Software and IT-CAST Proceedings

22-24 August 2016

Lockheed Martin Global Vision Center

2121 Crystal Drive

Arlington, VA 22202

Table of Contents

OVERVIEW

The Naval Center for Cost Analysis and the National Geospatial-Intelligence Agency presents the Software and Information Technology Cost Analysis Solutions Team (Software and IT CAST) meeting from August 22-24, 2016 at the Lockheed Martin Global Vision Center in Crystal City, Virginia. This meeting is organized with the support of US Army ARDEC, Lockheed Martin, and DOD cost agencies.

The Software and IT-CAST meeting is a venue to build coalitions with government and industry, to exchange cost data, share lessons learned, and establish best practices concerning software and information technology cost estimation. Topics of interest include:

- Software cost estimation
- Software schedule estimation
- Information Technology (IT) cost estimation
- Cost Data Collection and Analysis Best Practices
- Functional size measurements
- Early phase software and IT cost estimation
- IT Cost Measures and Benchmarks
- Measurements for agile or other developmental approaches
- Measurements for software maintenance and sustainment
- Measurements for cloud computing services - SaaS, PaaS, IaaS
- Measurements for IT help desk and support
- Measurements for data center and network consolidation

The program includes presentations, workshops, and contractor one-on-one discussions. Presentations and workshops are opened to all attendees. Contractor one-on-one discussions are restricted to federal employees who have registered.

COMMITTEE

Chair:
Wilson Rosa (NCCA)
Vjosa Dreshaj (NGA)

Coordination:
Corey Boone (NCCA)
Corrine Krause (NCCA)

Venue:
Gregory Nieman (Lockheed Martin)

Steering Committee:
Corinne Wallshein (NCCA)
Richard Mabe (AFCAA)
Andrew Murray (NGA)

Software and IT-CAST Agenda

22-24 August 2016

Lockheed Martin Global Vision Center

2121 Crystal Drive

Arlington, VA 22202

Monday, August 22, 2016 – Contractor Discussions (Restricted)			
0800 – 0830	**Registration**		
0830 – 1130	**Oracle Enterprise** One-on-One	Dr. Wilson Rosa (NCCA) Mr. Rizwan Jaka (Oracle)	Auditorium
1130 – 1300	Lunch		
1300 – 1445	**SAP** One-on-One	Dr. Corinne Wallshein (NCCA) Mr. Joe Duffy (SAP)	Auditorium
1445 – 1500	**Break**		
1500 – 1700	**General Dynamics** One-on-One	Mr. Richard Mabe (AFCAA) Mr. Steve Workman (General Dynamics)	Auditorium

Tuesday, August 23, 2016 – General Session (Open to All)			
0730 – 0800	**Registration**		
0800 – 0810	Opening Remarks	Dr. Wilson Rosa (NCCA)	Auditorium
0810 – 0840	Keynote Address	Honorable Dr. Jamie M. Morin (DCAPE)	Auditorium
0840 – 0910	Recognizing and Shaping Organizational Culture and Acquisition Strategy towards Optimal ERP Program Cost Profile	Mr. Mike Lennon (SAP)	Auditorium
0910 – 0940	New Approach for Sizing and Estimating Agile Software	Mr. Blaze Smallwood (Booz-Allen-Hamilton)	Auditorium
0940 – 0950	**Break**		
0950 – 1020	Cost Assessment Data Enterprise (CADE) Initiatives	Ms. Bess Dopkeen (OSD CAPE)	Auditorium
1020 – 1050	Software Resource Data Reports: Development and Maintenance	Mr. Richard Mabe (AFCAA)	
1050 – 1120	Government-Wide CDRL: Agile Software Metric Data Collection	Mr. William Plummer and Mr. Jeremiah Hayden (SPAWAR 1.6), Mr. Omar Mahmoud (Cask)	Auditorium
1120 – 1150	IT Cloud Services Cost Measures	Mr. Chris Harrell, Mr. Justin Snyder (VMware)	Auditorium
1150 – 1200	COCOMO III Workshop Overview	Dr. Barry Boehm and Dr. Brad Clark (USC)	Auditorium
1200 – 1300	**Lunch**		
1300 - 1630	COCOMO III Workshop	Dr. Barry Boehm and Dr. Brad Clark (USC)	Second Floor GVC-A

Tuesday, August 23, 2016 – Contractor Discussions (Restricted)			

Software and IT-CAST Agenda

22-24 August 2016

Lockheed Martin Global Vision Center

2121 Crystal Drive

Arlington, VA 22202

1300 – 1450	**VMware** One-on-One	Ms. Vjosa Dreshaj (NGA) Dr. Carol Traynor (VMware)	Auditorium
1450 – 1500	**Break**		
1500 – 1650	**Microsoft** One-On-One	Ms. Haset Gebre-Mariam (NCCA) Mr. Robert Miller (Microsoft)	Auditorium
Wednesday, August 24, 2016 – General Session (Open to All)			
0800 – 0830	**Networking**		
0830 – 0840	Opening Remarks	Ms. Jennifer Rose (NGA)	Auditorium
0840 – 0910	Keynote Address	Dr. Troy E. Meink (ODNI SRA)	Auditorium
0910 – 0940	Exploring DoD Software Growth by Contract Type and CMMI Level	Mr. Nick Lanham, Dr. Corinne Wallshein (NCCA)	Auditorium
0940 – 1010	Risk Analysis: Best Practices using the Space Mission Catalog	Ms. Arlene Minkiewicz (PRICE Systems)	Auditorium
1010 – 1020	**Break**		
1020 – 1050	*Using Functional Size and Source Code to estimate ERP and Cloud Based Big Data Analytics*	Mr. David Seaver (NSA)	Auditorium
1050 – 1120	NRO CAAG Agile Software Development Practices	Ms. Michelle Jones (Booz-Allen)	Auditorium
1120 - 1150	Software Maintenance Cost Estimation	Ms. Cheryl Jones (Army ARDEC)	Auditorium
1150 - 1200	**COSYSMO 3** Workshop Overview	Dr. Barry Boehm, Mr. Jim Alstad (USC)	Auditorium
1200 – 1300	**Lunch**		
1300 – 1645	**COSYSMO 3** Workshop	Dr. Barry Boehm, Mr. Jim Alstad (USC)	Second Floor GVC-A
Wednesday, August 24, 2016 – Contractor Discussions (Restricted)			
1300 – 1545	**Northrop Grumman Aerospace Systems** One-on-One	Mr. Richard Mabe (AFCAA) Mr. Steve Huniu (Northrop Grumman)	Auditorium
1545 – 1600	**Break**		
1600 – 1715	**Lockheed Martin** One-on-One	Dr. Wilson Rosa (NCCA) Mr. George Barbic (Lockheed Martin)	Auditorium

Opening Remarks and Keynote

Dr. Jamie M. Morin

Director

Office of the Secretary of Defense/Cost Assessment and Program Evaluation

Jamie Morin was confirmed by the Senate as the second Director of Cost Assessment and Program Evaluation for the Department of Defense on June 25, 2014. As director, he leads an organization responsible for analyzing and evaluating the Department's plans, programs, and budgets in relation to U.S. defense objectives, projected threats, allied contributions, estimated costs, and resource constraints. The CAPE organization continues the heritage of the Systems Analysis office created by Secretary Robert McNamara and later renamed as Program Analysis and Evaluation. To support better defense decision making, CAPE develops analytical tools and methods for analyzing national security planning and the allocation of resources. The CAPE role in ensuring that the costs of defense programs are properly estimated and presented accurately was enhanced by the Weapons Systems Acquisition Reform Act of 2009.

Prior to joining CAPE, Morin served for five years as the Assistant Secretary of the Air Force for Financial Management and Comptroller. As the Air Force's chief financial officer, he was the principal advisor to the Secretary and Chief of Staff of the Air Force on financial matters, responsible for the financial and analytical services necessary for the effective and efficient use of Air Force resources. This included directing the development of the Air Force budget, overseeing the Air Force Cost Analysis Agency and conducting Air Force accounting and finance operations.

From July 3, 2012 until April 29, 2013, he was appointed by the President as Acting Under Secretary of the Air Force, during which time he served as the service’s chief management officer, senior energy official, chair of the Air Force Space Board, and acting Secretary of the Air Force during absences of the Secretary.

From 2003 until 2009, Morin was a member of the professional staff of the U.S. Senate Committee on the Budget, serving as the committee’s lead analyst for the defense, intelligence, and foreign affairs budgets, responsible for drafting relevant sections of the congressional budget resolution and advising the Senate on enforcement of budget rules.

Earlier in his career, Morin served in the Office of the Under Secretary of Defense for Policy and as an economic development strategist with the firm J.E. Austin Associates. His academic research focused on U.S. national security policy, particularly the role of Congress in defense budgeting and policy making. He held fellowships at the University of Virginia’s Miller Center for Public Affairs and at the Center for Strategic and Budgetary Assessments, where he conducted research for the Pentagon’s Office of Net Assessment. He also served as a policy advisor on President Obama’s defense transition team.

Dr. Troy E. Meink

Assistant Director

National Intelligence for Systems and Resource Analyses (ADNI/SRA)

Prior to his ODNI appointment, Dr. Meink was the Deputy Under Secretary of the Air Force for Space and the Director, Executive Agent for Space Staff, Washington, D.C. He provided the principal support to the Under Secretary's role as the Headquarters U.S. Air Force focal point for space matters and in coordinating activities across the Air Force space enterprise.

Dr. Meink is from Lemmon, S.D., and entered the Air Force in 1988 through the ROTC program at South Dakota State University. His assignments have included operations and training, systems engineering, research and development, and program management of major defense acquisition programs. Dr. Meink began his career as an Air Force Navigator and then a lead test engineer for the design and evaluation of ballistic missile test vehicles for the Missile Defense Agency resulting in two successful launch campaigns. As a rated officer, he completed 100 sorties including eight combat and 29 combat support missions in support of operations Desert Shield, Desert Storm, and Provide Comfort.

As an Air Force civilian, Dr. Meink managed multiple next generation joint research and development programs transitioning global space capabilities, optical sensors, and advanced structures into DoD operations. He subsequently led multiple communications organizations within the Air Force and the Office of the Assistant Secretary of Defense, Networks and Information Integration. Prior to his assignment as the Deputy Under Secretary of the Air Force for Space and the Director, Executive Agent for Space Staff, he was the Director, Signals Intelligence Systems Acquisition at National Reconnaissance Office.

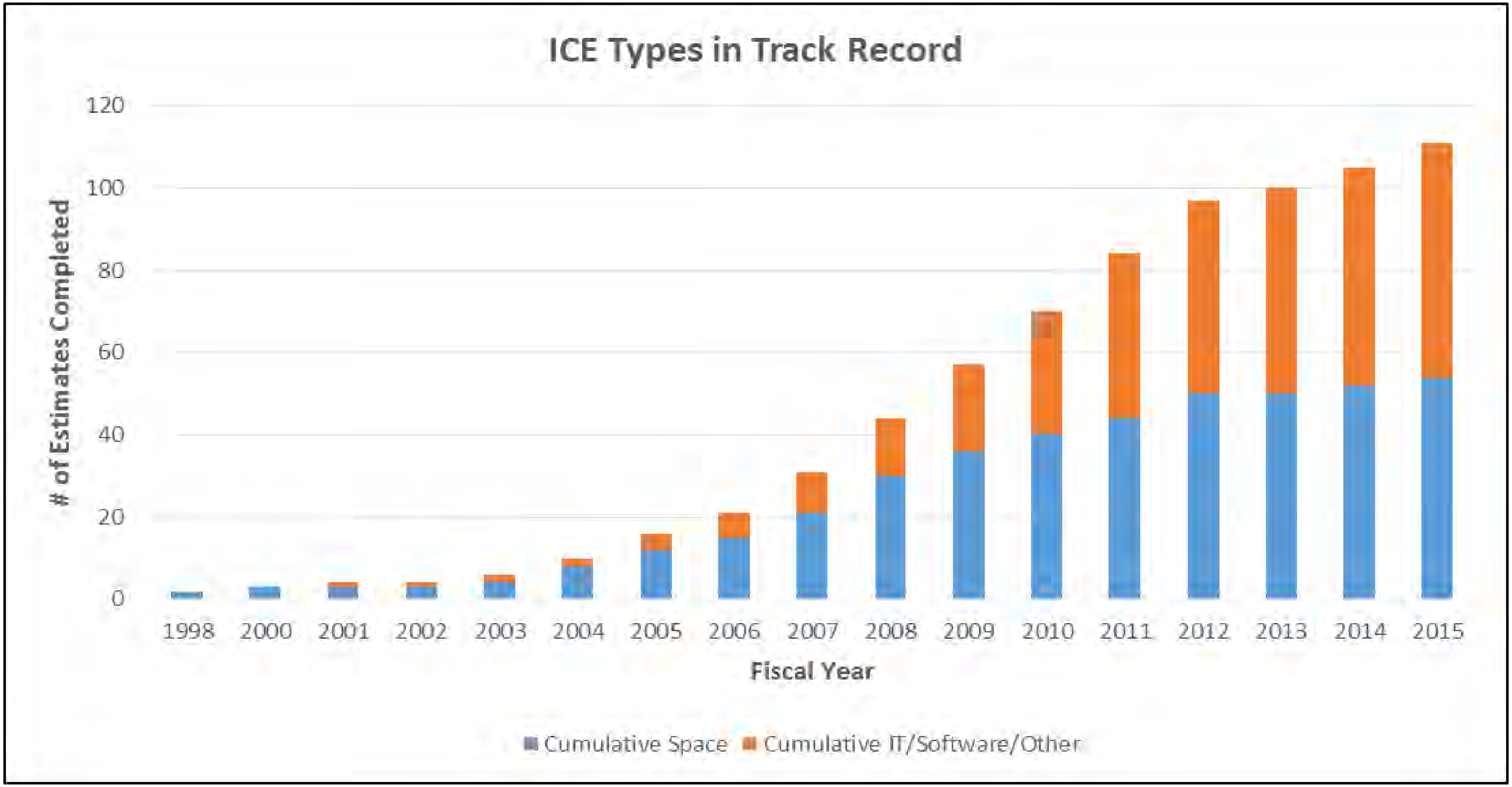
Dr. Meink has authored 20 articles in professional journals and conference publications, has been awarded three patents, and designed, built, and flown two experimental aircraft.

OFFICE OF THE DIRECTOR OF NATIONAL INTELLIGENCE

IT CAST Conference

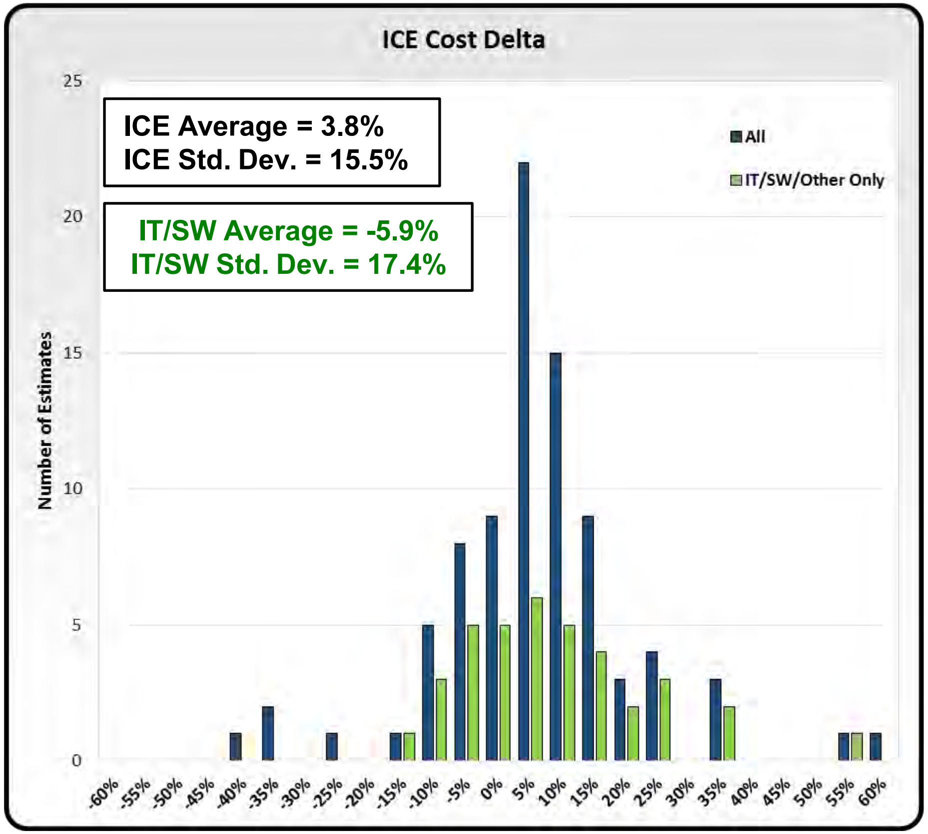
Systems and Resource Analyses

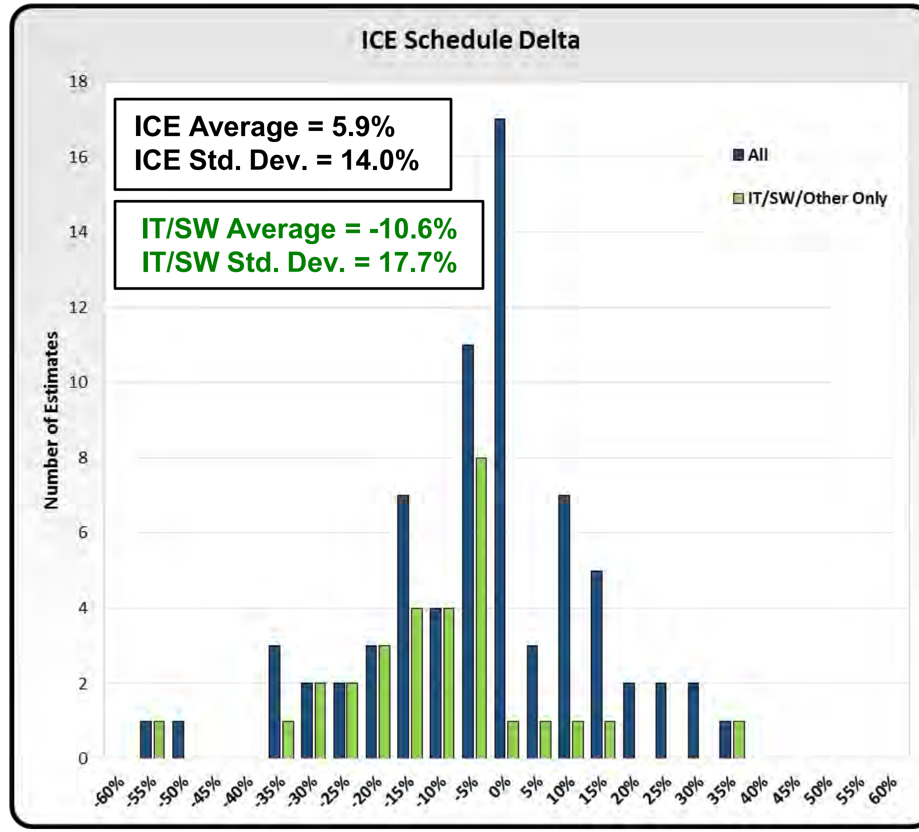
Independent Cost Estimates By Type



ODNI is performing ICEs on an increasing number of IT and Software programs

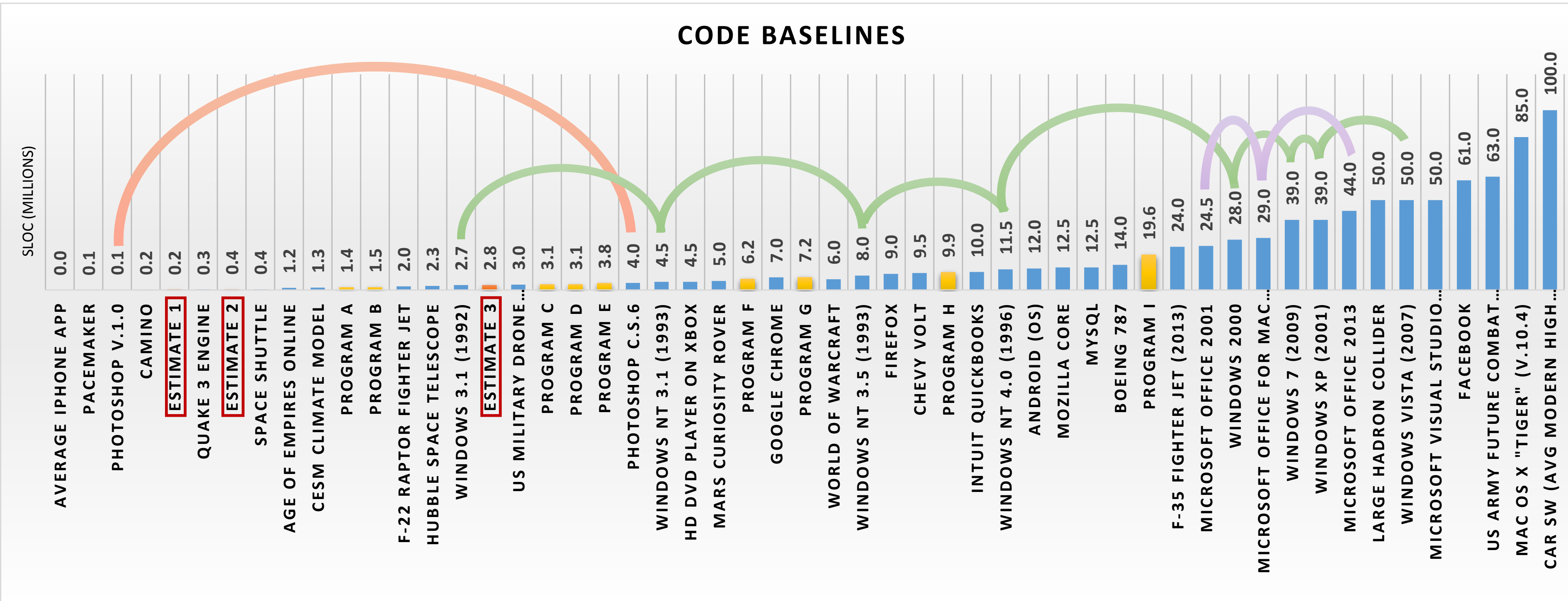
ICE Deltas: All versus IT/SW/Other





IT and SW programs are a focus area to collect more data and improve our estimating capabilities

Code Baselines for IC Programs and Other Software

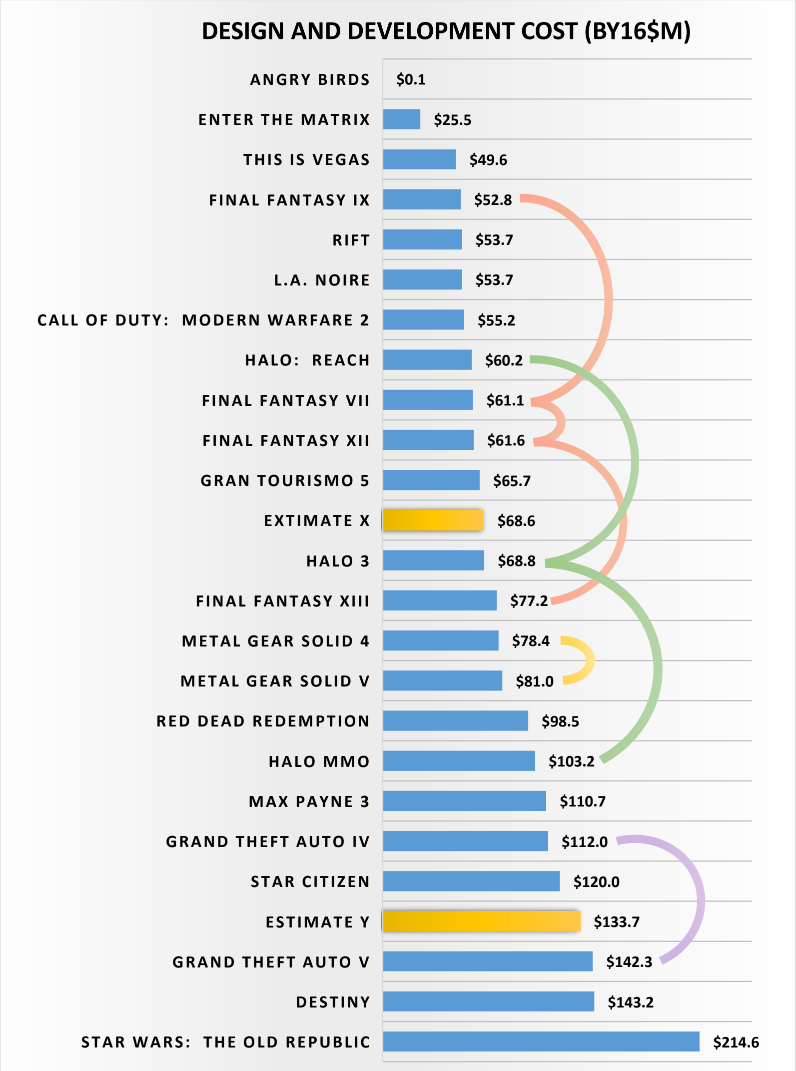


Key

- = Variety of Software
- = IC Program Actuals
- = Recent ICEs

Source: How Many Lines of Code is Your Favorite App? http://www.fastcodeesign.com/3021256/infographic-of-the-day

Design and Development Cost Comparison



Key

- = Video Games
- = Recent IC Estimates

Multiple sources; available upon request

Final Thoughts

- We understand space systems costs very well
- We need the same understanding for software & IT
- Increased data collection and sharing is critical!

