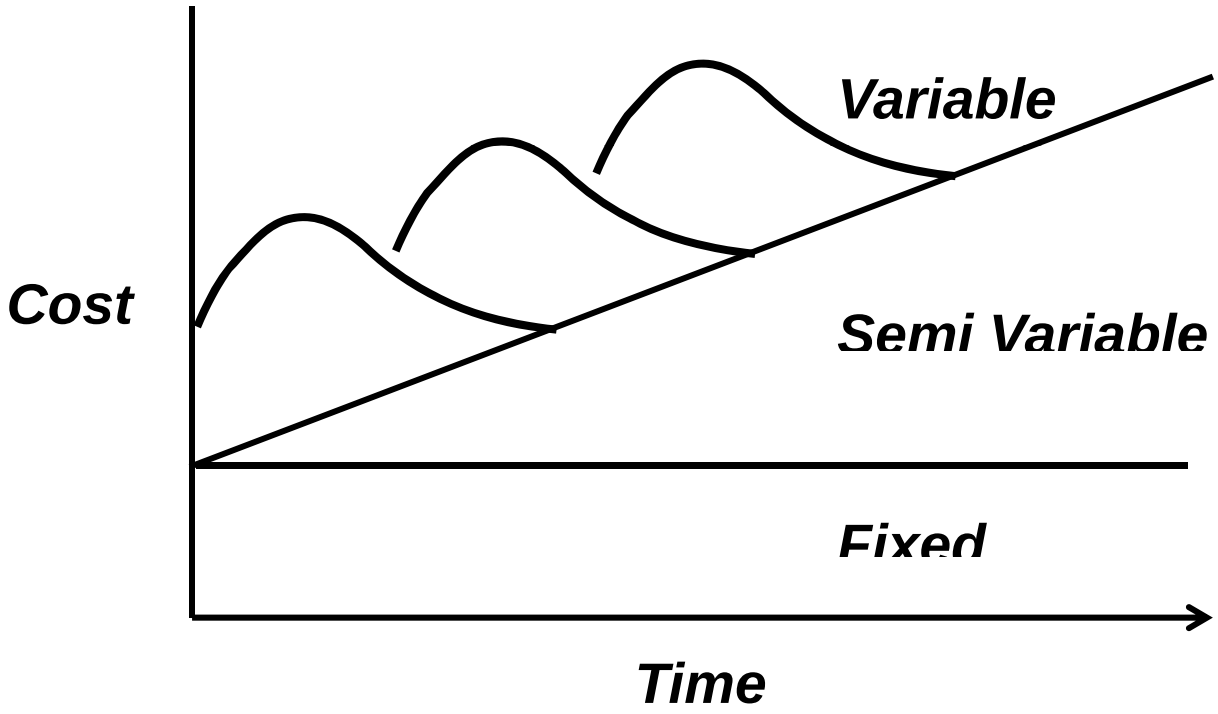
Preliminary Cost Relationships

- ***Results***
  - ***Estimated total system KSLOC and number of interfaces had the highest correlation to both funded and executed funding***
  - ***Requested and executed funding varied by a factor of 3***
  - ***There is an implied prioritization of applied funding by subsystem (domain - weapons system vs. training system)***
  - ***System age had no appreciable cost impact***
  - ***There is a non-linear relationship between software system size and cost for software maintenance (resource constraints)***
  - ***Measurable software development characteristics available early in the life cycle may be early predictors of software maintenance life cycle cost***

WBS Based Cost Estimating Relationships

WBS	Title	Army OPS-29 Mapping	Variability	Cost Drivers Examples
1.1	Software Release			Lower level cost roll-up
1.1.1 to 1.1.5	Planning, Req'ts, Design, Implementation, Development, Unit Testing, Integration & Test (typical SW Development)	Capability Sets FY(XX/XX) System Mission Capability	Variable Cost	Parametric modeling - Number of Requirements (Enhancements), Defects, Test Cases - added, reused, modified; App. Domain, Complexity, Reliability, Personnel Factors
1.1.6	Acceptance Test	Capability Sets FY(XX/XX) System Mission Capability	Variable Cost	Number of requirements / capabilities / … / test cases / etc.)
2.0	Licenses	Licenses	Semi-Var	Type and number of COTS products.
3.1	IAVA	IAVA	Semi-Var Cost	Parametric model (percentage distribution); Application domain, size, effort staffing, duration, productivity
4.0	Certifications & Accreditations	C&A	Fixed	Cost per C&A
5.0	SW Sustaining Engineering		Roll-up	Lower level cost roll-up
5.1 to 5.5	Analysis, Test Support, Delivery, Training	Organic Labor	Semi-Var Cost	LOE
5.6	Field Support	FSEs	Semi-Var	LOE, No. of Field Sites
6.0	Software Facilities & Infrastructure	System Infrastructure or System Open Door	Fixed	# People at facility, Simulation / Test Equip Maintenance
7.0	Management	Organic	Roll-up	Lower level cost roll-up
7.1 to 7.7	Program / Release / Risk / Contract Management		Fixed Cost	LOE