The IT CAST forum is a great opportunity to discuss IT and software issues, exchange best practices, and create ideas to drive innovative IT and software analyses

Recognizing and Shaping Organizational Culture and Acquisition Strategies toward an Optimal ERP Program Cost Profile

An SAP Perspective

Agenda / Story Flow

Benefits of an Educated Workforce

1. What do we spend on?
2. "Things that are inherently governmental"
3. Best Practices for ERP Implementations

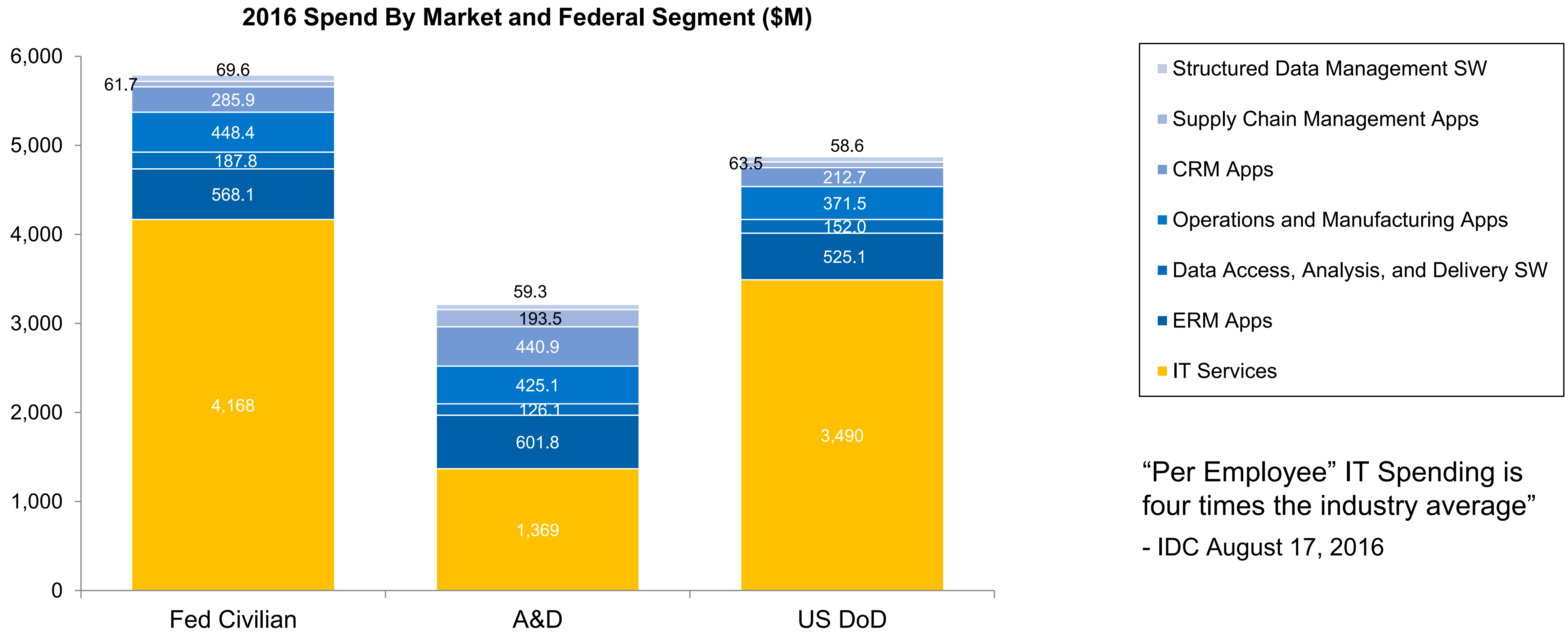
Beginning with the End in Mind

4. Plan and Execute Iteratively
5. Align Acquisition Strategy to Support Overall Plan
6. The Role of the Software Provider

What about the Cloud?

7. Why the Cloud?
8. Aligning the Dream of Cloud with the Realities of DoD

Federal IT Spend Software and IT Services

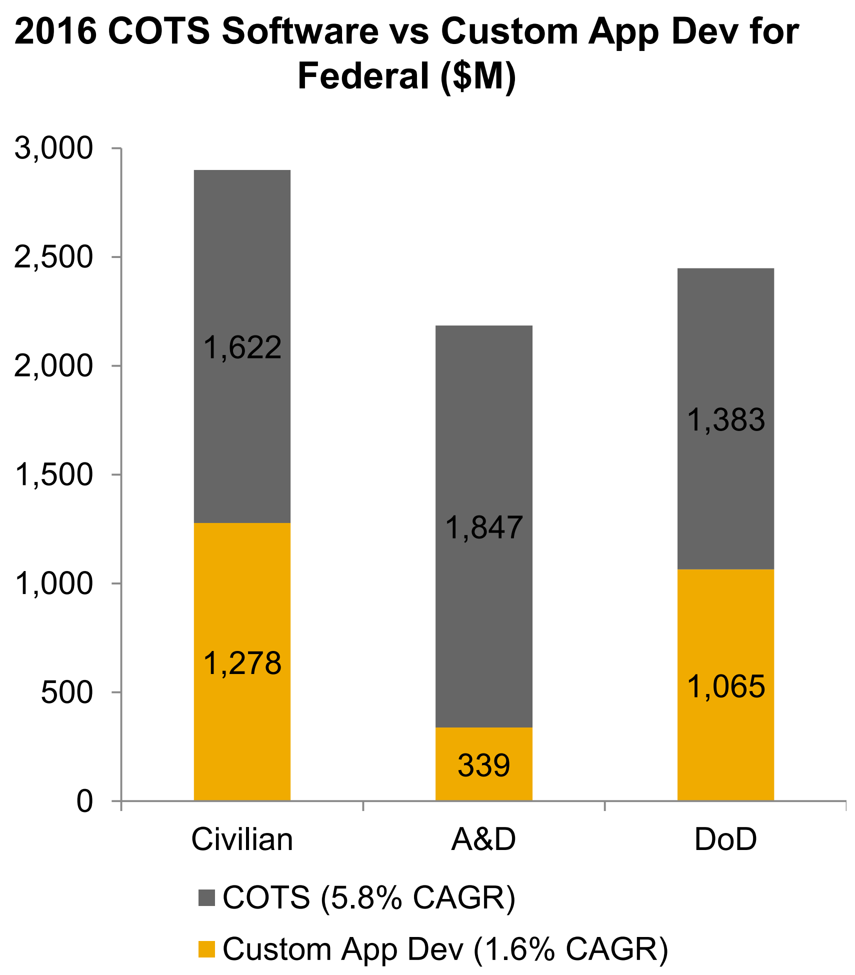


"Per Employee" IT Spending is four times the industry average"

- IDC August 17, 2016

Source: IDC April 2016

Federal IT Spend Software and IT Services Cont'd

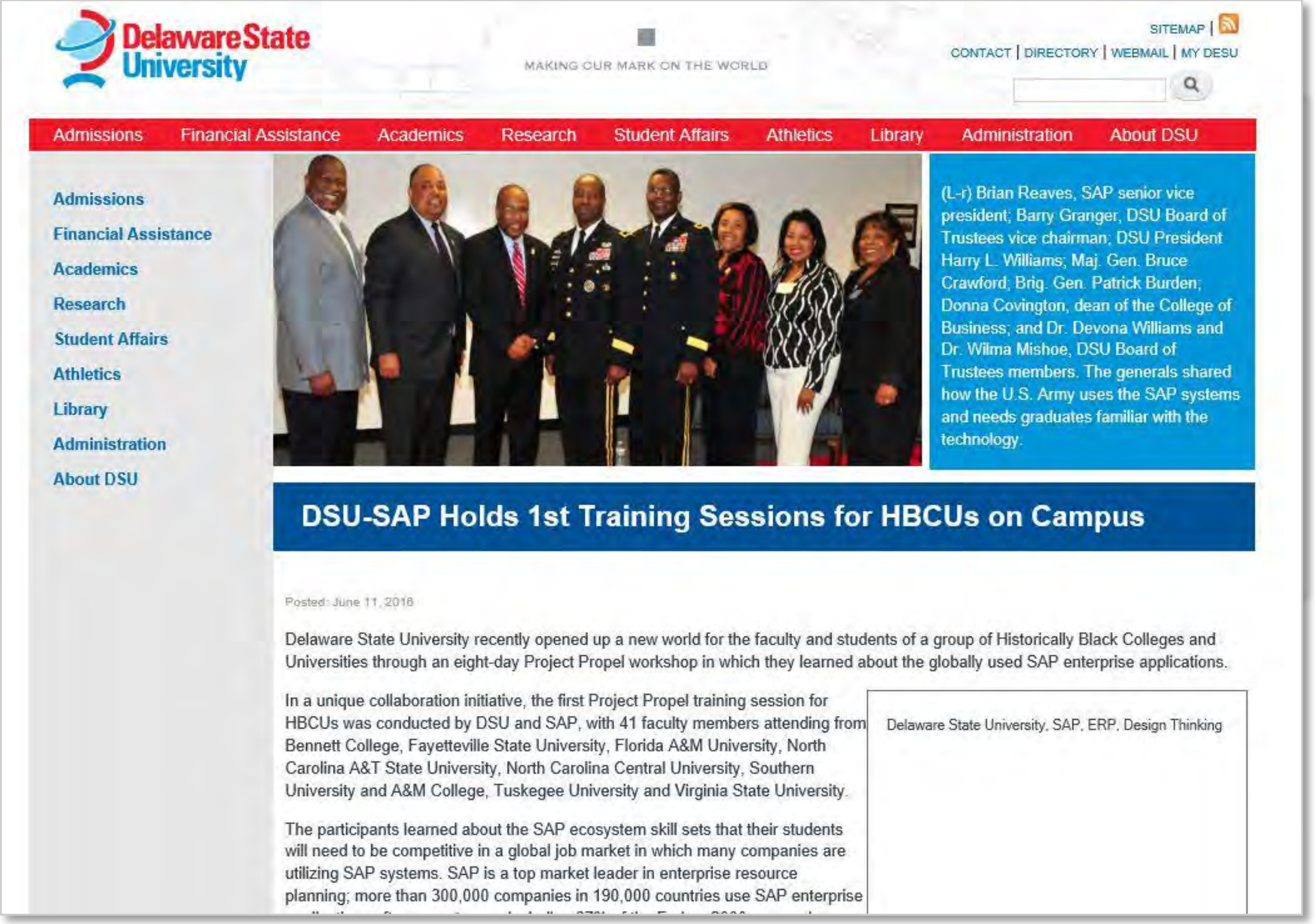


Source: IDC April 2016

- **Most government agencies have a desire to invest in commercial-off-the-shelf (COTS) software**
- **Most find that they need to customize their solutions anyway**
  - Connections to legacy systems
  - Workflow management
  - Application compatibilities
- **CAD excludes customization of COTS**
  - Additional large opportunity for customization of COTS which is significant in Federal
- **Standardization across agencies (and states) is having an impact on COTS for gov solutions. They are getting better**
- **Template based solutions allow more flexibility, and customization sometimes can be handled via configuration settings vs. full custom coding**
- **Government has a need for quick customization because of changing laws, reporting requirements, etc.**

Things That Are Inherently Governmental

- Control of the Requirements and Scope
- Establish a Center of Excellence (COE)
- Preparation of the Workforce

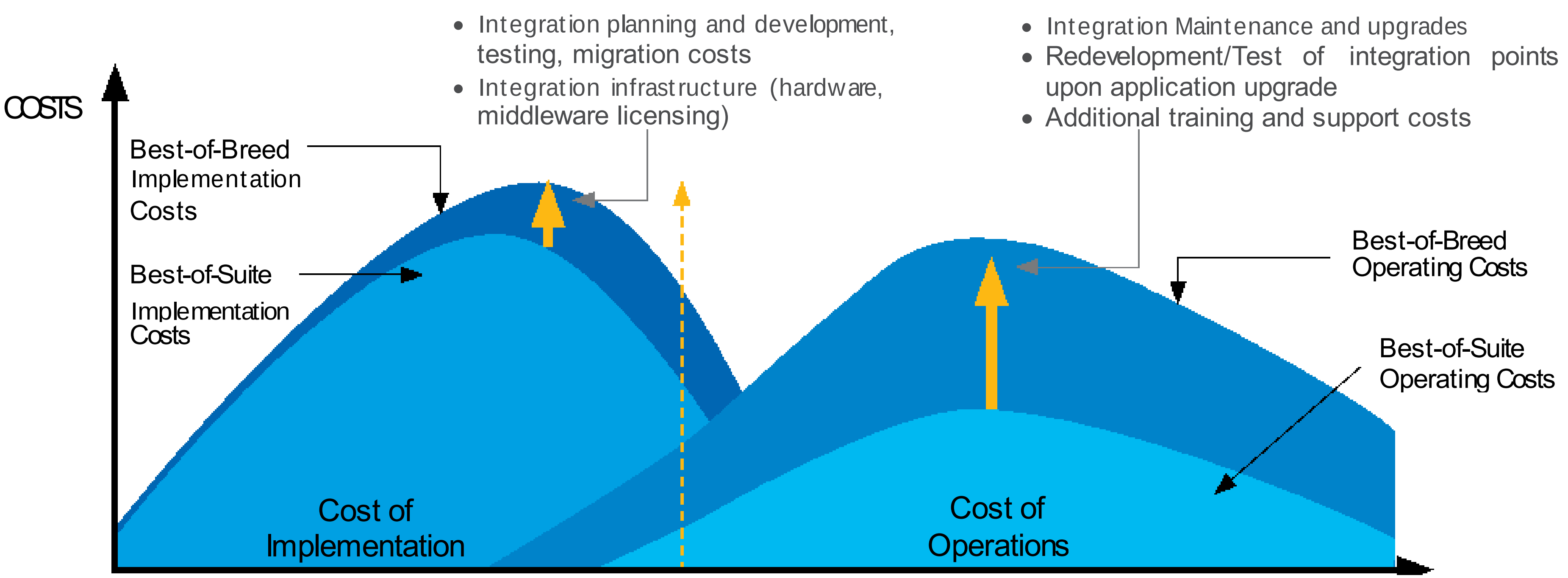


“An educated consumer is our best customer”– Sy Syms, Syms Corp.

SAP Digital Strategy

Forrester states the following:

"...in a situation where a large enterprise licenses and implements a single vendor solution consisting of an ERP package supplemented by one or more functional suites...If one of the functional suites is bought form a third party, the ... costs will be significantly higher. The most significant cost difference will be the integration cost, which can be little or nothing when a suite is purchased, but can be substantial when a third party tool must be integrated."



Best Practices for ERP Implementations

Adoption of Best Practices Varies at Each Stage

	Level 1	Level 2	Level 3	Level 4	Level 5
Project Management	Limited time and effort is spent during project start-up and blue printing phases	Some time is spent during project start-up to define goals and identify project team. But implementation strategy is not fully baked.	Some time and effort is spent during project start-up and blue printing phases, but the rigor is limited only to the 'important' functions and process areas	A lot of effort is invested in the scope document and defining the "as-is" and "to-be" processes, but they are still not finalized before starting the realization phase	Significant time and effort is invested during project start-up and blue printing phases to ensure successful foundation for project
Knowledge Transfer	Ad-hoc – basic standards exist in order to facilitate knowledge transfer	Some amount of participation from company during implementation to facilitate knowledge transfer	Company identifies people who should be receiving knowledge transfer, and incentivizes them to stay long-term with the company	Few strategies (like, train the trainer) are in place to drive knowledge transfer	Significant participation from company during implementation to drive knowledge transfer and reduce reliance on external consultants after go-live
Testing	A major portion of testing is done manually	Some amount of testing is automated in order to drive efficiencies	More than 50% of testing is automated	A significant portion of testing is automated in order to drive a higher quality test of system	Use of integrated portfolio of automation testing tools to minimize the use of manual testing, resulting in a more comprehensive test
Value Management	KPIs for projects were selected based on the business case objectives and strategic goals	Performance baselines are established for project KPIs prior to implementation	In blueprint design, business case objectives and critical process changes are clearly communicated and incorporated into design	Value checks are built into the blueprint and realization processes to monitor and assess the design compared to business case	KPIs are measured effectively after implementation
Governance, Portfolio and Risk Management	Lack of processes to manage risks along the entire implementation cycle	Ad-hoc - basic standards exist to manage risks during the implementation cycle	An informal risk management process exists in order to manage risks during the implementation cycle	A risk management process is defined as part of the implementation strategy in order to manage risks during the implementation cycle	A well defined risk identification and management process is put in place for managing risks along the entire implementation cycle

© COPYRIGHT SAP 2011 – Page 8 – This page is subject to the terms, conditions and assumptions contained in the Benchmarking Analysis. Information contained herein is not a guarantee of future results, performance or cost savings.

Low Maturity → High Maturity

Over 100 organizations took the survey. The study found that those that had a high maturity:

- Obtained 40% more value of their expected IT project value
- Had 64% lower project duration delays
- Had 71% lower project overruns

Plan and Execute Iteratively

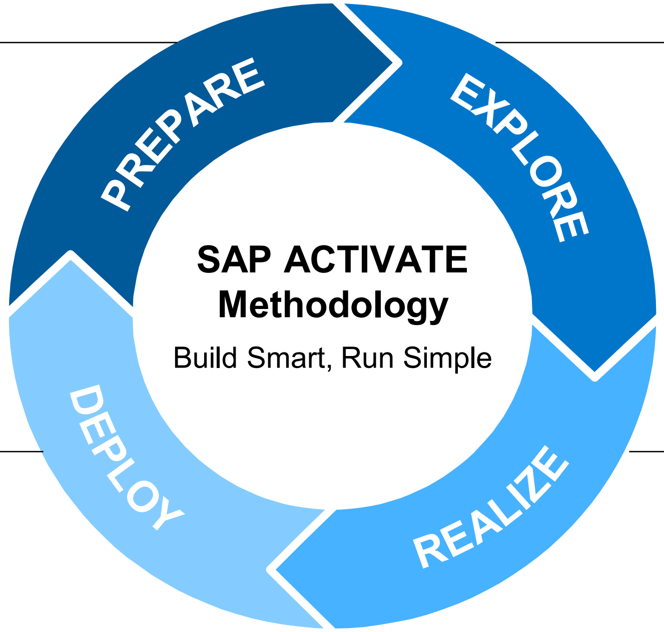
Roadmap –alignment with PoM

Waterfall vs. “Agile” Methodology

Phasing/’flow’ of personnel during lifecycle

Jump-start project with pre-assembled solution

Identify and validate delta requirements and gaps



Prototype, Build and Run solution leveraging Support Centers

Build customer solution in short, time-boxed sprints

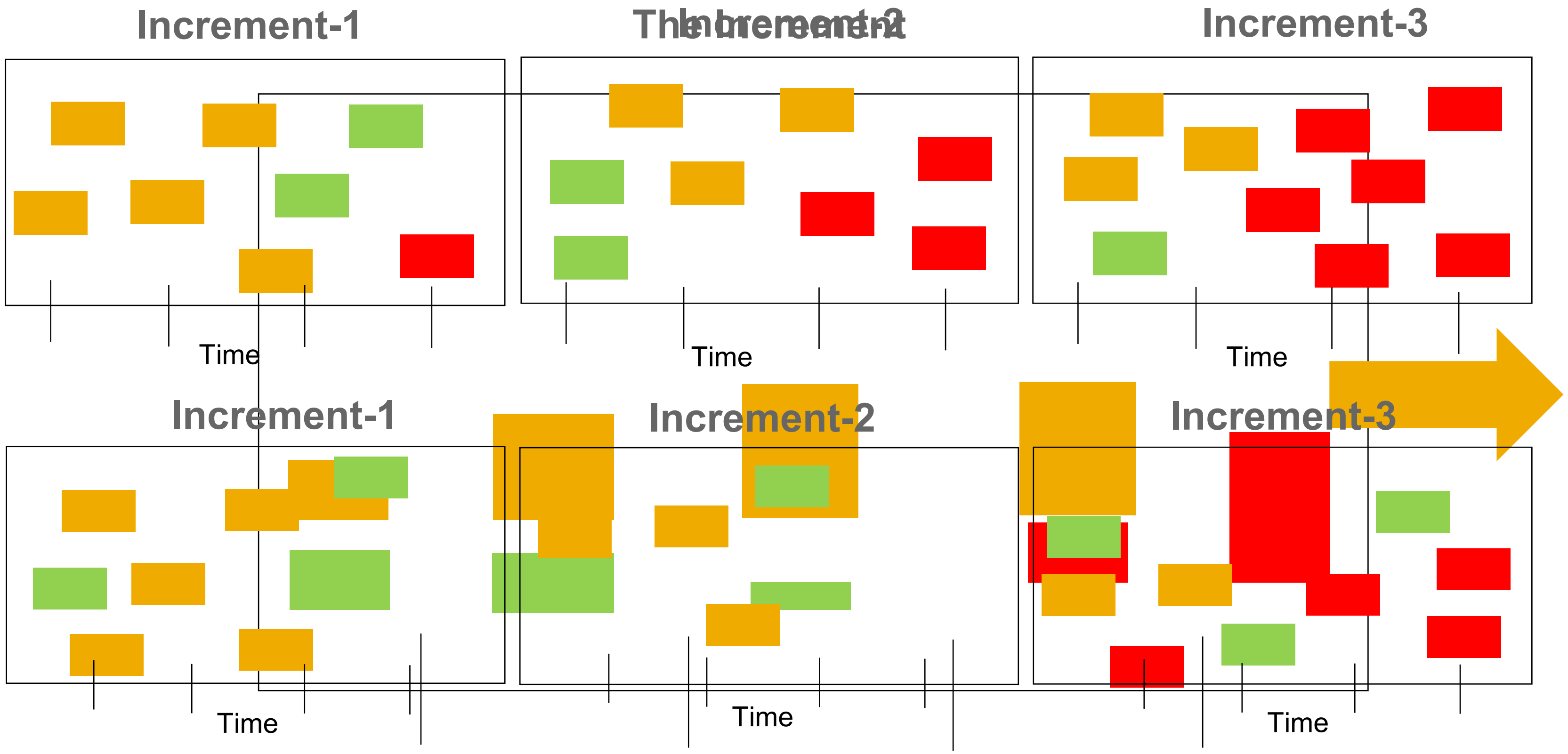
If I have 8 hrs to chop down a tree, I’d spend 6 sharpening my axe

Align Acquisition Strategy to Support Overall Plan

Preplanned Product Improvement (P^3I)