Types of Cost Estimating Relationships



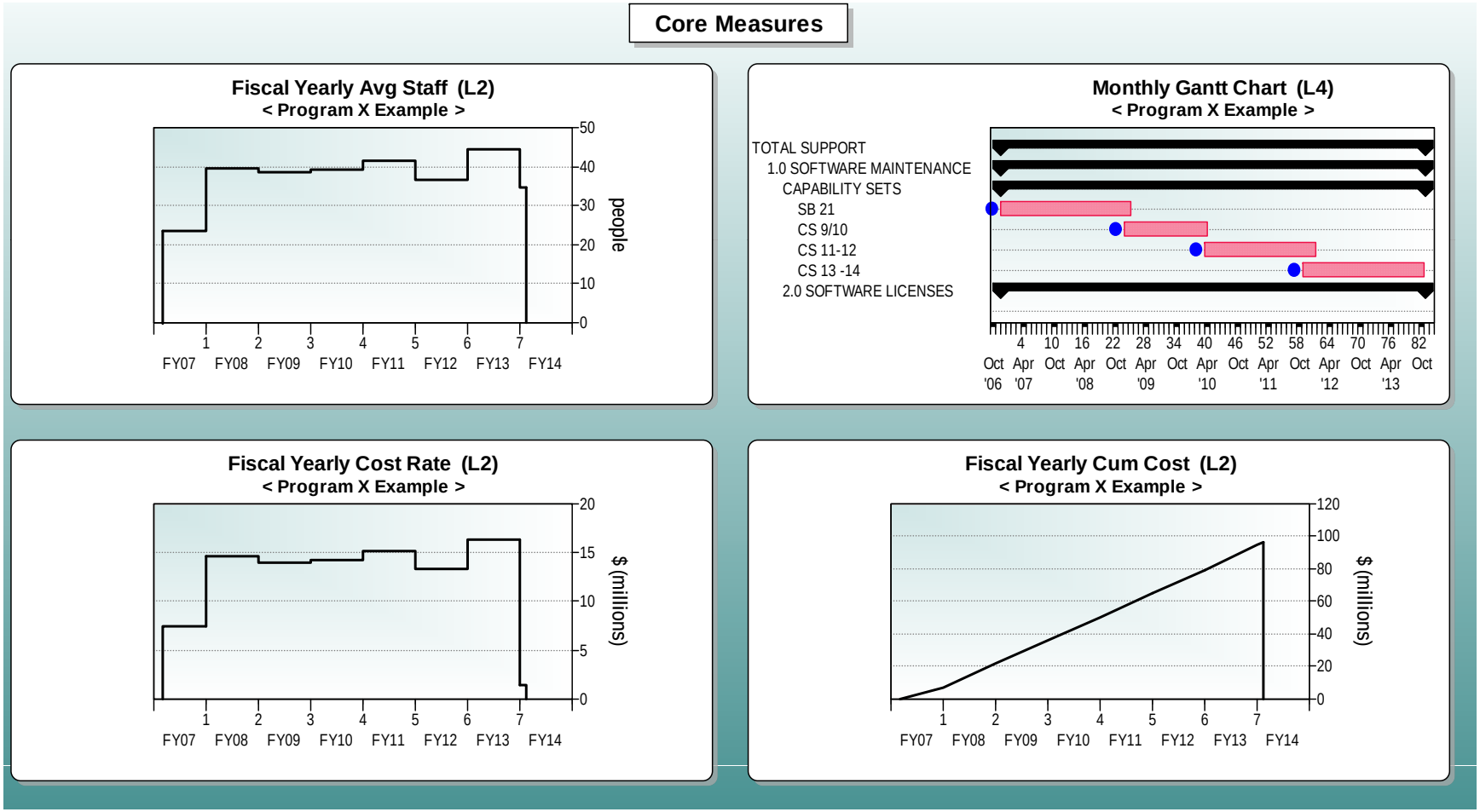
Software Maintenance Cost Estimation Model