Current Approach

Recommended Approach



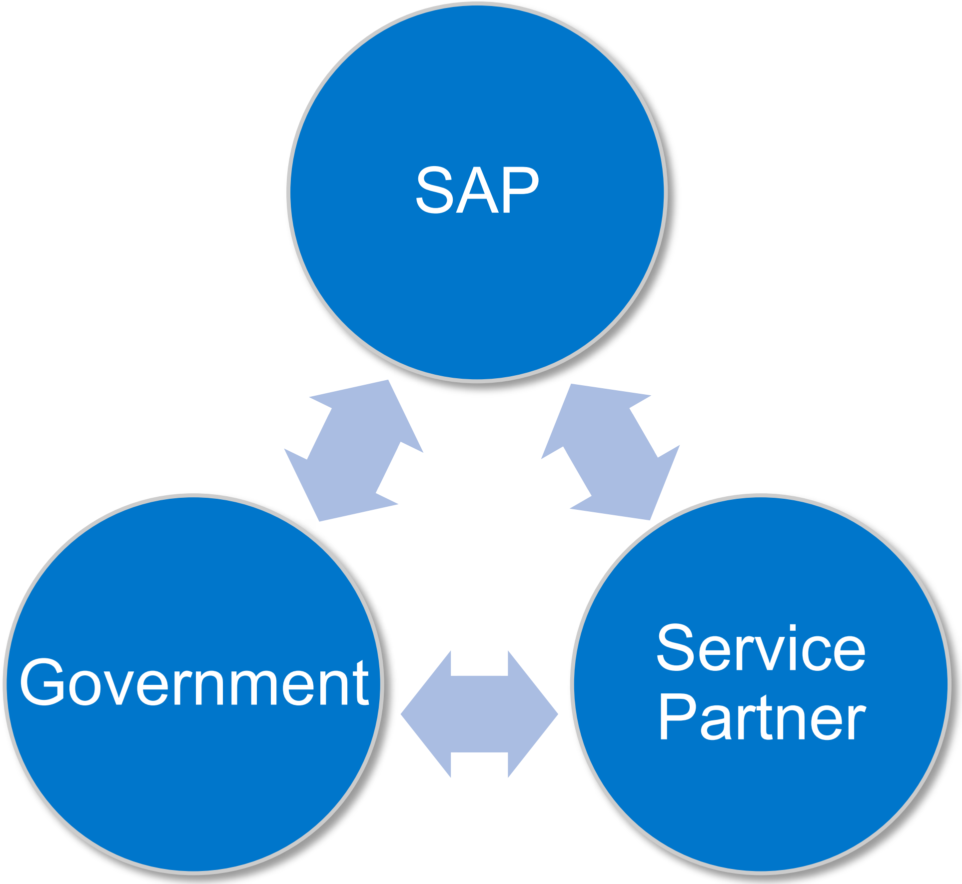
Procurement

Research

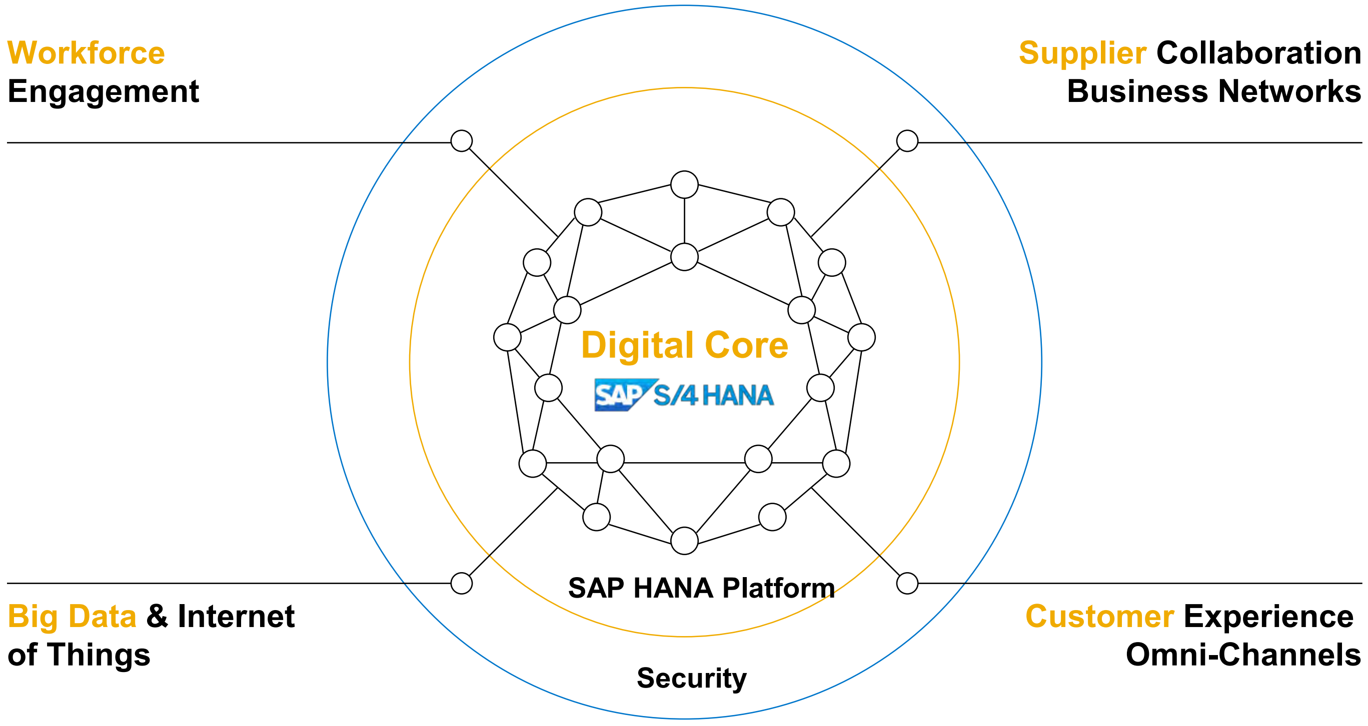
Operations & Maintenance

The Role of the Software Provider

- “Directional guidance” i.e. HANA/Cloud
- Access to the company/partners/providers and ‘birds of a feather’
- Events like Sapphire and TechEd
- Functional/Technical Architecture Services
- Share Industry Best Practices and Insight
- User Forums like DEIG and/or ASUG
- Help with Product Capabilities, and Implementation Best Practices



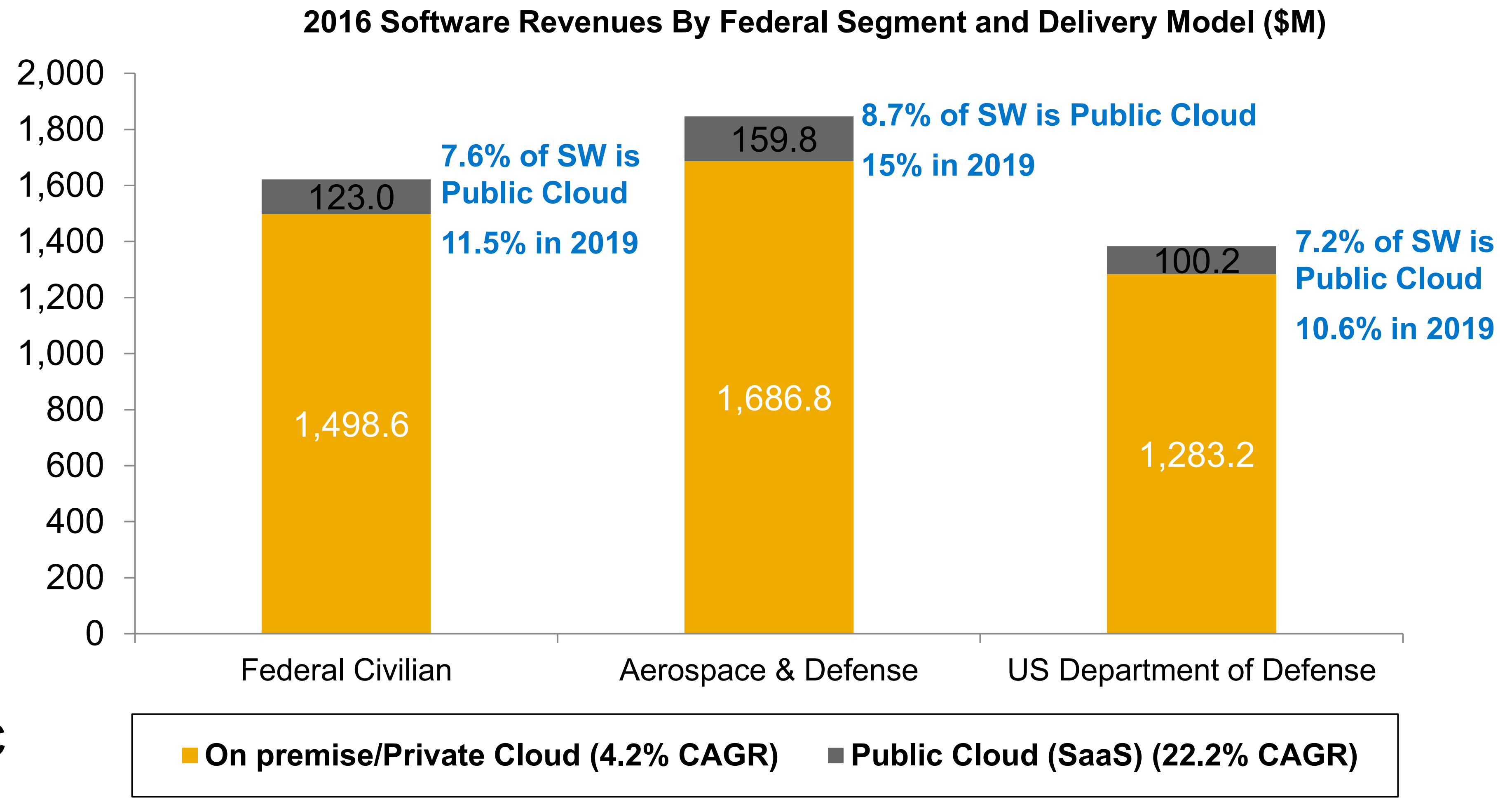
SAP Digital Strategy



Characteristics of the Digital Transformation journey

1. **Customers and employees** are hyper-connected, always on, with **seamless access anywhere and anytime**
2. **Cloud and hybrid cloud environments** have become the norm challenging traditional “protect the 4 walls” security approaches
3. **Digitally connected supply chains** are based on high trust and availability of all parties
4. The Internet of Things and Big Data bring **unprecedented data streams and volumes**
5. **Confidentiality, integrity and availability** of data is the basis for secure operations and trusted relationships

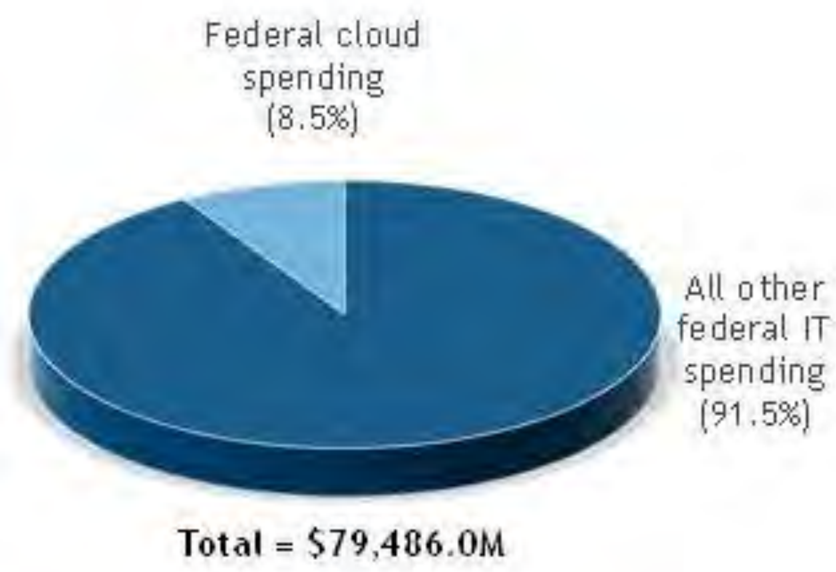
Public Cloud in Federal is a small portion of the software market in 2016 but fast growing at 22% CAGR



Source: IDC

About 8.5% of all federal IT spend is for “cloud” as termed by the government

Federal Fiscal 2016



Note: Shows entire Federal IT market, beyond SAP’s addressable market.

Source: IDC

About 8.5% of the federal government's IT spending, or $6.7 billion, in fiscal 2016 will go to cloud technologies – a marked increase from the 5% of IT spending that the government put toward cloud in fiscal 2015.

A further blurring of “cloud” may happen as more agencies obtain servers for their data centers which are remotely managed and updated by cloud providers, effectively blurring the lines between what is hosted and what is pure cloud.

US Federal Government classifies some types of shared services as “cloud” though it does not meet IDC’s cloud definition.

Such shared services could be private cloud for SaaS, PaaS or IaaS

Greater opportunity for IaaS vs SaaS in Federal due to overall lack of off the shelf packaged apps

“Shared Services Cloud” adds about 5 percentage points to Software Public Cloud proportions, which are ~7-8% share.

Clearly, lower price points for cloud services and the changing nature of cloud are helping to drive federal cloud consumption into new directions, resulting is substantial growth in some consumption patterns.

Why Organizations are Investing in the Cloud?

Cost and Consolidation	Improve Time of Delivery to Production	Improve Project Success Rates	Difficult Provisioning Innovation
Bottleneck in adapting to changes in business needs and requirements	An ever increasing need for faster time to value while simplifying the IT landscape	Pressure on IT and Business to reduce costs while still achieving business goals	Long lead times required for innovation

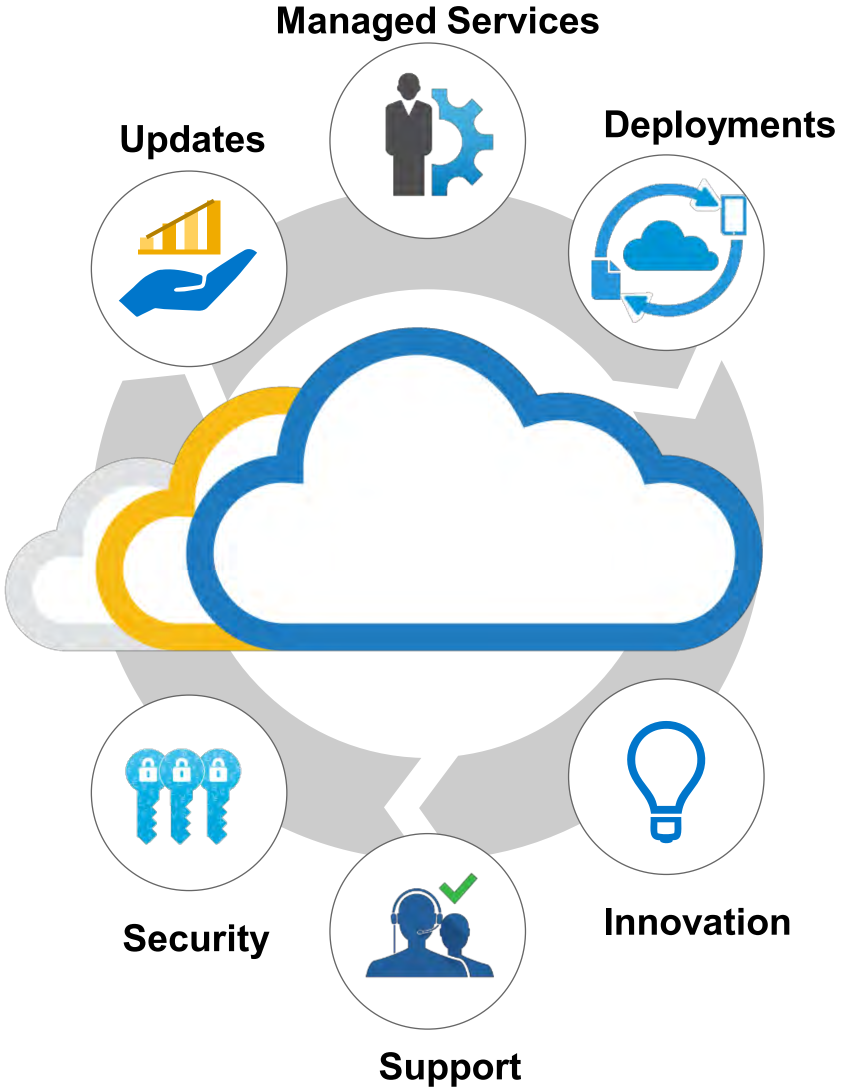
Pressure to ensure secure resources protecting your information

Cloud Adoption – Strategy for Success

Application

Information

Infrastructure



NS2 CLOUD
SAP Mission Critical Edge

Recommended Reading

1. Delivering Materiel Readiness- From “Blunt Force” Logistics to Enterprise Resource Planning; by Lieutenant General Larry Wyche, Deputy Commanding General, U.S. Army Materiel Command
2. Every Dollar Counts – Memorandum dated April 15, 2016 – Secretary of the Army, Washington, DC.
3. ASUG Implementation Services – Best Practices Survey- 2011
4. Article on ASC.ARMY.MIL COL Harry Culclasure, AESIP project manager and Mr. Thomas Neff (CHESS project leader). Topic: Acquisition strategy of six Army programs selecting a single enterprise resource planning solution…. (URL: http://asc.army.mil/web/news-better-to-best/)
5. Interview on Federal Radio – Lt. Col. Robert Williams, Product Manager for LMP and Joshua Call of Army Materiel Command. Topic: Army Logistics Modernization Program (LMP) Overview and Benefits
   – (URL: http://federalnewsradio.com/dod-reporters-notebook-jared-serbu/2016/07/army-says-logistics-system-saved-least-6-billion/)
6. “Per Employee" IT Spending: Federal Government is Nearly Four Times the Industry Average”, IDC August 17, 2016

Please write to Mr. Mike Lennon (Mike.Lennon@sap.com) if you want to receive electronic copies of documents per #1-3 above

Collaborative Requirements Scoring:
An Innovative Approach for Sizing Software Projects

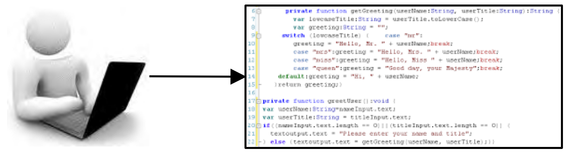
Blaze Smallwood
Software and IT-CAST 2016

Table Of Contents

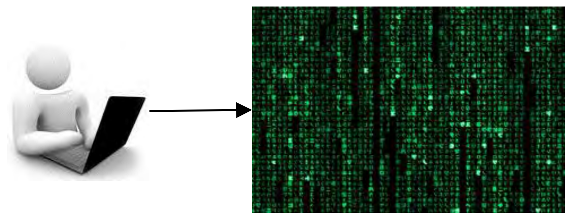
Table Of Contents

Existing software cost estimating techniques present various challenges, particularly when applying to modern systems with limited analogs and custom application requirements

- Estimating size using physical/functional measures is often non-intuitive to program engineers, particularly when estimating newer, lower-level requirements

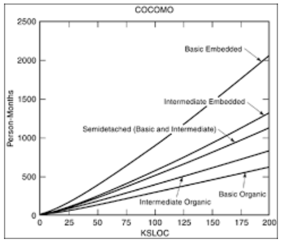


- Accounting for non/cross-functional and/or derived requirements can be challenging



- Explaining to PMs how estimates are derived based on physical/functional sizing can be difficult

- Estimates are typically difficult to break out at lower levels of detail, hindering trade space analysis



- When estimates are wrong, hard to diagnose why

Traditional methods work well for many programs, but not all; some could use an alternative approach based on current software practices

Table Of Contents

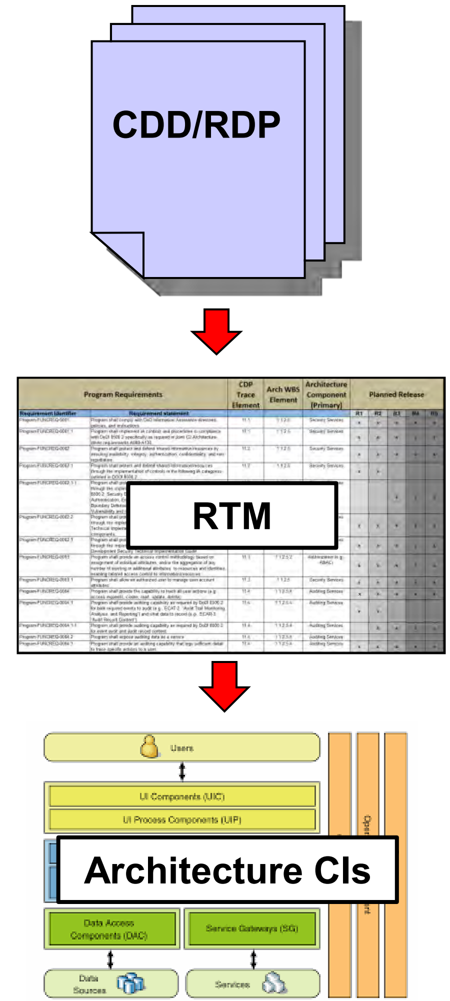
Table Of Contents

Step 1: Identify Appropriate Requirements to Size

Identifying appropriate requirements source and level of detail is essential to any sizing estimate

- Foundation would be functional/architectural requirements in program requirements document (e.g. Capability Definition Document (**CDD**) or Requirements Definition Package (**RDP**))
  - If **non-functional/cross-functional requirements** (architectural, usability, reliability, etc) are not captured in requirements document, they **can be identified and estimated during the sizing process**
- Requirements from CDDs or RDPs are often captured in a Requirements Traceability Matrix (**RTM**)
- Ideally, RTM would be used to map functional requirements to **components** in the intended **architecture**
  - Much more intuitive for SW engineers to estimate building components in an architecture

Ideal Requirements Flow



Best Practices for Requirements Identification

- Get agreement from estimate stakeholders (program manager, cost analyst, engineers) what requirements will be estimated and at what level
  - Need to identify the level of detail in requirements that fit time allocated to sizing and the appropriate units of measure (points, person-days, person-months)
- If documented requirements are all functional, work with engineers to identify non/cross-functional requirements/constraints before estimating session
- Identify trade space requirements up-front to inform later CAIV analysis
- Get Excel formats of requirements lists or matrices to help in building estimating template

Table Of Contents

Scoring sessions require upfront planning and coordination to be successful

- Identifying appropriate participants is key
  - "Scorers": Software engineering SMEs that will be providing estimates
    - A panel of four or five is ideal; minimum of three
    - Should be a mix of SMEs; at least a couple with experience in systems similar to the one being estimated, but independent SMEs add value, as well
  - Facilitator(s): Person who runs the meeting, records scores and assumptions, facilitates discussions, takes care of admin items
    - Ideally, two people – can be a lot for one person to handle
  - Other support SMEs: People knowledgeable about the program that can help scorers better understand requirements and constraints
    - Requirements analysts, test engineers, cyber experts, past users of similar system
- Coordinate schedules for participants and facilities early

Developing scoring materials and thinking through scoring process in advance will help the actual session go more smoothly

Step 2: Plan/Structure Scoring Session

- Develop scoring template based on requirements to be estimated

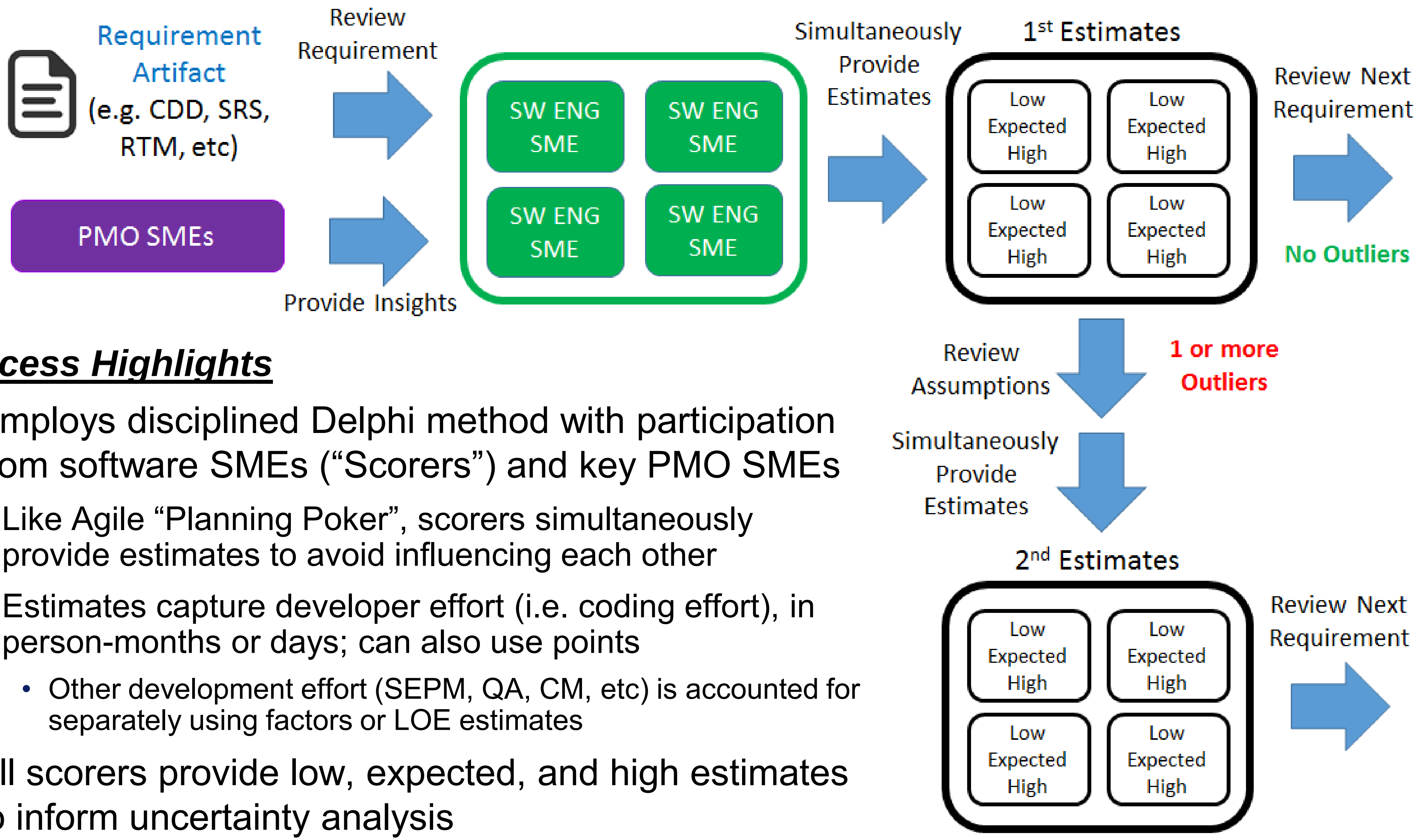
		Round 1									Round 2									
		Scorer 1			Scorer 2			Scorer 3			1			2			3			
Req #	Req. Description	Exp.	Low	High	Exp.	Low	High	Exp.	Low	High	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	Comments
1.1.1	The system shall enable the user to . . .																			
1.1.2	The system shall enable the user to . . .																			
1.2.1	The system shall enable the user to . . .																			
1.2.2	The system shall enable the user to . . .																			
1.2.3	The system shall enable the user to . . .																			
2.1.1	The system shall enable the user to . . .																			
2.1.2	The system shall enable the user to . . .																			
2.1.3	The system shall enable the user to . . .																			
[illegible]	[illegible]																			
X.X.X	The system shall enable the user to . . .																			