- ***Objectives***
  - ***Use available data to construct a multi-CER cost estimate for a sample program - compare to program actuals***
  - ***Identify program cost allocations***
  - ***Validate integrity - usability of provided data***
- ***Scope***
  - ***5 Year PPSS time period - 4 capability sets (2006-2011)***
  - ***All included software maintenance WBS elements***
  - ***Applied CERs - Proof of Concept***
  - ***Parametric model - capability sets / releases (SLIM)***
  - ***Semi-variable - IAVAs, licenses, certification, etc.***
  - ***Fixed costs - infrastructure and management***

Software Maintenance Cost Estimation Model

- ***Results for one sample program***
  - ***Software capability sets are most significant cost item based on program allocations and expenditures (55%)***
  - ***Licenses, IAVAs, and Certifications & Accreditations were relatively stable year-to-year (small adjustments for inflation) - due in part to limited COTS on this program***
  - ***Facilities and Management were also relatively stable year-to-year***
  - ***Requested funding is relatively consistent across the timeline***
  - ***Data was noisy - inconsistent parameters from different program data sources***
  - ***Model cost prediction was consistent with executed costs***

Initial Model



Integrated Software Maintenance Life Cycle Cost Estimation Model

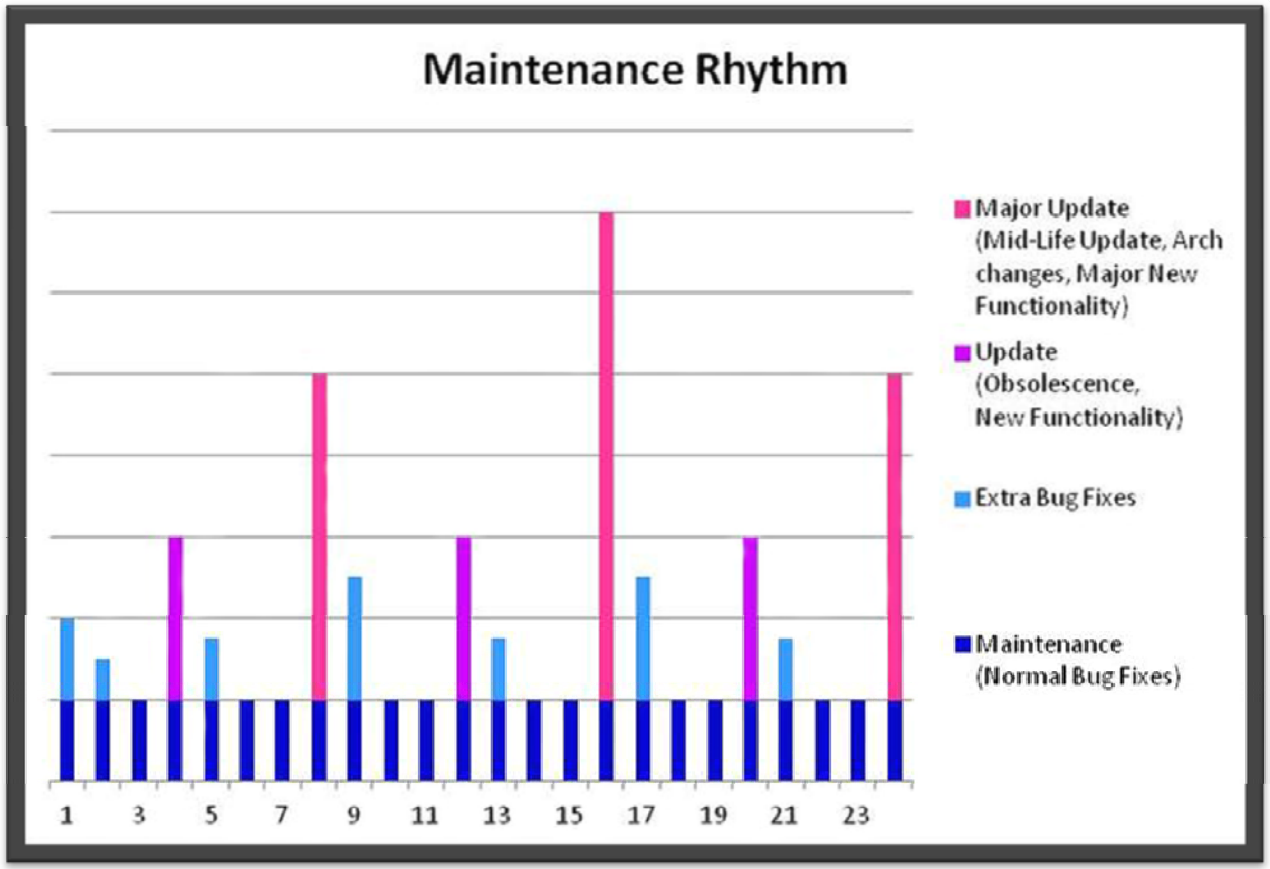
Integrated Cost Estimation Life Cycle Model