- Coordinate systems or materials needed to conduct scoring; session requires scorers to provide estimates anonymously and simultaneously
  - Decision analysis system, like Expert Choice
  - Agile Planning Poker app
  - Real-time collaboration tools, like Google Docs or Real Time Board
  - Screen sharing, like WebEx or Defense Collaboration Services (DCS)
  - White boards

Table Of Contents

Scoring process helps objectify subjective inputs and documents assumptions for each estimate

Step 3: Execute Scoring Session



Process Highlights

- Employs disciplined Delphi method with participation from software SMEs ("Scorers") and key PMO SMEs
  - Like Agile "Planning Poker", scorers simultaneously provide estimates to avoid influencing each other
  - Estimates capture developer effort (i.e. coding effort), in person-months or days; can also use points
    - Other development effort (SEPM, QA, CM, etc) is accounted for separately using factors or LOE estimates
- All scorers provide low, expected, and high estimates to inform uncertainty analysis

Best Practices for Executing Scoring Session

- Baseline all participants upfront on ground rules and assumptions
  - Ensure everyone is operating off of the same overarching technical assumptions
  - Agree on the scope of estimates to be provided; typically, developer effort from design to code unit test
  - Discussion before scores should be limited to technical assumptions only
    - No value statements -> “This is easy, should be minimal effort”
  - Second round is needed when one or more outlier scores
- Ensure everyone is clear on time constraints to stay on schedule
  - Calculate benchmark requirements to measure progress against time plan
- Capture all assumptions for each requirement in scoring spreadsheet; capture any other thoughts on white boards or smartboards & take pictures
- Use white board to capture “parking lot” items; could inform additional items to score, like cross-functional or derived requirements

Outputs of a successful scoring session

- Fully populated scoring template with all scores and documentation comments

		Round 1									Round 2			
		Scorer 1			Scorer 2			Scorer 3			1	2	3	
Req #	Req. Description	Exp.	Low	High	Exp.	Low	High	Exp.	Low	High				Comments
1.1.1	The system shall enable the user to . . .	8	6	10	8	6	10	6	4	8				Need to build API and widget XYZ
1.1.2	The system shall enable the user to . . .	6	4	8	8	6	10	6	4	8				Need to edit data handling APIs
1.2.1	The system shall enable the user to . . .	4	3	5	1	1	1	5	4	7				Need to build presentation widget
1.2.2	The system shall enable the user to . . .	7	5	9	6	4	8	10	7	13				Need to integtrate XYZ COTS tool
1.2.3	The system shall enable the user to . . .	7	5	9	10	7	13	4	3	5				Need to implement XYZ interface
2.1.1	The system shall enable the user to . . .	2	1	3	4	3	5	1	1	1				Need to implement XYZ function in data layer
2.1.2	The system shall enable the user to . . .													Effort included in estimate above
2.1.3	The system shall enable the user to . . .	3	2	4	1	1	1	3	2	4				Need to build presentation widget
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]				
X.X.X	The system shall enable the user to . . .	2	1	3	7	5	9	9	6	12				Need to build API and widget XYZ

- Additional information captured in the room during the session, such as sketches or assumptions on a white board or smart board captures

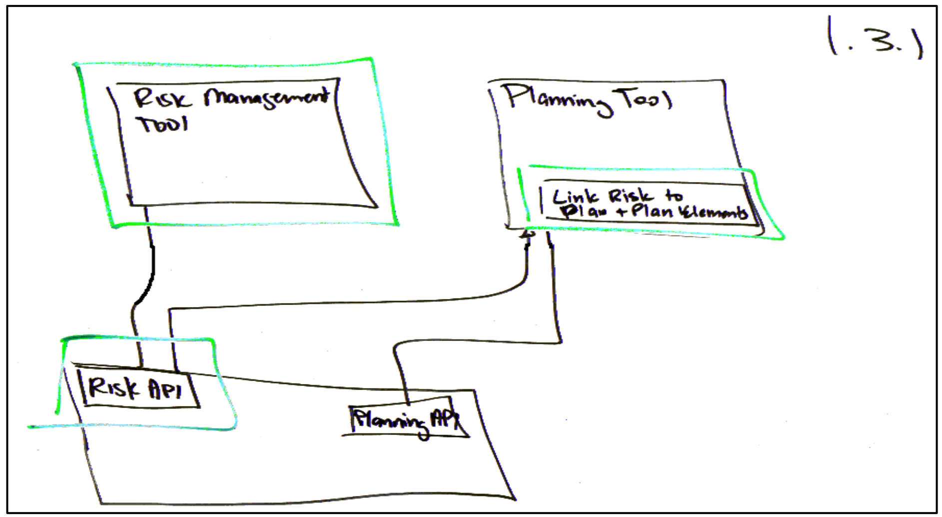


Table Of Contents

Step 4: Estimate Using Results

Other costs must be added to the scoring outputs to derive full software development cost estimate

Estimated based on scoring session outputs; provides foundation for the rest of the SW development estimate

Software Development Costs

Implementation (Coding) Costs

Scoring effort (Implement capabilities: Design -> Unit Test)

Non-capability development effort (Defects, HSI, etc.) → ~25-50% of scoring effort

Development Support (Non-Coding) Costs

PM, QA, CM, System-Level IAT, Documentation/CDRLs, etc. → ~80% of implementation costs

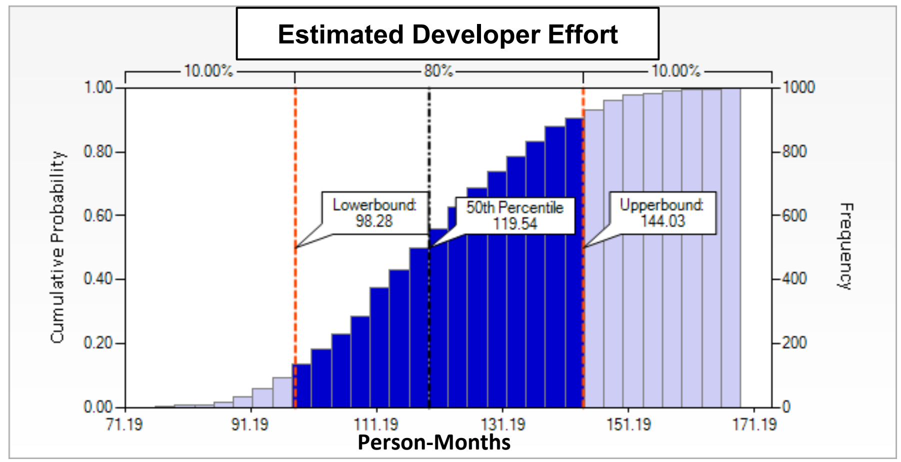
Can be estimated using a factor like this or using a LOE build-up

Step 4: Estimate Using Results

Ranges provided during scoring analysis provide solid inputs for robust uncertainty analysis

- Ranges provided by multiple scorers provide many possibilities for uncertainty bounds
  - Average of Expected
  - Min of Low, Max of High
  - Average of High as Expected
- Uncertainty can be applied at whatever level in requirements desired

		Round 1									Round 2		
		Scorer 1			Scorer 2			Scorer 3			1	2	3
Req #	Req. Description	Exp.	Low	High	Exp.	Low	High	Exp.	Low	High			
1.1.1	The system shall enable the user to . . .	8	6	10	8	6	10	6	4	8			
1.1.2	The system shall enable the user to . . .	6	4	8	8	6	10	6	4	8			
1.2.1	The system shall enable the user to . . .	4	3	5	1	1	1	5	4	7			
1.2.2	The system shall enable the user to . . .	7	5	9	6	4	8	10	7	13			
1.2.3	The system shall enable the user to . . .	7	5	9	10	7	13	4	3	5			
2.1.1	The system shall enable the user to . . .	2	1	3	4	3	5	1	1	1			
2.1.2	The system shall enable the user to . . .												
2.1.3	The system shall enable the user to . . .	3	2	4	1	1	1	3	2	4			
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]			
X.X.X	The system shall enable the user to . . .	2	1	3	7	5	9	9	6	12			



Uncertainty analysis is a pivotal step to bound cost and schedule estimates

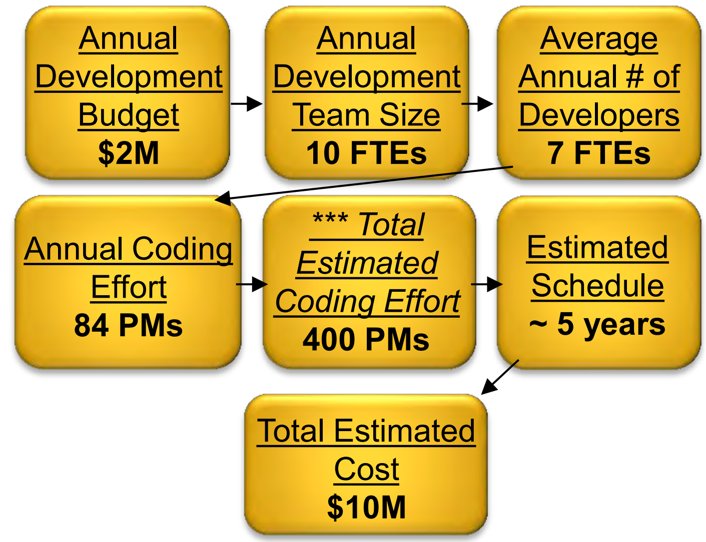
Uncertainty adjusted developer estimates can be used to build up total estimate using multiple methodologies

Step 4: Estimate Using Results

- Variable schedule/CAIV driven methodology based on annual budget constraints
  - Annual budget determines team size and number of developers
  - # of developers coupled with scoring session effort estimates determine total estimated schedule
  - Non-coding effort can be added along that schedule using factors or LOE
- Fixed schedule driven methodology based on schedule constraints
  - Targeted schedule determines how developer effort gets spread and required # of developers
  - Non-coding effort can be added using factors or LOE
  - Requires sanity check on required development team size for reasonableness

Examples

Variable Schedule / Fixed Annual Cost



Fixed Schedule / Variable Annual Cost

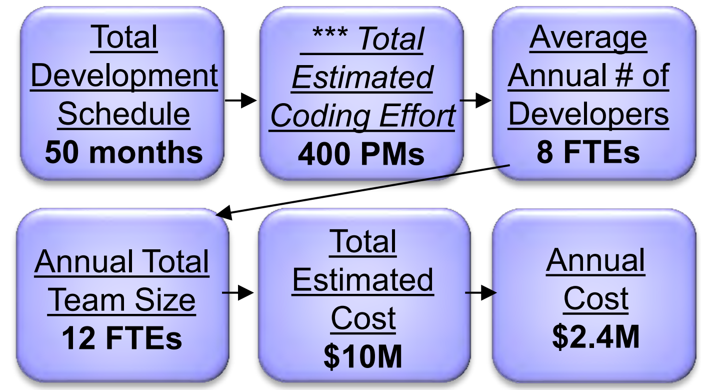
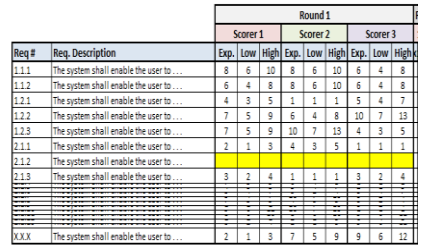


Table Of Contents

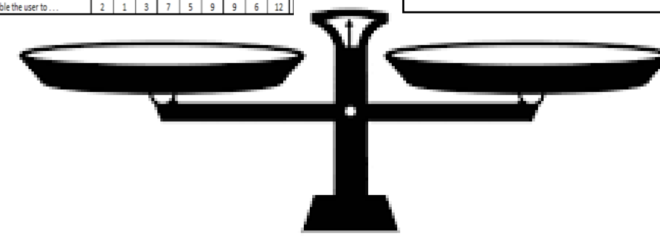
Utilizing this process to estimate size of a software project has various benefits and a few challenges

Benefits
More intuitive scope sizing methodology
Sizing explicitly relates complexity to effort
Systematic scoring process
Scoring ranges inform robust uncertainty
Allows for trade-off analysis at requirement level
Challenges
Subjective sizing inputs; limited analogous data
New type of cost model required
Sizable effort/coordination to run scoring session



		Round 1								
		Scorer 1			Scorer 2			Scorer 3		
Req #	Req. Description	Exp.	Low	High	Exp.	Low	High	Exp.	Low	High
1.1.1	The system shall enable the user to . . .	8	6	10	8	6	10	6	4	8
1.1.2	The system shall enable the user to . . .	6	4	8	8	6	10	6	4	8
1.2.1	The system shall enable the user to . . .	4	3	5	1	1	1	5	4	7
1.2.2	The system shall enable the user to . . .	7	5	9	6	4	8	10	7	13
1.2.3	The system shall enable the user to . . .	7	5	9	10	7	13	4	3	5
2.1.1	The system shall enable the user to . . .	2	1	3	4	3	5	1	1	1
2.1.2	The system shall enable the user to . . .									
2.1.3	The system shall enable the user to . . .	3	2	4	1	1	1	3	2	4
X.X.X	The system shall enable the user to . . .	2	1	3	7	5	9	9	6	12

$$PM = A \times Size^{E} \times \prod_{i=1}^{n} EM_i$$



For many projects, the benefits outweigh the challenges, and challenges can be mitigated

Table Of Contents

Collaborative requirements estimating is a viable new methodology for informing cost estimates for software projects

- While traditional methodologies are still viable for various types of software projects, this new methodology is viable for newer projects with limited analogs and new requirements
  - Scoring methodology is intuitive to software engineers and aligns with how software teams estimate, plan, and execute work
  - Disciplined scoring process attempts to add objectivity and documentation to subjective inputs
  - Scoring ranges enable detailed uncertainty analysis
  - Estimates at low requirement levels enable detailed scope trade-off analysis
  - Easy to explain to decision makers and diagnose estimating error

Questions?

For further information . . .

Blaze Smallwood
Associate

Booz | Allen | Hamilton

Booz Allen Hamilton Inc.
Office 309.359.3160
Mobile 619.850.6123
smallwood_blaze@bah.com

COST ASSESSMENT DATA ENTERPRISE

CADE Vision & Current Initiatives
Software and IT CAST

August 2016

Bess Dopkeen, PM CADE, OSD CAPE
bess.r.dopkeen.civ@mail.mil

Last Updated: August 12, 2016

COST ASSESSMENT DATA ENTERPRISE

CADE Objectives/KPPs

OSD CAPE

OBJECTIVES

Provide decision makers with relevant, high quality, timely and actionable analyses for better acquisition strategies and execution

- Move from reactive to proactive
- Insight equates to trust and facilitates faster and more knowledgeable decision making
- Facilitate telling the program's "story", holistic analysis

Improve Analyst Productivity (at all levels: OSD, Services, PMOs)

- Increase output per unit time, without degrading confidence in results
- Provide near real-time access to data, more data, and less burden on the analyst to retrieve and process
- Reduce time for analyst to climb the program familiarization learning curve

Comprehensiveness

- Having all DoD's relevant data at analysts' fingertips for comprehensive assessments, regardless of analysis type

Community Knowledge Sharing

- Gain insight from previous and fellow analysts and data stakeholders

KPPs

Quality and Transparency of Source Data

- Where it comes from, what we know about it – consistency
- Enterprise data stewardship – Enterprise agreement and accountability for what data means and how it's used
- Reporting Compliance Improvement

Properly Secured

COST ASSESSMENT DATA ENTERPRISE

Comprehensiveness and Our Vision

Comprehensive Data Availability:

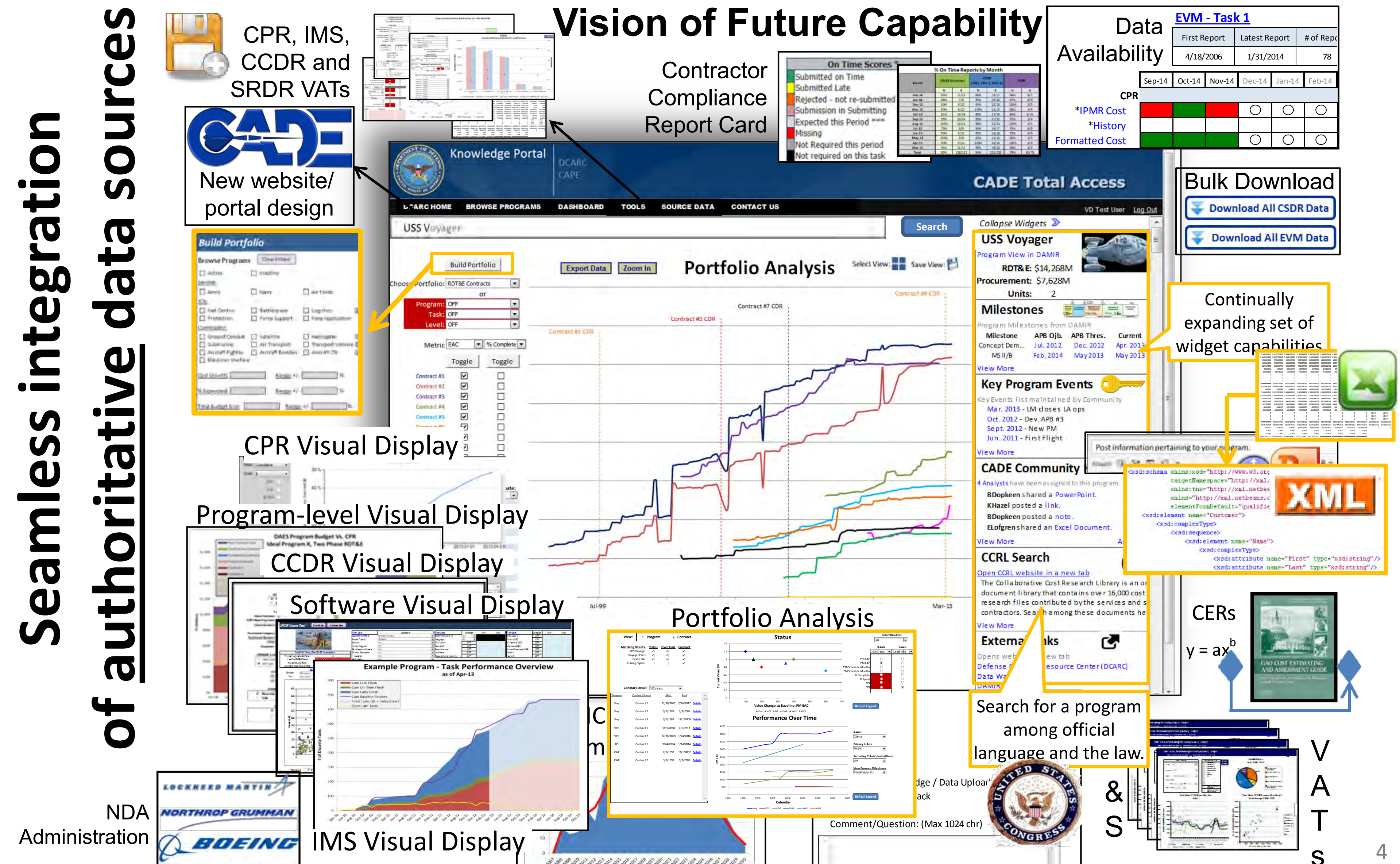
All information at the analyst's fingertips – centralized virtual library with everything in it

- **Cost Data (CCDRs/1921s)**: Contains all an analyst needs to build an estimate
  - **FlexFiles**: New generation of cost data collection
- **Cost Analysis Requirements Description (CARD) / Technical Data ("1921-T")**: Programmatic and technical descriptions analysts need to build estimates
- **Software Resource Data Report (SRDR):** Software effort, size, and schedule estimating approaches including analogy, parametric and commercial models
- **Institutional Knowledge:** What analysts need to know about the data
- **Policy Improvements**
- **Community Support**

Our Vision for CADE:

Cost analysts will have all of this data and institutional knowledge at their fingertips. It will be the exception – not the rule – to go back to industry to do our estimates

CADE Vision of the Future: Total Analyst Access



COST ASSESSMENT DATA ENTERPRISE

CADE Coalition: The Cost Community, AT&L & Industry

Cost

FlexFile: Daron Fullwood, CAPE
CSDR/EVM Co-Plan, WBS Alignment: John McGregor, AT&L PARCA/EVM
1921-3: Mike Biver and Carol Moore, CAPE
Sustainment: Tom Henry, CAPE; Lisa Mably, AFCAA

Technical

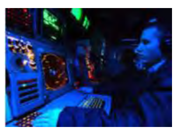
SRDR: Ranae Woods, AFCAA
CARD: Curt Khol, CAPE
Tech Data WG: Greg Hogan, AFCAA
MAIS WG: Richard Mabe, AFCAA

Commodity Study Joint Effort

Aircraft, UAV

Missiles

Radar, C2 Center, C4I



ICBM

O&S

Ships

Space

WTV

MAIS

Office Collaboration

Air Force
AFCAA CEM joint effort on CADE, commodity leads, Contracts Databases, SMC early FlexFile prototypes

Navy
FlexFile, JCARD (NAVAIR), Ships WG, CCRL, CER Handbook

USMC
USMC BOM/CER Effort

Army
JIAT, ACDB/WTV prototype, WTV CIPT, TACOM, Historical Data Migration

MDA
MDA-DCARC alignment, CCRG

AT&L
EVM-CR, CSDR/EVM Co-Plans, DAVE (DAMIR, AIR, Kaleidoscope) DDR&E/SE tech data; Big Data initiative, LM&R CARD input, DCMA, DPAP, DAU

Industry
CSDR Focus Group, Joint Training, NDIA, **FlexFile Pilot Leads:** LMCO, Boeing, NGC, BAE, GDLS, HII, Ball Aerospace **CIPTs:** Aviation, JSCC, O&S, Software and IT, WTV

Service Cost Agency Leads

DASA-CE
David Henningsen
Sean Vessey

NCCA
Duncan Thomas
John Fitch

AFCAA
Ranae Woods
Greg Hogan


FlexFiles
The Future of Cost Analysis

COST ASSESSMENT DATA ENTERPRISE

FlexFiles: Objectives

A Win-Win Government and Industry Partnership

Today's Shortcomings

CCDR Data

- Time consuming to industry
- No details below the CCDR functional labor categories
- Allocations are not transparent
- Limited Data sampling over time
- Allows for human error

Ad-Hoc Data Calls

- Time consuming to industry
- Requires burdensome site visits
- Limited Access to Data
- Allows for human error

The FlexFile Solution

Increase Efficiency:

- Collect data according to the contractor's management structure
- Removal of legacy 1921 forms
- Reduce ad hoc/supplemental government data collection efforts
- Much easier and less time consuming for Industry – allows them to reduce back end support
- Automation: data flows directly from contractor systems into ours

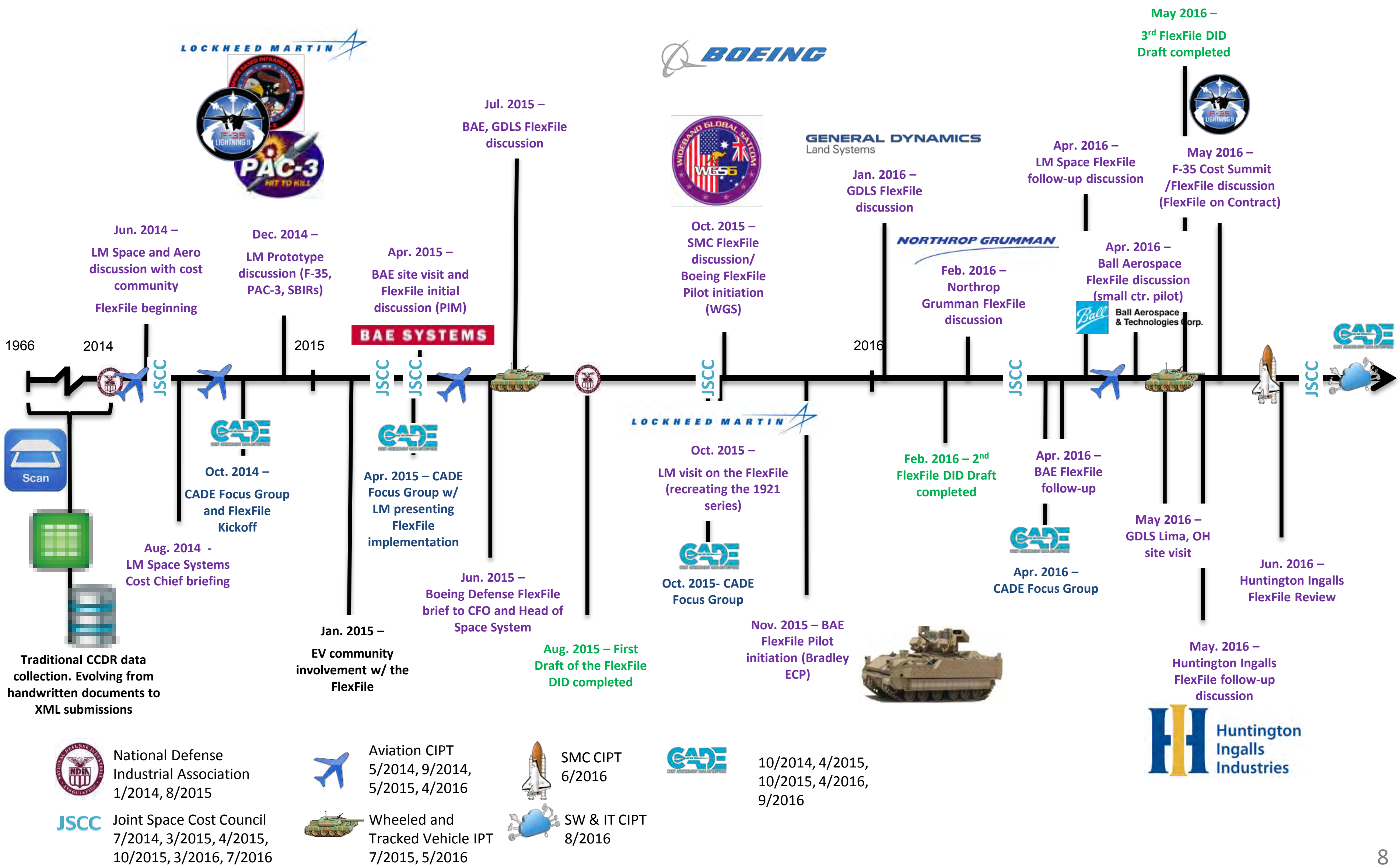
Improving Data Quality:

- Eliminate Human Error/Subjectivity
- Collect raw data, and use technology to eliminate arbitrary allocations and errors
- Consistent application of Mil-STD-881C to both EV and CSDR data – data Alignment
- Review and mapping pre-contract award

Ensure Completeness:

- Provides much more insight and analysis flexibility
- Annual submissions
- Receive data over time
- Include cost and supporting technical data

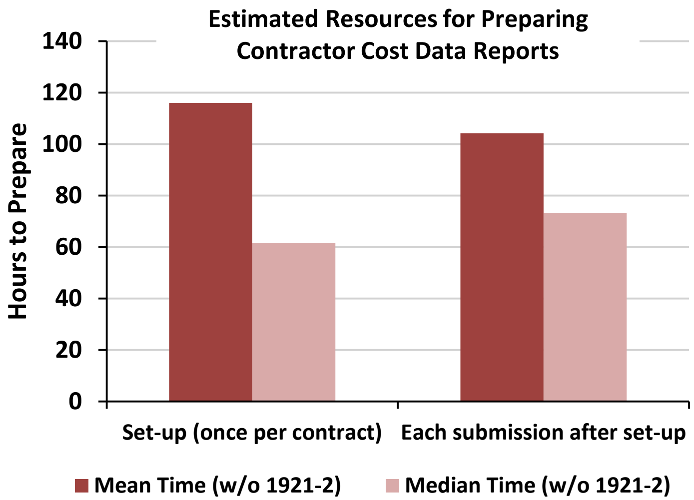
History of the FlexFile Effort



More Data, Less Time

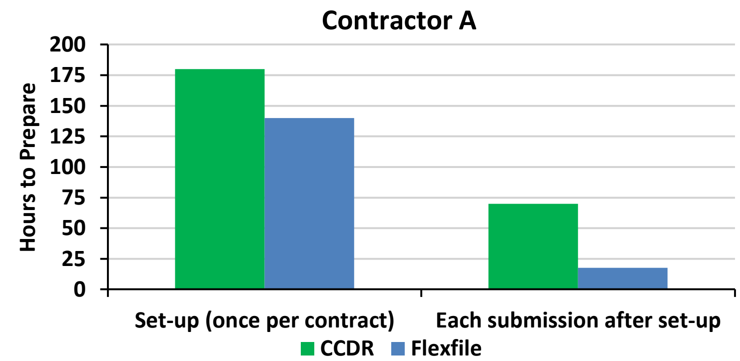
Today's Burdensome CCDR Process

1921, 1921-1 formatted CCDRs today require an average of *533 hours* per contract, assuming set-up and 4 reports

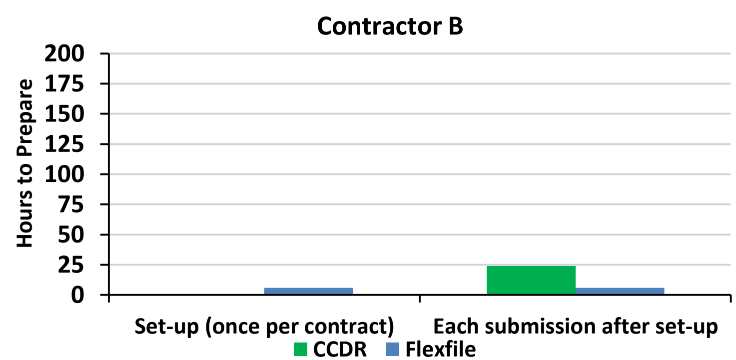


This is how we have collected contractor cost data for nearly 50 years

Efficiencies Realized from the FlexFile



- Complex program requiring post-extraction allocations
- FlexFiles covers 1921 & 1921-1
- Provided enough detail to replace 1921-2



- Did not require post-extraction allocations
- Provided manufacturing floor hours report
- Had automated CCDR reporting with scripts

COST ASSESSMENT DATA ENTERPRISE

FlexFile Draft DID (May/August 2016)

OSD CAPE

A *WBS Dictionary & Remarks*

WBS Index
Definitions by WBS
Cost Content
Work Content
Supplier & GFE elements
Contractor Remarks, Comments by WBS Element
Direct-Reporting Subs

B *Metadata*

Program Name
Contract #
Approved Co-Plan #
Contractor Name, Location, POC
As of Date
Submission Event Name
Phase
Report Type

C *Contractor Cost Data Report*

Unallocated Actual Costs & Hours
WBS, Control Account, Work Package
Data by Month
Rec-Qty. Vs. Rec-Time vs. Nonrecur.
CLIN & Lot
Functional Rate (Gov & Internal)

D *Contractor Cost Data Field Dictionary*

Contractor Internal Accounting Data Field Descriptions

E *Allocation Methodology*

Contractor's Distribution of Unallocated Actual Costs
Unit/Lot Level Allocation

As required by Co-Plan

F *Estimates at Completion*

Estimates at Complete (EAC) by WBS Element (as required by Co-Plan)

G *Supplemental Information*

MRP Floor Hours Report

CADE COST ASSESSMENT DATA ENTERPRISE

FlexFile Critical Path

CRAWL

June 2016 – December 2016

- -Tailored FlexFile DID
- XLS FlexFile submission
- Recreate 1921, -1; Consistent w/-2

Immediate change that can easily be executed

WALK

January 2017 – June 2017

- -Approved FlexFile DID
- Define FlexFile XML schema

Near-term change that can be executed once processes are streamlined

RUN

July 2017 - Future

- XML FlexFile submission
- 1921-T submission

Long-term change executed by a mature organization

COST ASSESSMENT DATA ENTERPRISE

Strategic Data Planning

March 18, 2016: CAPE & PARCA signed CSDR/EVM Co-Plan Pilot Memo

OFFICE OF THE SECRETARY OF DEFENSE
1800 DEFENSE PENTAGON
WASHINGTON, D.C. 20301-1800

COST ASSESSMENT AND
PROGRAM EVALUATION

MAR 18 2016

MEMORANDUM FOR: SEE DISTRIBUTION

REFERENCE: Contractor Cost and Software Data Reporting (CSDR) and Earned Value Management (EVM) Co-Plan

As required by Tables 7 and 8 of DoD Instruction 5000.02, "Operation of the Defense Acquisition System," CSDR and EVM reports must be submitted for a variety of contracts issued by the Department. In order to increase efficiency, improve data quality, and foster efficient and consistent cost reporting, the Office of the Secretary of Defense, Cost Assessment and Program Evaluation (CAPE), and Office of the Secretar
Technology and Logistics (AT&L) Performance Assessmen
(PARCA) organizations are proposing a CSDR/EVM Co-Pl
reporting. All contracts requiring CSDR and EVM reportin
CSDR/EVM Co-Plan (using Directives Division (DD) Form
planning process.

Richard P. Burke
Deputy Director
Cost Assessment

Gary Bliss
Director
Performance Assessments and
Root Cause Analyses

Standard Co-Planning Benefits

STANDARD CO-PLAN

One Voice, One Direction

PM, CAPE, PARCA & Services

- Upfront planning pre-RFP release
- Standardization by commodity area and phase
- Reduction in contractor administrative burden
- Integrated Post Award Meeting
- Aligned with SRDR, Tech data & IPMR goals
- Less allocations caused by CCDR and CPR misalignment
- Support needs of both Cost & EV communities

Work Products:

Standard CSDR Plans for each commodity area

- Rooted in analytical requirements
- Based on assessment of contemporary plans

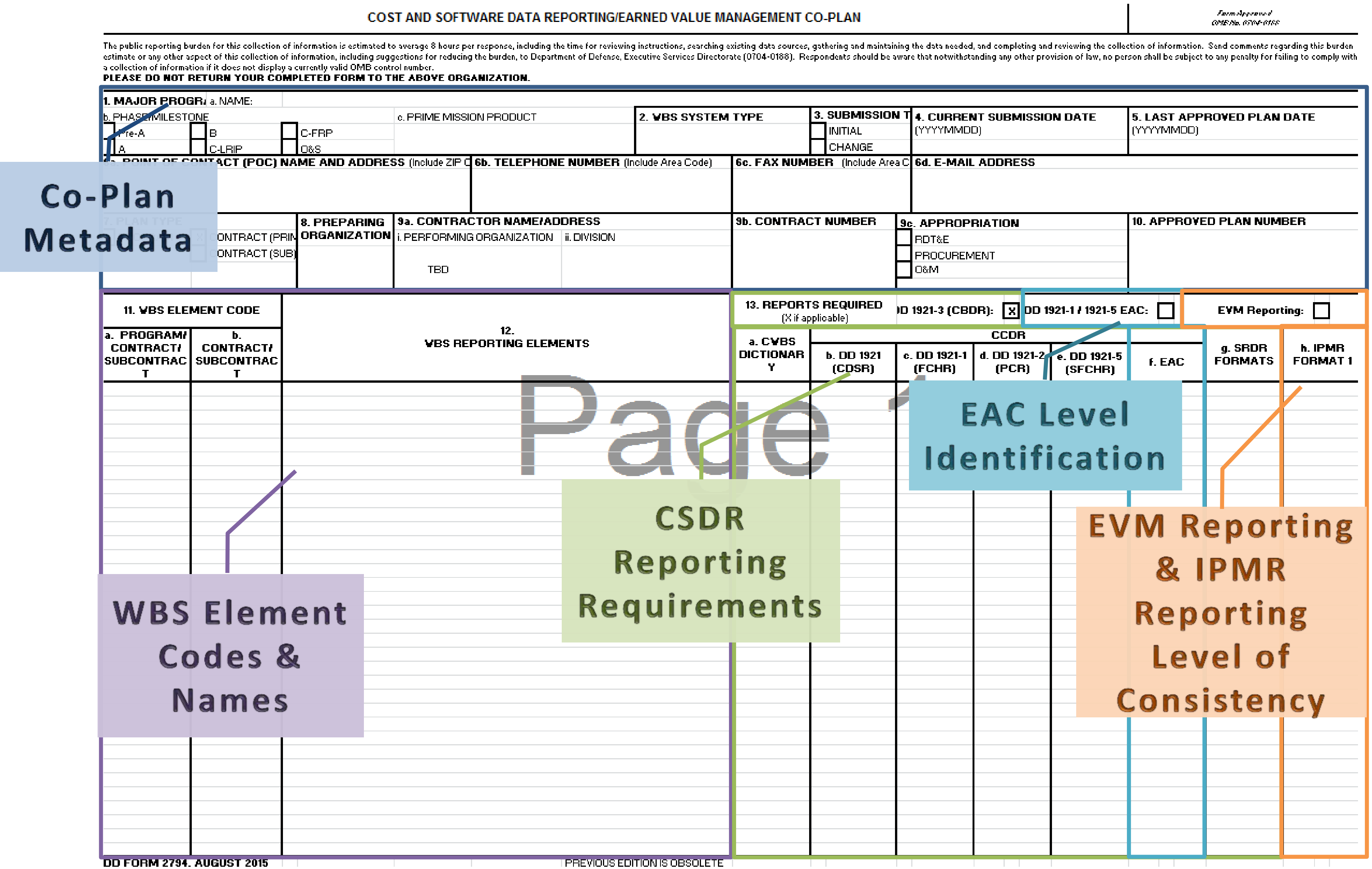
Extensions to 881C appendices

- Added detailed child elements to give granularity into high-interest topics
- Removed extraneous elements; clarified content or improved their location

Implemented as starting point for DCARC and PARCA CSDR/EVM Co-Planning

- Better communication of expectations to Industry
- Serve as a starting point for a new program CWIPT's initial plan
- Serve as a reporting ideal for on-going program's tailoring of new contract plans

CSDR/EVM Co-Plan Format



COST AND SOFTWARE DATA REPORTING/EARNED VALUE MANAGEMENT CO-PLAN

Form Approved OMB No. 0704-0188

The public reporting burden for this collection of information is estimated to average 8 hours per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Executive Services Directorate (0704-0188). Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

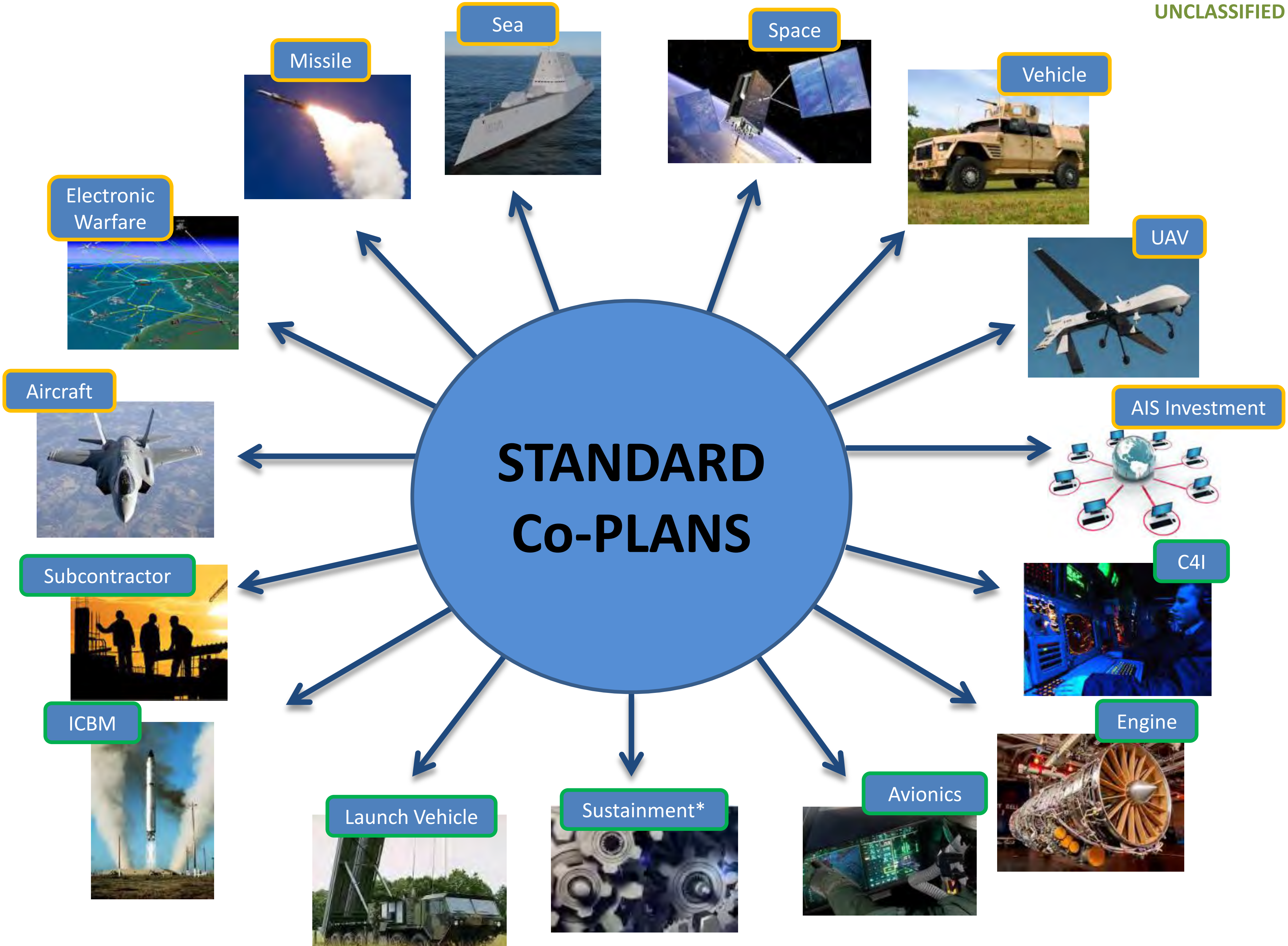
PLEASE DO NOT RETURN YOUR COMPLETED FORM TO THE ABOVE ORGANIZATION.

1. MAJOR PROGR/ a. NAME:
b. PHASE/MILESTONE: Pre-A, B, C-FRP, A, C-LRIP, O&S
c. PRIME MISSION PRODUCT
2. WBS SYSTEM TYPE
3. SUBMISSION T: INITIAL, CHANGE
4. CURRENT SUBMISSION DATE (YYYYMMDD)
5. LAST APPROVED PLAN DATE (YYYYMMDD)
POINT OF CONTACT (POC) NAME AND ADDRESS (Include ZIP C
6b. TELEPHONE NUMBER (Include Area Code)
6c. FAX NUMBER (Include Area C
6d. E-MAIL ADDRESS
7. PLAN TYPE: ONTRACT (PRIM, ONTRACT (SUB)
8. PREPARING ORGANIZATION
9a. CONTRACTOR NAME/ADDRESS: i. PERFORMING ORGANIZATION, ii. DIVISION, TBD
9b. CONTRACT NUMBER
9c. APPROPRIATION: RDT&E, PROCUREMENT, O&M
10. APPROVED PLAN NUMBER

11. WBS ELEMENT CODE: a. PROGRAM/CONTRACT/SUBCONTRACT	b. CONTRACT/SUBCONTRACT	12. WBS REPORTING ELEMENTS	13. REPORTS REQUIRED (X if applicable): a. CWBS DICTIONARY	CCDR: b. DD 1921 (CDSR)	c. DD 1921-1 (FCHR)	d. DD 1921-2 (PCR)	e. DD 1921-5 (SFCHR)	f. EAC	g. SRDR FORMATS	h. IPMR FORMAT 1

DD 1921-3 (CBDR): [X] DD 1921-1 / 1921-5 EAC: [ ] EVM Reporting: [ ]

DD FORM 2794, AUGUST 2015 PREVIOUS EDITION IS OBSOLETE



***Includes an AIS Sustainment WBS**

CADE
COST ASSESSMENT DATA ENTERPRISE

Technical Data

COST ASSESSMENT DATA ENTERPRISE

Tech Data Streamlining

Lessening Burden and Improving Data

OSD CAPE

Current State:

- Current data collection methods are ad-hoc, inefficient and scattered across the department
- Result: We re-construct technical analogies for nearly every estimate

Commodity Unique Templates

- Estimating Requirements
- Comprehensive data source

Program Template

Acquisition Template

PMP Hardware Template

Nonhardware Template

Cost Template

Other Templates
(OGC, Software, O&S, Roles, Configuration...)