- *Two key components*
- *Software maintenance cost risk-uncertainty model*
  - *Risk information model*
  - *Risk interdependency analysis model*
  - *Cost estimation application process model*
- *Integrated life cycle cost allocation model*
  - *Program software life cycle product-activity profile*
    - *Post-development software releases*
    - *Fielded software sustaining engineering support*
    - *Software maintenance site infrastructure*
  - *Maintenance resource-funding allocation model(s)*
  - *Integrated cost estimation results*

Cost Risk-Uncertainty Model

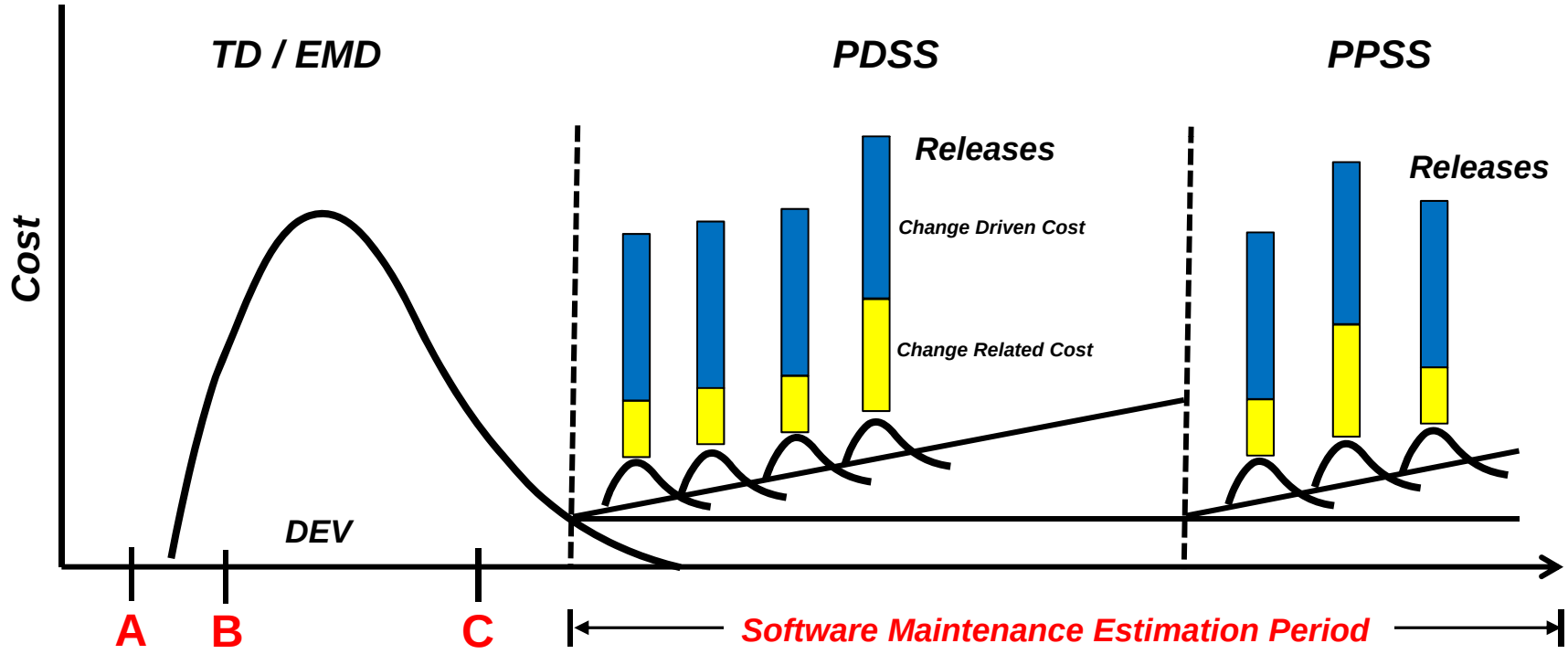
- ***Create a mechanism for identifying and evaluating software maintenance program risks at all pertinent decision levels***
- ***Identify those risks that directly impact program software maintenance cost uncertainty***
- ***Develop a methodology for integrating risk-uncertainty information into program software maintenance life cycle cost estimates***