Metadata

MIL-STD 881C WBS (Oct '11)

Air Force
Army
Navy

CARD

- Interim Policy (Jun '15)
- Updated Format
- Annual Delivery
- Historical Record

Tech Data Report **1921-T**

- SWaP parameters
- Complement to cost and software
- Completes the CARD

Other Data

- Collaborate with Acquisition and Technical Communities; SE, L&MR and ARA

CADE

- Authoritative data source
- One stop shop for analysis
- Visual Analytics

FY15 | FY16 | FY17

Interim CARD
Draft 1921-T

Final CARD
Final 1921-T

Train and Transition

CADE
COST ASSESSMENT DATA ENTERPRISE

CADE Path Forward

CADE Planned Accomplishments – Next 12 months

Policy:

- CARD Guidance Policy Memo
- SRDR DID Policy Memo
- Draft Tech Data DID
- Co-Plan Pilots
- FlexFiles – FlexFile Prototype Instruction and Prototypes (Next Draft DID Update)

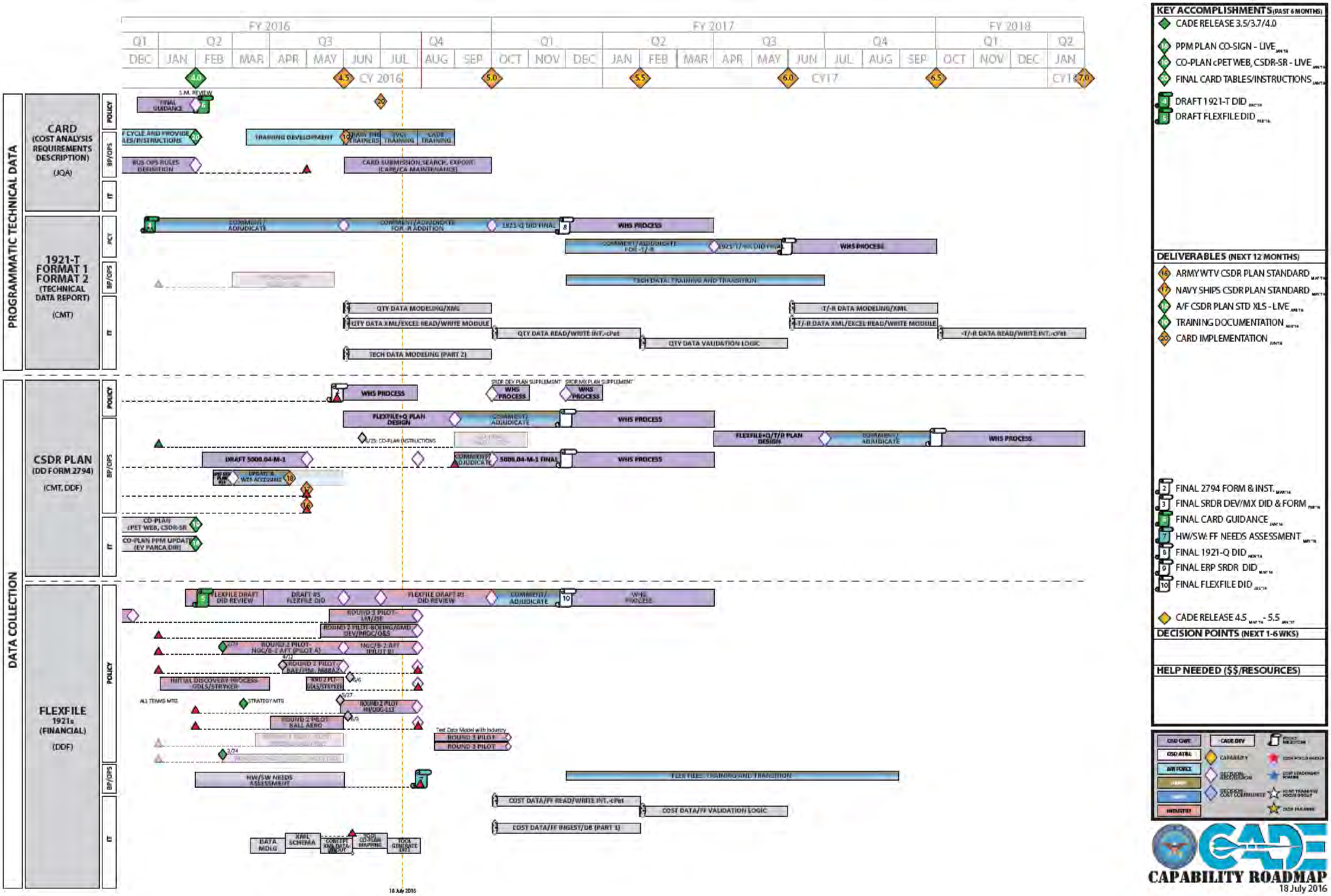
Business Process:

- Co-Plan (DD Form 2794)
- CSDR Plan Standards
- CARD Submission
- ICE Submission

IT Development 5.0 to 5.5:

- Cross Program Query Follow-on
- 1921-3 Data Availability
- MARs Data Availability
- FlexFile IT Development
- 1921-Q IT Development

CADE Initiatives



COST ASSESSMENT DATA ENTERPRISE

CADE Closing: Why It's Important

Improved Acquisition Outcomes:

- **Authoritative Quality Data**: Cooperative planning and compliance lead to better data and improved program management
- **Cost Realism**: Provide real-time cost data for analysis and facilitating quicker contract negotiations
- **Full view of Weapons Systems Program Performance**: Visual analytics, trend analysis and technical data to improve cost realism and make informed decisions

Efficient and Effective Analysis (at all levels: OSD, Services, PMOs):

- Improved Analytical Rigor and Productivity
- More time for analysis and execution; Less time collecting and feeding data
- More comprehensive assessments and reduced burden on industry

Cost Community Coordination:

- Revolutionizing cost data collection
- Cost community ownership of leadership, training and estimating responsibility
- Improving terminology and practices across Departments

Let's continue to become more efficient together

Cost Assessment Data Enterprise

Software Resources Data Reporting (SRDR)

Updates and Revisions:
Development SRDR
Maintenance SRDR
ERP Development SRDR

SW and IT CAST

Presented By:
Mr. Richard Mabe, AFCAA

23 August 2016

Purpose

- Provide an overview of the new SRDR DID (DI-MGMT-8035) and reporting formats:
  - Format 1- Development (DD Form 3026-1)
  - Format 2- Maintenance (DD Form 3026-2)
- Now available on the DCARC public website http://dcarc.cape.osd.mil/CSDR/FormsReporting.aspx
- Example CDRLs also available http://dcarc.cape.osd.mil/CSDR/Planning.aspx
- Introduce an ERP Development SRDR
  - Addendum to DI-MGMT-8035
  - Format 3- DD Form 3026-3

Cost Assessment Data Enterprise

History

- Software (SW) development/support cost is significant
- Quality data underpins quality SW cost estimate
- Data collection via SRDRs began in 2004
  - Size and Effort focus, but collect over 170 data fields
  - Inconsistent/non-standard data and formats
  - SRDRs available from DCARC but manually input in various "databases" (e.g. NAVAIR Excel spreadsheet)
- Data widely used by cost community but in need of more standardization and quality improvement
  - ~ 40% of data to-date is "good" for primary cost analysis use
  - ~ 20% of data is good for growth analysis (i.e. initial & final)

SRDRs – A success story ready for its next chapter…

Cost Assessment Data Enterprise

SRDR Working Group – Vision and Recommendations

One OSD-hosted, central, user-friendly, authoritative, real-time software cost, technical, programmatic database and tool

Recommendation	Benefit
1. Revised SRDR Data Item Description (DID) for Development	1. Reduces inconsistency, lack of visibility, complexity, and subjectivity in reporting
2. New SRDR Maintenance Format included in DID	2. Aligned w/ dev. but w/ unique data/metrics for maintenance phase
3. Joint Validation & Verification (V&V) Guide, Team, and Process	3. Higher quality, less duplication; joint team & guide gives early, consistent feedback to contractors
4. CADE Software Database Design and Implementation	4. Avoids duplication, variations - ONE central vs many distributed; Based on surveyed best practices and user expectations

Formats

SRDR Initial & Final (Old)

- Software size, effort, and schedule
- Data dictionary (Descriptions)
- Initial Developer Report- Due at beginning of project increment (estimates)
- Final Developer Report- Due at completion of project increment (actuals)

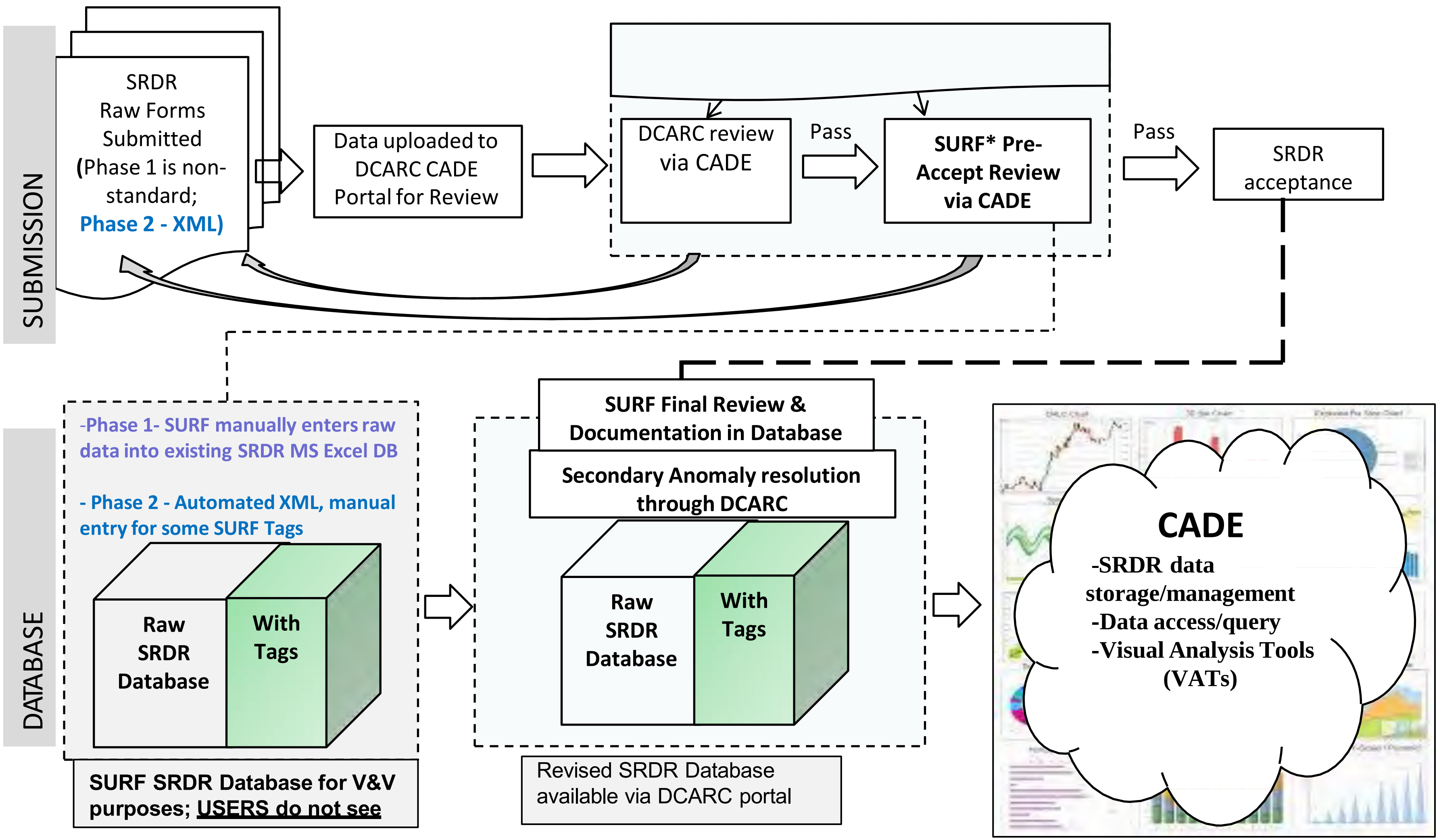
SRDR Development (New)

- Part 1: Software size, technical parameters, descriptions, and schedule
- Part 2: Effort -actuals, EACs, and time phased (tie to 1921-1)
- Specific Release Level and CSCI Level Reporting
- Initial (estimates), Interim (actuals to-date/estimates), and Final (actuals, monthly) Reporting

SRDR Maintenance (New)

- Part 1: Software size, technical parameters, descriptions, and schedule
- Part 2: Effort (actuals) (tie to 1921-5)
- Release Level Reporting
- Annual (actuals) Reporting

Process



*SRDR Unified Review Function

V&V Process

- **SURF Team:**

DCARC Analyst: SRDR Submission received from DCARC

	CAPE	Navy	Marine Corps	Air Force	Army	SPAWAR	MDA
SURF Primary:	William Raines	Corrinne Wallshein	Noel Bishop	Ethan Henry Ron Cipressi	Jim Judy Jenna Meyers James Doswell	Jeremiah Hayden	Dan Strickland
SURF Secondary:	Various	Scott Washel Dane Cooper Stephen Palmer Philip Draheim	John Bryant	Janet Wentworth Chinson Yew Eric Sommer	Michael Smith Michael Duarte	Min-Jung Gantt	Various

- **Performs pre and post SRDR acceptance V&V w/ DCARC**
- **Uses a detailed first-ever published *joint* V&V guide**
  - Training guide and used to determine SRDR quality tags for database
- **Submits distributed amongst SURF members to balance workload**

Cost Assessment Data Enterprise

Software Development Report

SRDR Implementation Process

CWIPT Plan Development

- The Program Office (PO) will use cPet to insert the commodity-based standard WBS or O&S CRS and tailor accordingly
- Elements with software data will be identified
- CDRL for SRDR will reference approved Plan (CSDR or Flexfiles)

SRDR Placed on Contract

- Post Award Contract Meeting between contractor and CWIPT will further define the software data requirements
- Plan revision possible with CSDR Supplement

CSDR Plan Supplement Revisions

- Revise Page 3 of the CSDR Plan as Release dates, Release Names, and CSCIs are defined

Software Data Submitted to CSDR Submit-Review

- Contractor will submit 3026-1 and/or 3026-2 according to the requirements in the approved Plan
- Part 1 will be submitted in XML to the Submit-Review
- Part 2 (if required) will be submitted in MS Excel to the Submit-Review
- Validated according to requirements within approved Plan

Approved Data Available in CADE

- Cost and software data parsed and viewed together within CADE for analysis
- Requires that software data submissions align with cost submissions

Cost Assessment Data Enterprise

Human Readable Format*

SECTION 3.1.3 UNCLASSIFIED SECURITY CLASSIFICATION	OMB Control Number 0704-0188 Expiration Date: 8/31/2016

The public reporting burden for this collection of information is estimated to average 16 hours per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Executive Services Directorate, Directives Division, 4800 Mark Center Drive, East Tower, Suite 02G09, Alexandria, VA 22350-3100 (0704-0188). Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. PLEASE DO NOT RETURN YOUR COMPLETED FORM TO THE ABOVE ORGANIZATION.

SOFTWARE RESOURCES DATA REPORTING: Metadata SECTION 3.2.1

MAJOR PROGRAM NAME: SECTION 3.2.1.1	PRIME MISSION PRODUCT SECTION 3.2.1.3
PHASE/MILESTONE: SECTION 3.2.1.2 ☐ Pre-A ☐ B ☐ C - FRP ☐ A ☐ C - LRIP ☐ O&S	REPORTING ORGANIZATION TYPE SECTION 3.2.1.4 ☐ PRIME/ASSOCIATE CONTRACTOR ☐ DIRECT-REPORTING SUBCONTRACTOR ☐ GOVERNMENT
PERFORMING ORGANIZATION SECTION 3.2.1.5 a. NAME: SECTION 3.2.1.5.1 b. ADDRESS:	DIVISION SECTION 3.2.1.5.2 a. NAME: b. ADDRESS:
APPROVED PLAN NUMBER SECTION 3.2.1.6	CUSTOMER SECTION 3.2.1.7
TYPE ACTION a. CONTRACT NO: SECTION 3.2.1.8.1 d. NAME: SECTION 3.2.1.8.4	b. MODIFICATION NO.: SECTION 3.2.1.8.2 c. SOLICITATION NO.: SECTION 3.2.1.8.3 e. TASK ORDER/DELIVERY ORDER/LOT NO.: SECTION 3.2.1.8.5
REPORT TYPE SECTION 3.2.1.10 INITIAL ☐	INTERIM ☐ FINAL ☐
PERIOD OF PERFORMANCE SECTION 3.2.1.9 a. START DATE *(YYYYMMDD)*: b. END DATE *(YYYYMMDD)*:	APPROPRIATION SECTION 3.2.1.16 ☐ RDT&E ☐ PROCUREMENT ☐ O&M

SUBMISSION NUMBER	SECTION 3.2.1.11
RESUBMISSION NUMBER	SECTION 3.2.1.12
REPORT AS OF *(YYYYMMDD)*	SECTION 3.2.1.13
DATE PREPARED *(YYYYMMDD)*	SECTION 3.2.1.15

POC NAME (Last, First, Middle Initial)	DEPARTMENT	TELEPHONE (Include Area Code)	EMAIL ADDRESS
SECTION 3.2.1.14			

REMARKS SECTION 3.2.1.17

SECTION 3.1.3 UNCLASSIFIED
SECURITY CLASSIFICATION

DD Form 3026-1, MAY 2016

*Double Click Object to see Complete Worksheet

Cost Assessment Data Enterprise

Software Maintenance Report

Cost Assessment Data Enterprise

Human Readable Format*

SECTION 3.1.3 UNCLASSIFIED

SECURITY CLASSIFICATION

OMB Control Number 0704-0188
Expiration Date: 8/31/2016

SOFTWARE RESOURCES DATA REPORTING: Metadata SECTION 3.2.1

MAJOR PROGRAM NAME: SECTION 3.2.1.1	PRIME MISSION PRODUCT SECTION 3.2.1.3
PHASE/MILESTONE: SECTION 3.2.1.2 ☐ Pre-A ☐ B ☐ C - FRP ☐ A ☐ C - LRIP ☐ O&S	REPORTING ORGANIZATION TYPE SECTION 3.2.1.4 ☐ PRIME/ASSOCIATE CONTRACTOR ☐ DIRECT-REPORTING SUBCONTRACTOR ☐ GOVERNMENT
PERFORMING ORGANIZATION SECTION 3.2.1.5 a. NAME: SECTION 3.2.1.5.1 b. ADDRESS:	DIVISION SECTION 3.2.1.5.2 a. NAME: b. ADDRESS:
APPROVED PLAN NUMBER SECTION 3.2.1.6	CUSTOMER SECTION 3.2.1.7

TYPE ACTION
- a. CONTRACT NO: SECTION 3.2.1.8.1
- b. MODIFICATION NO.: SECTION 3.2.1.8.2
- c. SOLICITATION NO.: SECTION 3.2.1.8.3
- d. NAME: SECTION 3.2.1.8.4
- e. TASK ORDER/DELIVERY ORDER/LOT NO.: SECTION 3.2.1.8.5

REPORT TYPE SECTION 3.2.1.10 — INITIAL ☐ INTERIM ☐ FINAL ☐

PERIOD OF PERFORMANCE SECTION 3.2.1.9 a. START DATE *(YYYYMMDD)*: b. END DATE *(YYYYMMDD)*:	APPROPRIATION SECTION 3.2.1.16 ☐ RDT&E ☐ PROCUREMENT ☐ O&M	SUBMISSION NUMBER SECTION 3.2.1.11 RESUBMISSION NUMBER SECTION 3.2.1.12 REPORT AS OF *(YYYYMMDD)* SECTION 3.2.1.13 DATE PREPARED *(YYYYMMDD)* SECTION 3.2.1.15

POC NAME (Last, First, Middle Initial) SECTION 3.2.1.14	DEPARTMENT	TELEPHONE (Include Area Code)	EMAIL ADDRESS

REMARKS SECTION 3.2.1.17

SECTION 3.1.3 UNCLASSIFIED

SECURITY CLASSIFICATION

DD Form 3026-2, MAY 2016

*Double Click Object to see Complete Worksheet

Cost Assessment Data Enterprise

ERP Software Development Report

CAPE
COST ASSESSMENT & PROGRAM EVALUATION

Cost Assessment Data Enterprise

History

- **July 2015: compared the draft Development and Maintenance SRDR forms and revised DID to a proposed ERP version developed in 2012 by Dr Wilson Rosa**
  - Determined as a WG that the edits to add ERP specific data fields to the new SRDR formats were significant enough to warrant a separate form for ERP Development
  - But: the format and content for the Maintenance SRDR were adequate for ERP programs
- **Initial draft form and DID for ERP presented on 2 November to the WG**
- **Multiple iterations followed through March 2016**
- **Current version is a Final WG version (as of 1 May 2016)**:
  - Reviewed by service cost agencies, ERP integrators, service ERP program cost chiefs, others in DOD
  - Form in Excel
  - DID in Word

Cost Assessment Data Enterprise

Way Ahead

- **Review and process through OSD and Services**
  - To be reviewed by CADE team to ensure can be processed same as SRDR
  - Review and approve at Service Cost Chief level (Deputy Assistant Secretaries for Cost and Economic Analysis)
- **Submit as addendum to SRDR documents**
  - Same format as SW Development SRDR Form and DID
  - Cut and paste identical data fields from the SRDR
  - Share DID paragraphs 1.0 through 3.2
  - ERP SRDR unique instructions in DID Format 3 (paragraph 3.3 and 3.4)
- **Beginning work on V&V approach**
- **Will test with real-world program**

Software Resource Data Reports (SRDRs)

ERP SRDR
Development Only
(In Work)

- Part 1: Software size, technical parameters, implementation
- Part 2: Effort -actuals, EACs, and time phased (tie to 1921-1)
- Specific Release Level Reporting
- Initial (estimates), Interim (actuals to-date/estimates), and Final (actuals, monthly) Reporting

Cost Assessment Data Enterprise

Human Readable Format*

Enterprise Resource Planning (ERP) Software Development Report

SECTION 3.1.3 UNCLASSIFIED SECURITY CLASSIFICATION	OMB Control Number XXXX-XXXX Expiration Date: MM/DD/YYYY

The public reporting burden for this collection of information is estimated to average 16 hours per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Executive Services Directorate, Directives Division, 4800 Mark Center Drive, East Tower, Suite 02G09, Alexandria, VA 22350-3100 (0704-0188). Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. PLEASE DO NOT RETURN YOUR COMPLETED FORM TO THE ABOVE ORGANIZATION.

ENTERPRISE RESOURCE PLANNING (ERP) SOFTWARE RESOURCES DATA REPORTING: METADATA Section 3.2

MAJOR PROGRAM NAME: SECTION 3.2.1.1	PRIME MISSION PRODUCT SECTION 3.2.1.3
PHASE/MILESTONE: SECTION 3.2.1.2 ☐ Pre-A ☐ B ☐ C - FD ☐ A ☐ C - FDD ☐ O&S	REPORTING ORGANIZATION TYPE SECTION 3.2.1.4 ☐ PRIME/ASSOCIATE CONTRACTOR ☐ DIRECT-REPORTING SUBCONTRACTOR ☐ GOVERNMENT
PERFORMING ORGANIZATION SECTION 3.2.1.5 a. NAME: SECTION 3.2.1.5.1 b. ADDRESS:	DIVISION SECTION 3.2.1.5.2 a. NAME: b. ADDRESS:
APPROVED PLAN NUMBER SECTION 3.2.1.6	CUSTOMER SECTION 3.2.1.7

TYPE ACTION SECTION 3.2.1.8	a. CONTRACT NO: SECTION 3.2.1.8.1	b. MODIFICATION NO.: SECTION 3.2.1.8.2	c. SOLICITATION NO.: SECTION 3.2.1.8.3
	d. NAME: SECTION 3.2.1.8.4	e. TASK ORDER/DELIVERY ORDER/LOT NO.: SECTION 3.2.1.8.5	

REPORT TYPE SECTION 3.2.1.10	INITIAL ☐	INTERIM ☐	FINAL ☐

PERIOD OF PERFORMANCE SECTION 3.2.1.9 a. START DATE *(YYYYMMDD)*: b. END DATE *(YYYYMMDD)*:	APPROPRIATION SECTION 3.2.1.16 ☐ RDT&E ☐ PROCUREMENT ☐ O&M	SUBMISSION NUMBER SECTION 3.2.1.11 RESUBMISSION NUMBER SECTION 3.2.1.12 REPORT AS OF *(YYYYMMDD)* SECTION 3.2.1.13 DATE PREPARED *(YYYYMMDD)* SECTION 3.2.1.15

POC NAME (Last, First, Middle Initial) SECTION 3.2.1.14	DEPARTMENT	TELEPHONE (Include Area Code)	EMAIL ADDRESS

REMARKS SECTION 3.2.1.17

SECTION 3.1.3 UNCLASSIFIED
SECURITY CLASSIFICATION

DD Form 3026-X, MMMM 2016

*Double Click Object to see Complete Worksheet

Back-Up Slides

Cost Assessment Data Enterprise

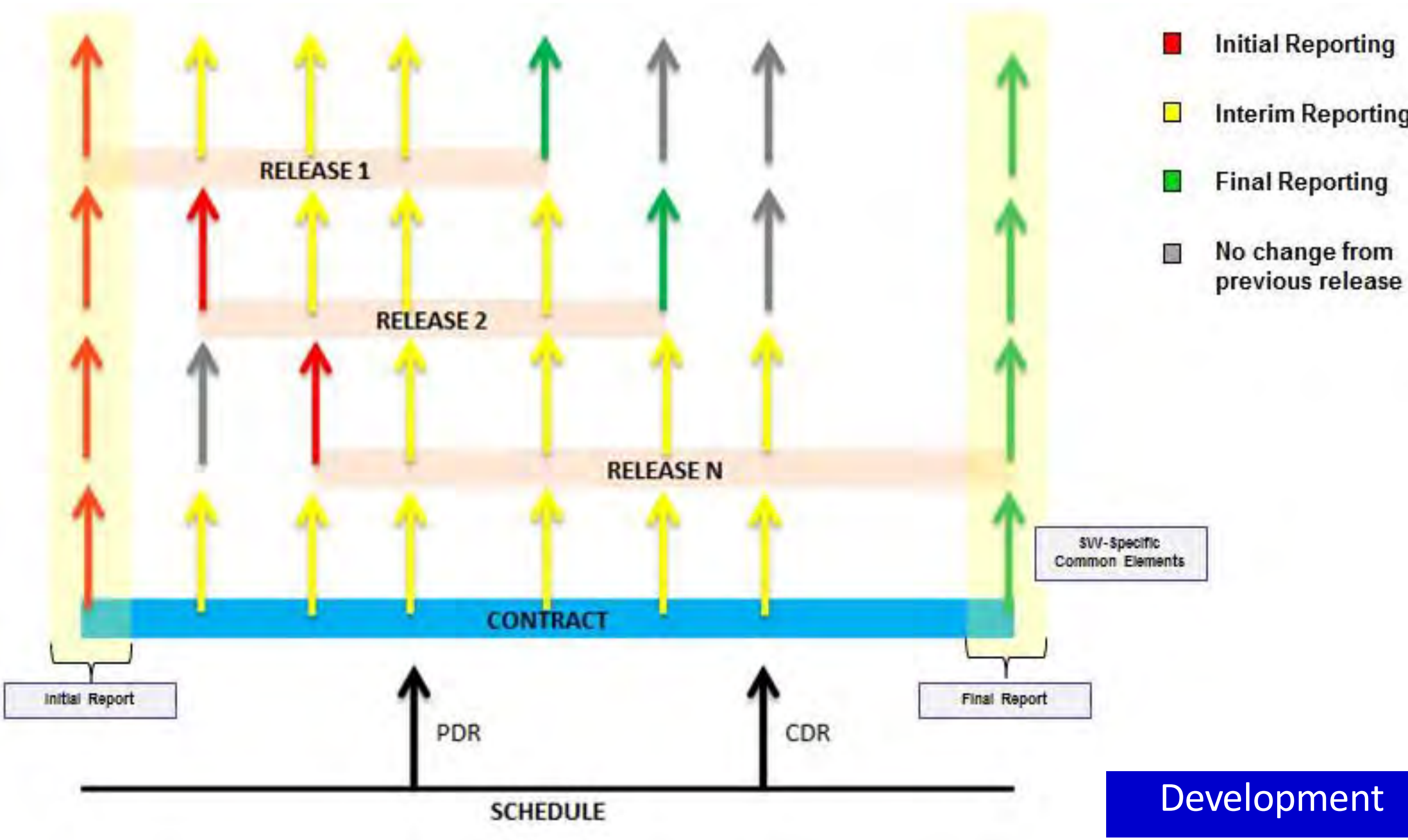
Software Reporting Frequency

Cost Assessment Data Enterprise

SRDR DID/Forms

Development Reporting Paradigm

OSD CAPE



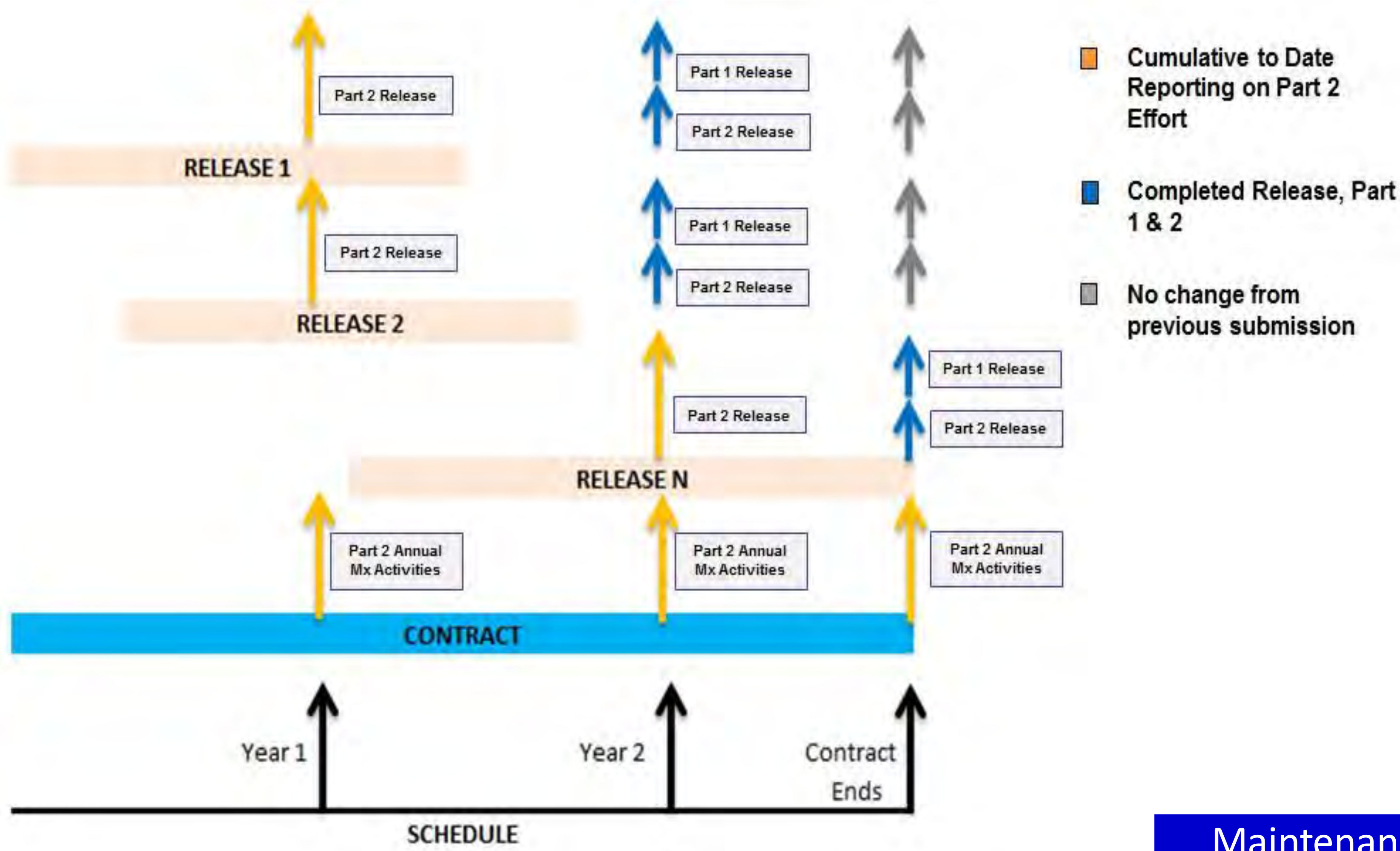
Development

Cost Assessment Data Enterprise

SRDR DID/Forms

Maintenance Reporting Paradigm

OSD CAPE



Maintenance

Government-Wide CDRL
Agile SW Metric Data Collection

Agile Project Management Controls and a central repository of Agile SW Metric Data for Government Use, Evaluation, and Analysis

William Plummer (SPAWAR 1.6)
Jeremiah Hayden (SPAWAR 1.6)
Omar Mahmoud (Cask)

August 23, 2016

Agenda

- Agile Software Metric Data Collection
- Metric Collection Team
- Data Analysis Benefactors
- Suggested SRDR Addendum
- Agile Metric Data Collection CDRL
  - Visualization and Trending Charts
- Case Study
  - Initial Baseline Estimates (Database Utilization)
  - Measuring In-Progress Performance
  - Caution of Agile Metric Usage
- CDRL Key Takeaway
- Get Involved
- Wrap Up/Questions

Agenda

- Agile Software Metric Data Collection
- Metric Collection Team
- Data Analysis Benefactors
- Suggested SRDR Addendum
- Agile Metric Data Collection CDRL
  - Visualization and Trending Charts
- Case Study
  - Initial Baseline Estimates (Database Utilization)
  - Measuring In-Progress Performance
  - Caution of Agile Metric Usage
- CDRL Key Takeaway
- Get Involved
- Wrap Up/Questions

Agile SW Metric Data Collection

- **<u>Purpose:</u>** Development of an Agile Metric Database by utilizing Agile SW CDRL that aligns with PARCA's guide to Agile and EVM System compliance

- **<u>Goal:</u>** Standardize metric reporting to inform cost estimation of Agile software development project costs across various application domains and government agencies

- **<u>Benefits:</u>** Initial project baseline cost estimates, in-progress metric analysis and projections, feature estimation, technical debt analysis, earned value estimation, CER development, and much more…

- **<u>What We Need:</u>** Government agencies to utilize Agile Metric CDRL and include the SPAWAR 1.6 cost organization as a recipient of the data (NDA to be signed)

Agenda

- Agile Software Metric Data Collection
- Metric Collection Team
- Data Analysis Benefactors
- Suggested SRDR Addendum
- Agile Metric Data Collection CDRL
  - Visualization and Trending Charts
- Case Study
  - Initial Baseline Estimates (Database Utilization)
  - Measuring In-Progress Performance
  - Caution of Agile Metric Usage
- CDRL Key Takeaway
- Get Involved
- Wrap Up/Questions

Agile SW Metric Data Gathering Team Composition	
William Plummer and Jeremiah Hayden	SPAWAR 1.6 - Special Studies Team Lead and Agile Metric Data Collection Leads
Omar Mahmoud	SPAWAR 1.6 (Contractor) Data Collection
Dan Strickland	Missile Defense Agency
Richard Mabe	Air Force Cost Analysis Agency
James Judy	Office of the Deputy Assistant Secretary of the Army for Cost & Economics
Brian Fersch	Air Force Life Cycle Management Center
Maria Goodman	Missile Defense Agency
Vjosa Dreshaj	National Geospatial-Intelligence Agency
Bakari Dale	Office of the Assistant Secretary of the Army for Financial Management & Comptroller
Pamela McDonald	Missile Defense Agency
Dr. Wilson Rosa	Naval Center for Cost Analysis
David Fersch	Office of the Assistant Secretary of the Army for Financial Management & Comptroller
James Doswell	Air Force
Kelly Hazel	Office of the Secretary of Defense Cost Assessment & Program Evaluation
Mary Anne Scully	Air Force Life Cycle Management Center

Agenda

- Agile Software Metric Data Collection
- Metric Collection Team
- $\boxed{\text{Data Analysis Benefactors}}$
- Suggested SRDR Addendum
- Agile Metric Data Collection CDRL
  - Visualization and Trending Charts
- Case Study
  - Initial Baseline Estimates (Database Utilization)
  - Measuring In-Progress Performance
  - Caution of Agile Metric Usage
- CDRL Key Takeaway
- Get Involved
- Wrap Up/Questions

Data Analysis Benefactors

- The following organizations can benefit from this and future data collection efforts:
  - Army, Navy, Air Force, Marine Corp, and other Non-DOD Government Agencies and Analysts
  - Cost Component Agencies (CAPE, NCCA, AFCAA, DASN CE, MDA, others)
  - Program Office Estimates
  - Contract Evaluation
  - Possible SRDR Addendum for Agile Metric Collection

Agenda

- Agile Software Metric Data Collection
- Metric Collection Team
- Data Analysis Benefactors
- Suggested SRDR Addendum
- Agile Metric Data Collection CDRL
  - Visualization and Trending Charts
- Case Study
  - Initial Baseline Estimates (Database Utilization)
  - Measuring In-Progress Performance
  - Caution of Agile Metric Usage
- CDRL Key Takeaway
- Get Involved
- Wrap Up/Questions

Suggested SRDR Addendum

SOFTWARE DEVELOPMENT REPORT, FORMAT 1 (DD FORM 2630-D): Part 1 Software Development Technical Data, Release-CSCI Level SECTION 3.3.2			
Release ID SECTION 3.3.2.1.1	1	Release Name SECTION 3.3.2.1.2	Release 1
CSCI ID SECTION 3.3.2.2.1	FCS-1	CSCI Name SECTION 3.3.2.2.2	Flight Control Software CSCI 1
WBS Element Code SECTION 3.3.2.3.1	1.1.3.2.2	WBS Element Name SECTION 3.3.2.3.2	Flight Control Software Release 1 CSCI 1
Outsourced Development Organizations SECTION 3.3.2.4			
Name SECTION 3.3.2.4.1	Primary SECTION 3.3.2.4.3	Location SECTION 3.3.2.4.2	
Outsourced Development comment SECTION 3.3.2.4.4			
Name	Primary	Location	
Outsourced Development comment			etc.
Product and Development Description SECTION 3.3.2.5			
Functional Description SECTION 3.3.2.			
Software Development Characterization SECTION 3.3.2.5.2			
Software State of Development (*Check one only*) SECTION 3.3.2.5.3	Prototype	Production-Ready	Mix
Operating Environment(s) (*Check all that apply*) SECTION 3.3.2.5.4			
Surface Fixed	Surface Vehicle	Ordnance Systems	Other
Surface Mobile	Air Vehicle	Missile Systems	If Other, provide explanation
Surface Portable	Sea Systems	Space S	
SECTION 3.3.2.5.5			
Manned	Unmanned		
Primary Application Domain (*Check one only*) SECTION 3.3.2.5.6			
Microcode and Firmware	Communication		Software Tools
Signal Processing	System Software		Mission Planning
Vehicle Payload	Process Control		Custom AIS Software
Vehicle Control	Scientific and Simulation		Enterprise Service System
Other Real-Time Embedded	Test, Measurement, and Diagnostic Equipment		Enterprise Information System
Command and Control	Training		
Application Domain Comments			
Development Process SECTION 3.3.2.5.7			
Software Development Method SECTION 3.3.2.5.8			
This product is a new development	SECTION 3.3.2.5.9		
Software Reuse SECTION 3.3.2.5.10			
Name	Description		

SRDR Addendum for Estimating SW Dev Costs on Agile Projects

- Agile Product Roadmap
  - Establishes the PMB
- Features Completed per Sprint
  - Earned Value
- Total Issues in Backlog per Sprint
  - Scope Creep
- Monthly Cost Expenditures (EAC)
  - Estimates for Future Conditions
- Sprint Length (weeks)
  - Schedule Estimate Time-box
- Non-Story Related Issues
  - Technical Debt Ratio

Agenda

- Agile Software Metric Data Collection
- Metric Collection Team
- Data Analysis Benefactors
- Suggested SRDR Addendum
- Agile Metric Data Collection CDRL
  - Visualization and Trending Charts
- Case Study
  - Initial Baseline Estimates (Database Utilization)
  - Measuring In-Progress Performance
  - Caution of Agile Metric Usage
- CDRL Key Takeaway
- Get Involved
- Wrap Up/Questions

How can YOU assist???

Agile Metric Data Collection CDRL

Quantitative Analysis

Qualitative Assessment/Baseline

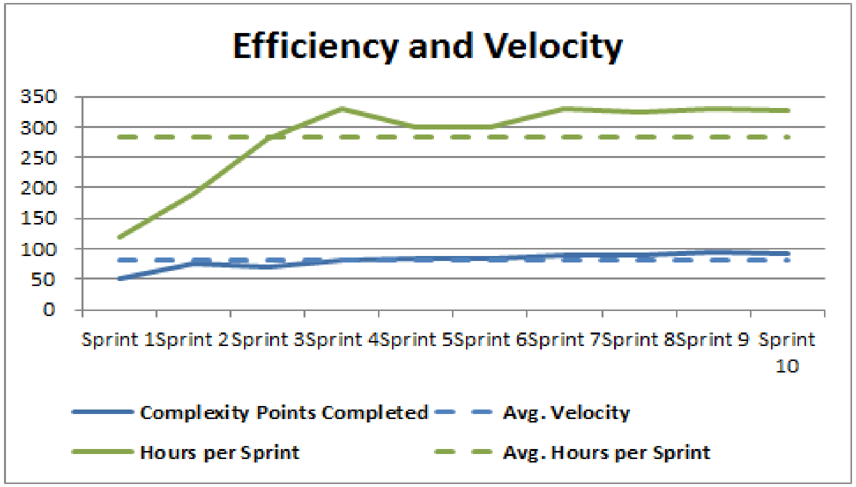
Step 1 - Application Domain:	Command & Control
Step 2 - DoD Branch:	Army
Step 3 - ACAT:	ACAT II
Step 4 - Program Name:	Weather Tracking and Visualization
Step 5 - Contract Type:	Cost Plus Incentive Fee
Step 6 - SW Dev Organization (Company Name):	ACME Developers
Step 7 - CMMI Level SW Dev Organization (Company Name):	CMMI Level 4
Step 8 - Financial Base Year:	2015
Step 9 - Weeks per Sprint	2
Step 10 - Total Number of Features/Reqs. Planned	40
Step 11 - Number of Planned Sprints	25

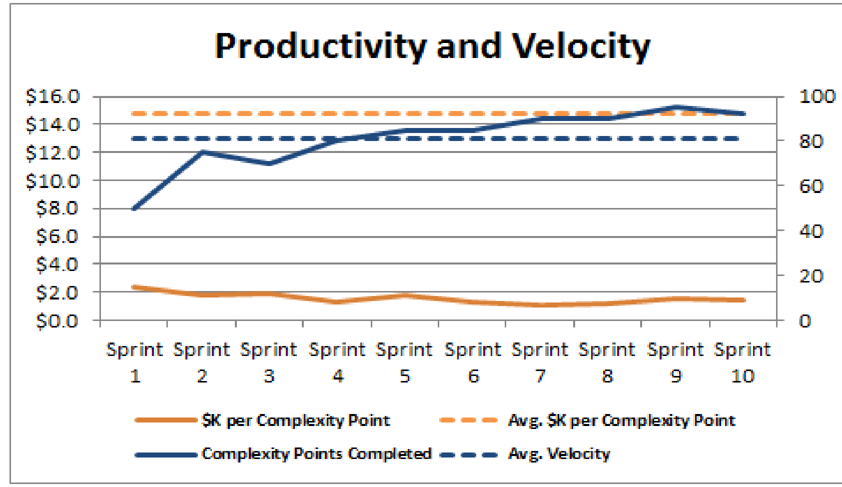
Step 12 - Layout the Agile Metrics by Sprint:

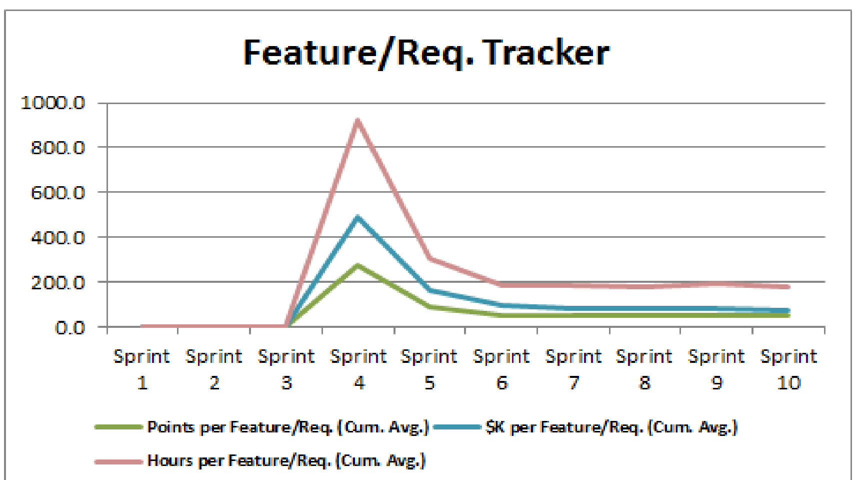
Agile Metric Data Collection (per Sprint)	Sprint 1	Sprint 2	Sprint 3	Sprint 4	Sprint 5	Sprint 6	Sprin					
Complexity Points Completed	50	75	70	80	85	85	90					
# Features/Reqs. Completed	0	0	0	1	3	4	2					
# of Stories Completed	17	14	30	22	34	28	19					
# of Defect/Bug Completed	1	2	1	5	6	2	10					
# of Enhancement Completed	0	1	4	10	2	8	5					
# of Test Completed	0	0	7	5	1	3	6					
# of (Other Issues) Completed	0	0	1	1	2	0	0					
# of FTEs	5.0	5.0	5.0	6.0	7.0	7.0	7.0					
# Total Backlog Issues	500	482	465	423	403	360	319					
$K per Sprint	$120.0	$131.0	$137.0	$104.0	$150.0	$114.0	$98.[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
Hours per Sprint	120	190	280	330	300	300	330	325	330	328	2,833.0	283.3
Avg. Velocity	81.2	81.2	81.2	81.2	81.2	81.2	81.2	81.2	81.2	81.2	812.0	81.2
Actual Feature/Req. Burndown	40	40	40	39	36	32	30	28	27	24	N/A	N/A
Planned Feature/Req. Burndown per Sprint	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	N/A	N/A
Linear Feature/Req. Burndown (Expected)	38.4	36.8	35.2	33.6	32.0	30.4	28.8	27.2	25.6	24.0	N/A	N/A
Avg. Features/Reqs. Completed per Sprint	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	16.0	1.6
Points per Feature/Req. (Cum. Avg.)	0.0	0.0	0.0	275.0	90.0	55.6	53.5	52.1	55.4	50.8	N/A	63.2
$K per Feature/Req. (Cum. Avg.)	$0.0	$0.0	$0.0	$492.0	$160.5	$94.5	$85.4	$80.2	$85.5	$78.0	N/A	$1,141.0
Avg. $K per Feature/Req.	$1,141.0	$1,141.0	$1,141.0	$1,141.0	$1,141.0	$1,141.0	$1,141.0	$1,141.0	$1,141.0	$1,141.0	N/A	$11,410.3
$K per Complexity Point	$2.4	$1.7	$2.0	$1.3	$1.8	$1.3	$1.1	$1.2	$1.6	$1.5	N/A	$14.7
Avg. $K per Complexity Point	$14.7	$14.7	$14.7	$14.7	$14.7	$14.7	$14.7	$14.7	$14.7	$14.7	N/A	$147.3
$ per Hour	$1,000.0	$689.5	$489.3	$315.2	$500.0	$380.0	$297.0	$332.3	$451.5	$417.7	N/A	$4,262.4
Avg. Hours per Sprint	283.3	283.3	283.3	283.3	283.3	283.3	283.3	283.3	283.3	283.3	2,833.0	283.3
Hours per Point (Cum. Avg.)	2.4	2.5	3.0	3.3	3.4	3.4	3.5	3.5	3.5	3.5	N/A	3.2
Complexity Points per Hour (Cum. Avg.)	0.4	0.4	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	3.2	0.3
Avg. Complexity Points per Hour	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	N/A	0.3

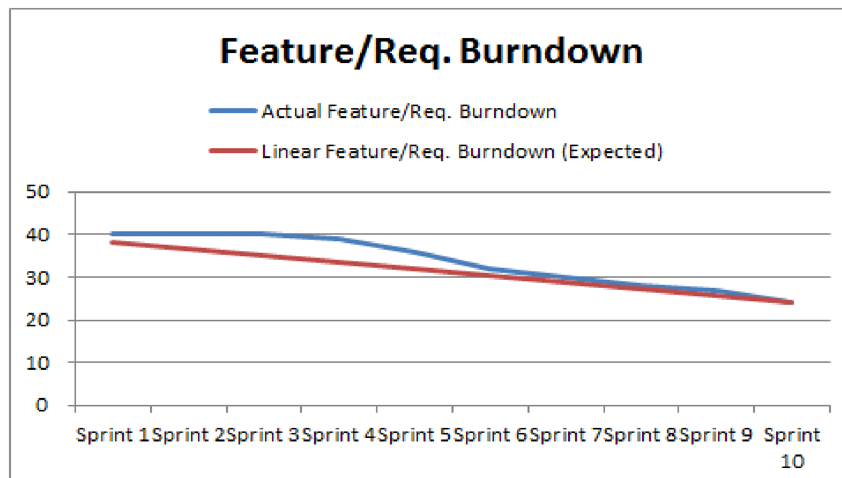
Utilize the CDRL and provide results to SPAWAR 1.6 for database development and metric consolidation