Program Software Maintenance Release Profile



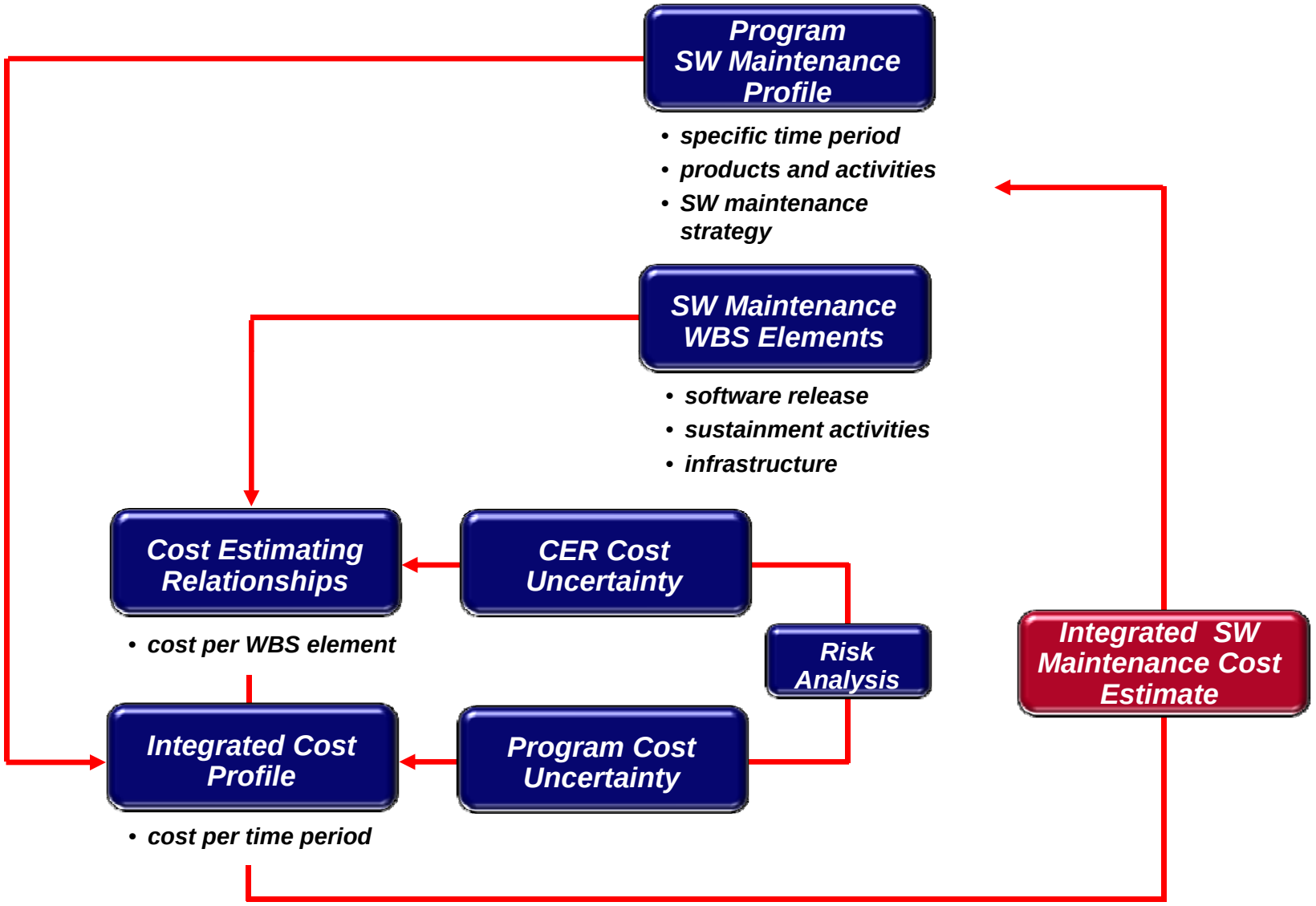
Cycles are different - platform dependent
User needs drive release content

Notional Software Maintenance Life-Cycle Cost Allocation Model



Program Information Requirements

- ***System and software maintenance strategy***
- ***Included products and activities***
- ***Software release profile - PDSS/PPSS***
  - ***Release type and schedule (high level)***
  - ***Release profile stability***
  - ***User base - deployment strategy***
  - ***Configuration information***
  - ***Performing organizations***
  - ***Projected operational environment***
  - ***Program and software risk analysis***
  - ***Historical profile information (later)***



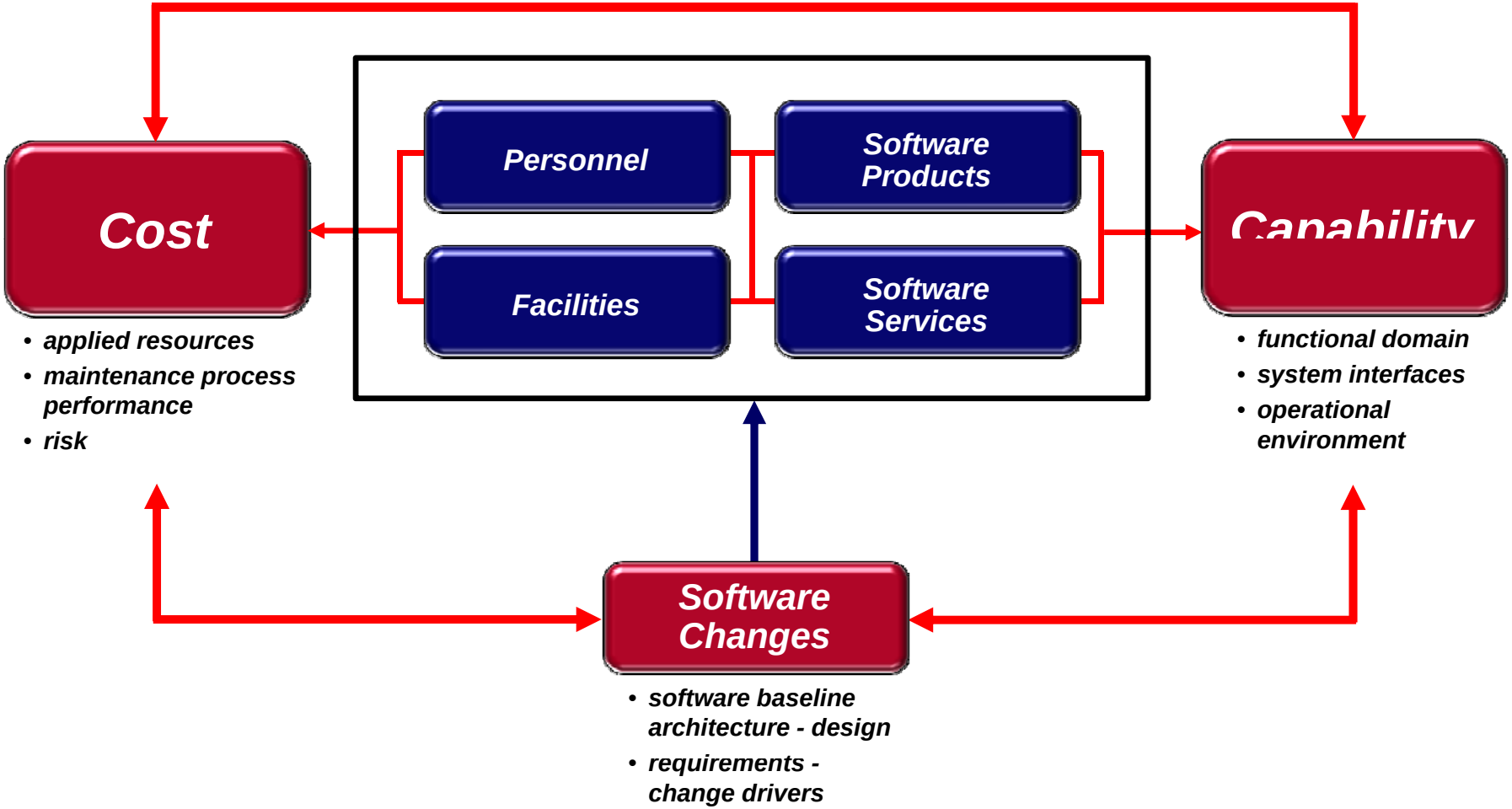
Long Term Objective

AH-64D Longbow "Night Fury"
10,000 Flying Hours Reached on 28 June 2011

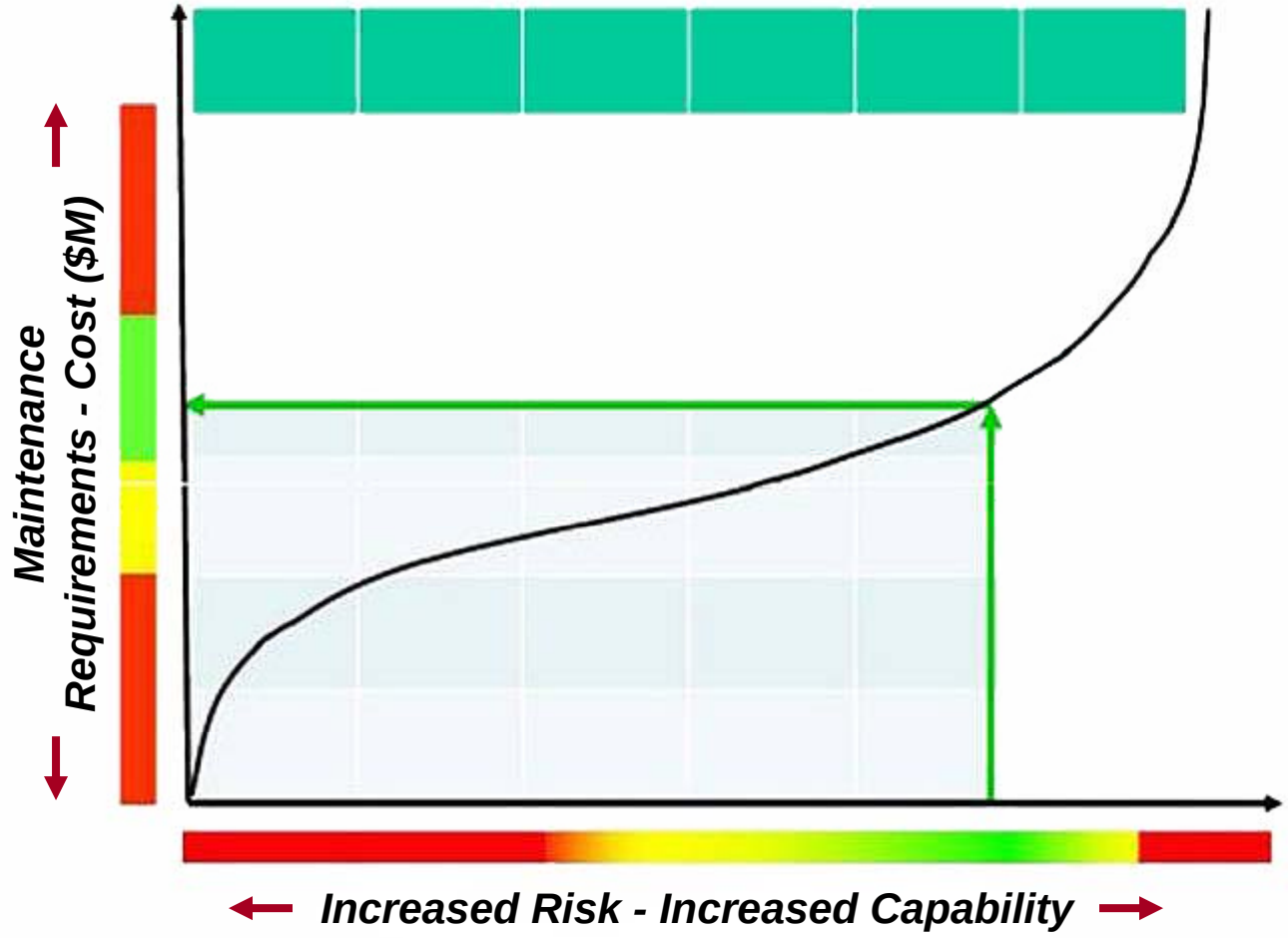
340,000 hours of maintenance by hundreds of weapons technicians in conjunction with countless hours of repairs and inspections performed by avionics, electrical and environmental, engine, fuels and structural personnel

If there was a 20% cut on avionics software maintenance, what would be the mission impact?

Software Maintenance Cost-Capability Framework



How much should software maintenance cost?



"It's All About the Money", Dr. Chien Huo, CAPE, November 2011

Summary

- ***The projected Army operational and economic environment places an increasing emphasis on the performance of software maintenance and sustaining engineering efforts***
- ***Accurate software maintenance life cycle cost estimates are critical to ensuring that objective resource information is available to program decision makers***
- ***The estimation methodology must be configured to address the unique characteristics and projected sustainment profile of each program***

Contact Information

Jim Judy
U.S. Army Office of the Deputy Assistant Secretary of the Army for Cost & Economics (ODASA-CE)
5701 21st Street
Building 216
Fort Belvoir, VA 22060-5546
(703) 697-1612
james.m.judy.civ@mail.mil

Cheryl Jones
U.S. Army RDECOM - ARDEC
Quality Engineering & System Assurance
RDAR-QES-A
Building 62
Picatinny Arsenal, NJ 07806-5000
(973) 724-2644
cheryl.jones5@us.army.mil

Jeramia Poland
U.S. Army ODASA-CE
5701 21st Street
Building 216
Fort Belvoir, VA 22060-5546
(703) 697-1640
Jeramia.poland.civ@mail.mil

Dr. Brad Clark
Software Metrics, Inc.
4345 High Ridge Road
Haymarket, Va. 20169
(703) 754-0115
brad@software-metrics.com

Additional Information

Army Software Maintenance Study

- ***Sponsored by U.S. Army Office of the Deputy Assistant Secretary of the Army for Cost & Economics (ODASA-CE)***
- ***Collaborative environment - diverse perspectives***
- ***Army, Air Force, Navy, OSD, Industry participation***
- ***Initial focus on ACAT-1 weapons system software maintenance costs***
- ***Technical approach based on measurement and estimation best practices***

Software Maintenance Product Sizing

- *Overall concept of "E size" for maintenance products*
- *E Size for types of changes (different drivers)*
- *Composite E Size for different types of releases*
- *Concept - profile the program maintenance strategy based on numbers and types of releases (variable costs) - add semi-fixed costs related to supporting the fielded software (costs not based on amount of work to construct a release)*
- *Variable costs driven by type and number of product changes*
- *Semi-fixed costs driven by number of deployed platforms, domain, operating tempo, etc.*
- *Similar to software development sizing "E Sloc" but focused on change drivers/release content and support scope*

Software Maintenance Change Drivers

- *The factors that cause the changes that are made to an operational software baseline*
- *Software changes characterized by:*
  - *source of change*
  - *type of change*
  - *impact of change (scope, complexity, etc.)*
  - *priority of implementation*
- *Change drivers can impact the operational software configuration, associated sustaining engineering activities, and the implemented maintenance infrastructure*
- *Most change requirements are allocated to planned releases per time period*
- *Deferred change requirements are defined as "backlog"*
- *Drivers with different characteristics are costed differently*

Software Change Drivers

External Drivers

- *Operational User*
  - *Functional modifications*
  - *Functional additions*
  - *Functional deletions*
- *Stakeholder*
  - *Threat*
  - *Mission doctrine*
  - *System interoperability*
  - *External testing/IV&V*
  - *External audits*
- *Mandate*
  - *Legal/Regulatory/Policy*
- *Technology*
  - *Technology obsolescence*
  - *Infrastructure changes*

Internal Drivers

- *Legacy*
  - *Technical debt*
  - *Deferred functionality*
- *Maintenance Operations*
  - *Maintainer skill set*

Types of Changes

- *Corrective (identified defects)*
- *Preventive (latent faults)*
- *Perfective (functional enhancements)*
- *Adaptive (new HW/SW environments)*