Visualization and Trending Charts

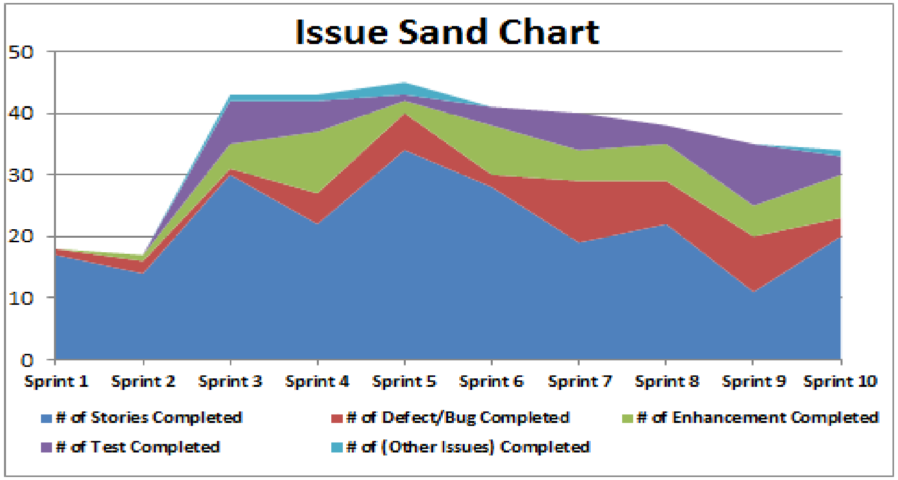


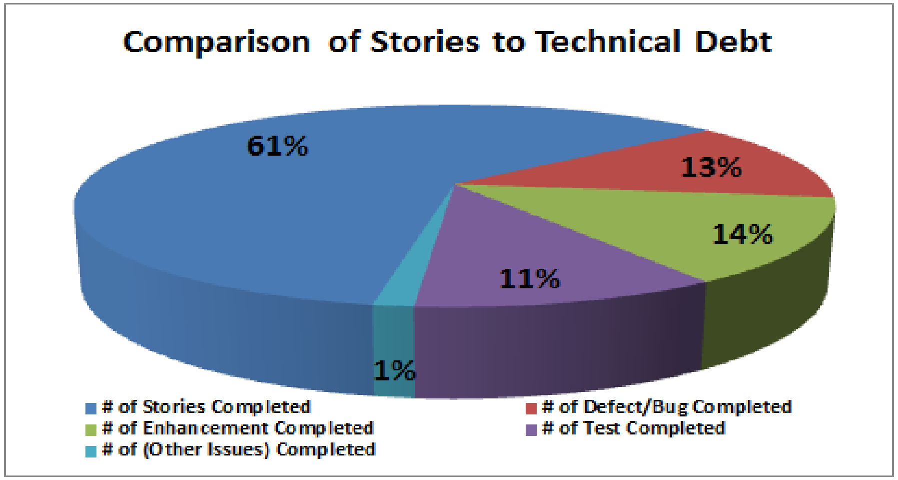


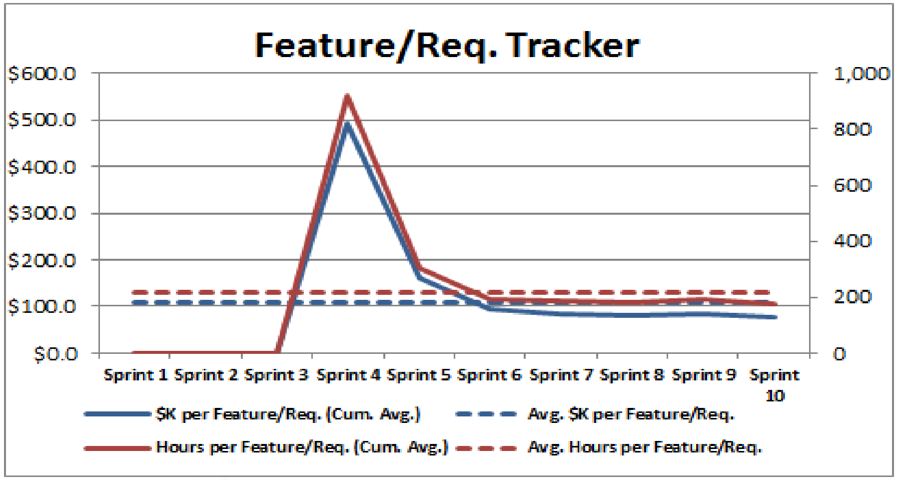




Visualization and Trending Charts (cont'd)







Agenda

- Agile Software Metric Data Collection
- Metric Collection Team
- Data Analysis Benefactors
- Suggested SRDR Addendum
- Agile Metric Data Collection CDRL
  - Visualization and Trending Charts
- Case Study
  - Initial Baseline Estimates (Database Utilization)
  - Measuring In-Progress Performance
  - Caution of Agile Metric Usage
- CDRL Key Takeaway
- Get Involved
- Wrap Up/Questions

How can YOU benefit from the Agile Database and CDRL Template???

TEAM SPAWAR

Case Study – Initial Baseline Estimates
(Database Dashboard Utility)

Application Domain: -->	Radar
ACAT: -->	ACAT I
CMMI: -->	5
Contract Type: -->	CPIF

Schedule Productivity:--> Hours/Feature

Low	Exp	High
80.0	100.0	120.0

Cost Productivity:--> Cost/Feature (BY16$K)

Low	Exp	High
$30	$45	$60

Artifact Trend:--> Requirement (Effort)

Low	Exp	High
50%	65%	75%

Input: Development Environment

Output: Database Metric Range for Initial Baseline Estimates

<u>Formulate Initial Baseline Estimates</u>
Total Project Cost: $5,000,000
Schedule: 20 Sprints (2 weeks/Sprint)
Anticipated Technical Debt: 35%

Case Study – Measuring In-Progress Performance

- Project A:

Background: Agile SW project issued to ACME Contracting with 3 years of experience working in an Agile environment.

SCRUM Master recently received certification from an 8 hour online Agile webinar.

  - Command and Control Project developed for the Air Force
  - ACAT III on a CPIF Contract
  - CMMI Level 3

- Project B:

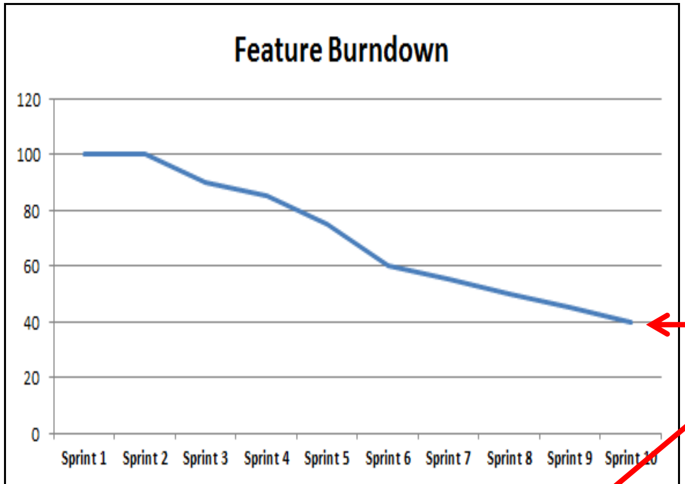
Background: Agile SW project issued to Prestine Developers with over 10 years of experience working in an Agile environment.

SCRUM Master has 15 years of experience leading organized teams.

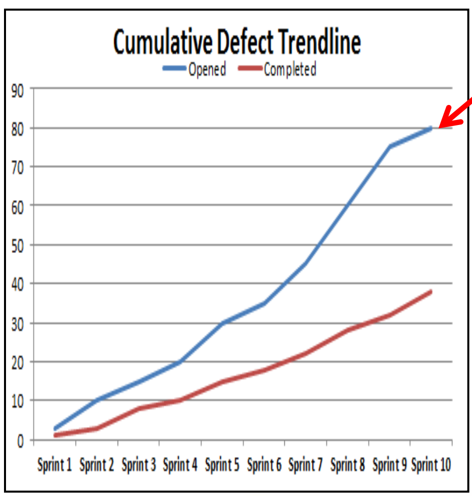
  - Radar Project for the NAVY
  - ACAT II on a CPAF Contract
  - CMMI Level 4

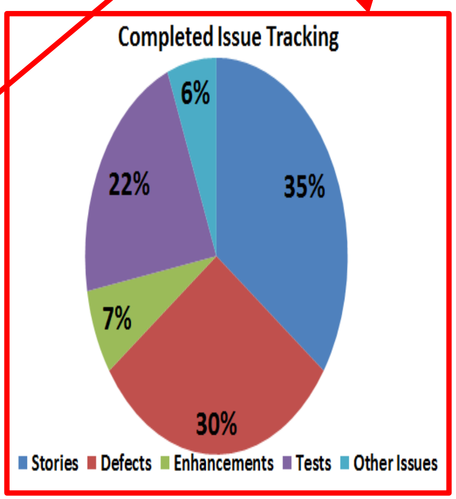
How do these projects measure up and what information can you glean from their metrics?

Project A:



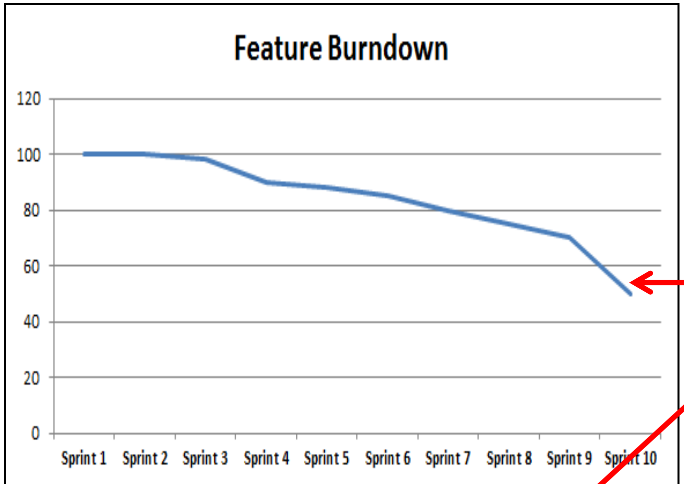
Ahead in completed features, but notice technical debt growth



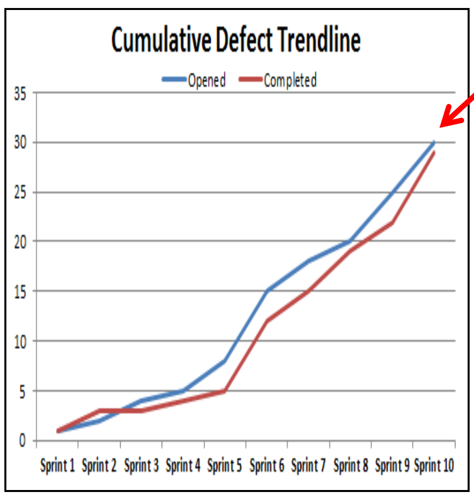


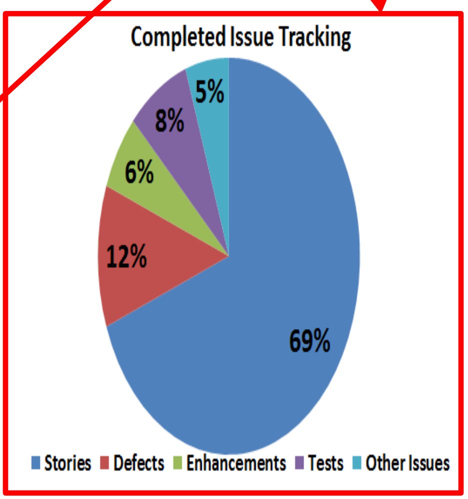
Technical Debt reflects 65% of the total effort

Project B:



Better balance of overall project quality





Technical Debt reflects 31% of the total effort

Case Study - *Caution when Using Agile Metrics*

- Caution should be used when using Agile related metrics to baseline or estimate other projects
  - Story points are subjective and relative
    - Defined by each development team
    - Particular to teams history and skillset
  - Sprint timelines may vary across organizations
- Considerations:
  - Velocity is fairly stable after 3 sprints
    - Agile teams experience “storming and norming” productivity
  - Project Performance should be estimated by Hours and/or Cost expended per Completed Feature
  - Story points estimates can be used if the point definitions, scope of effort, and development team are analogous

Agenda

- Agile Software Metric Data Collection
- Metric Collection Team
- Data Analysis Benefactors
- Suggested SRDR Addendum
- Agile Metric Data Collection CDRL
  - Visualization and Trending Charts
- Case Study
  - Initial Baseline Estimates (Database Utilization)
  - Measuring In-Progress Performance
  - Caution of Agile Metric Usage
- $\boxed{\text{CDRL Key Takeaway}}$
- Get Involved
- Wrap Up/Questions

Key Takeaway of the CDRL Template

- Monthly EVM reporting and various trending and estimates on a Sprint by Sprint basis
- More robust government database to be shared across government organizations
- Allows for key factors that can be used in future estimating analysis:
  - $/hr, $/feature, $/sprint, % of Requirements vs. Technical Debt, and Buffering for unplanned effort
- Broad dataset will provide insight into a large array of programs

Agenda

- Agile Software Metric Data Collection
- Metric Collection Team
- Data Analysis Benefactors
- Suggested SRDR Addendum
- Agile Metric Data Collection CDRL
  - Visualization and Trending Charts
- Case Study
  - Initial Baseline Estimates (Database Utilization)
  - Measuring In-Progress Performance
  - Caution of Agile Metric Usage
- CDRL Key Takeaway
- Get Involved
- Wrap Up/Questions

Get Involved

To participate in the database development or for a copy of the Agile Metric CDRL

Reach out to:

Jeremiah Hayden (jeremiah.hayden@navy.mil)

William Plummer (william.s.plummer@navy.mil)

Agenda

- Agile Software Metric Data Collection
- Metric Collection Team
- Data Analysis Benefactors
- Suggested SRDR Addendum
- Agile Metric Data Collection CDRL
  - Visualization and Trending Charts
- Case Study
  - Initial Baseline Estimates (Database Utilization)
  - Measuring In-Progress Performance
  - Caution of Agile Metric Usage
- CDRL Key Takeaway
- Get Involved
- Wrap Up/Questions

Wrap up Questions

Back up

Terms Used in Presentation

- **Buffering:** Planning effort for unplanned events due to changes in requirements, scope creep, defects, or usability assessments and feedback
- **Human Systems Integration (HSI)**: Describes possible usability issues with the software
- **Issue/Artifact**: Detailed description of a feature of the software that needs to be developed, tested, integrated, QA'ed, etc.
- **Product Backlog/Backlog**: Database or list that records all Issues or Artifacts the software team will work on. Categories include: Planned Sprint, Story Points, Resource, Issue type, Status, etc.
- **Sprint**: Fixed-time box in which SW dev activity occurs (typical ranges are 2 or 4 weeks)
- **Story Points/Points**: Quantitative-subjective measure of the requirement to fully plan, develop, test, and integrate a feature into the overall software development deliverable
- **Technical Debt**: Refers to other non-requirement related activities that are incurred on the project as a result of requirements development (e.g., Defects, Enhancements, HSI, etc.)
- **Velocity**: Schedule productivity typically described as "completed points per sprint"

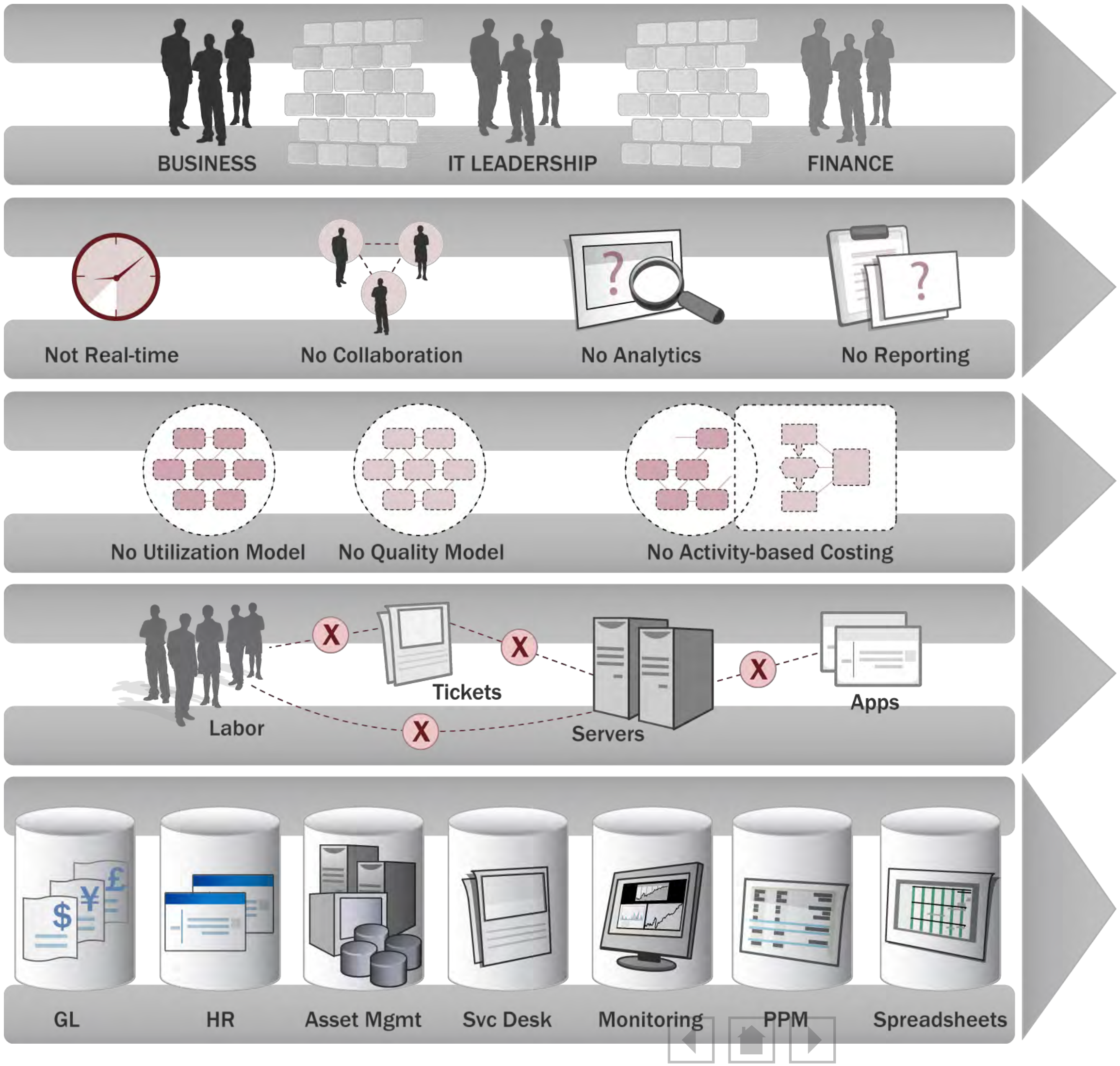
vRealize Business™

Presentation

Agenda

- Overview

The Challenge Today



Siloed teams relying on disconnected data

Point in time, manual data to make decisions

Fragmented views of cost, performance and forecast of services

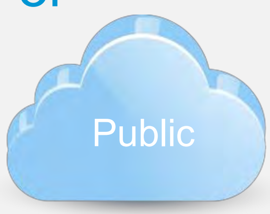
No mapping of the relationships between asset, service, labor and performance info

Many, disparate data sources

Common IT Business Management Problems

Decisions Made by Gut Feel	Treated Like Free, Unlimited Resource	Difficult to Explain IT Costs

or



IT Consumption

IT Costs

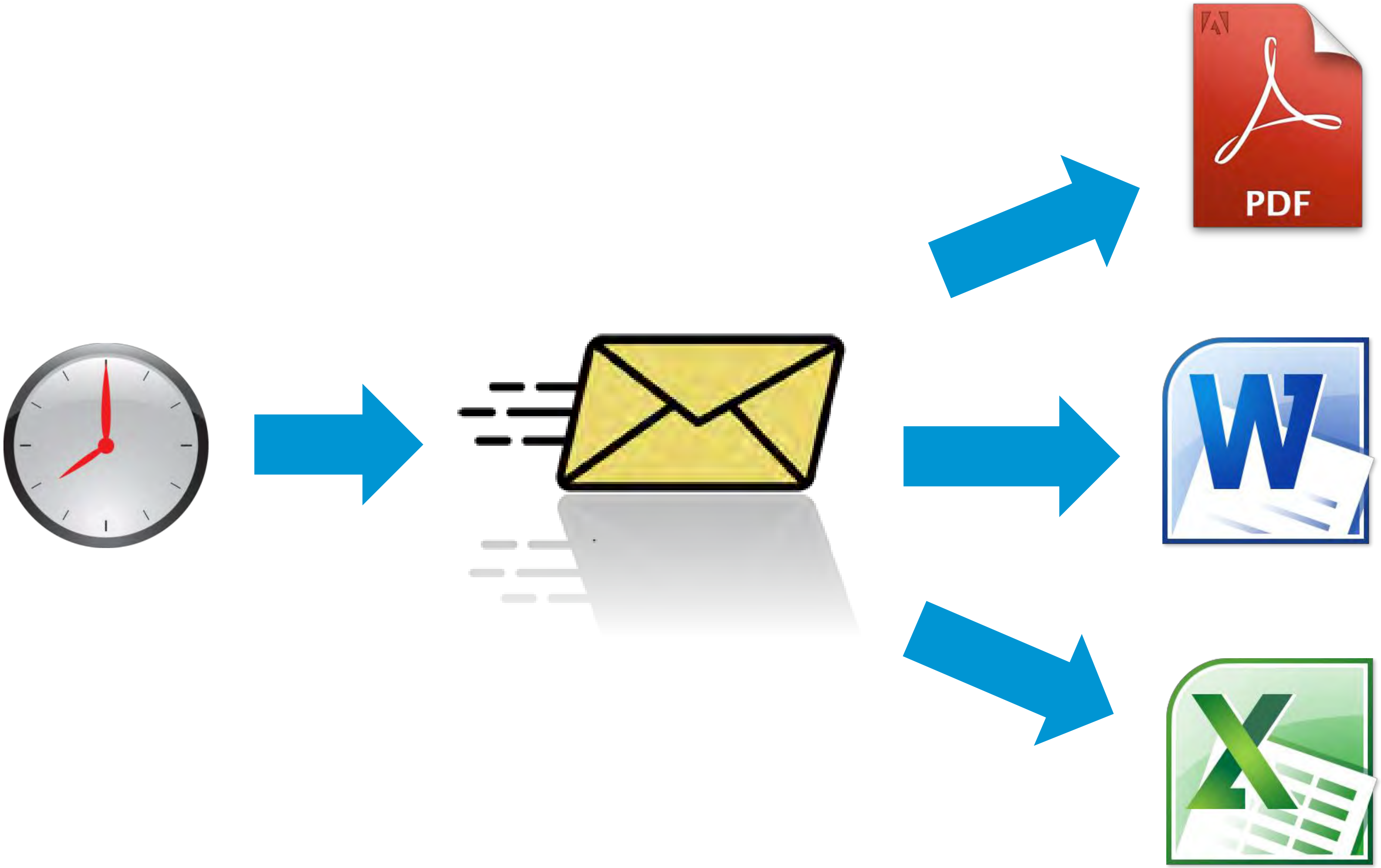
vmware®


AND WARREN HERE IS IN CHARGE OF OUR GUT FEELINGS

IT Business Management Solutions

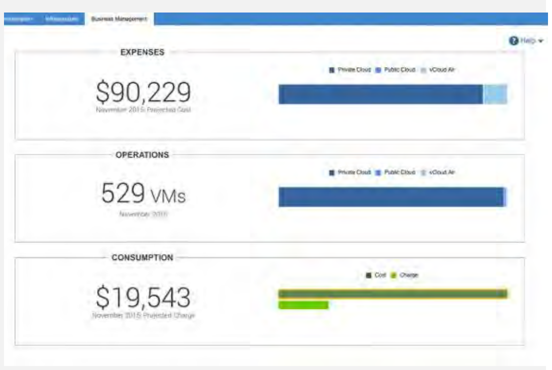
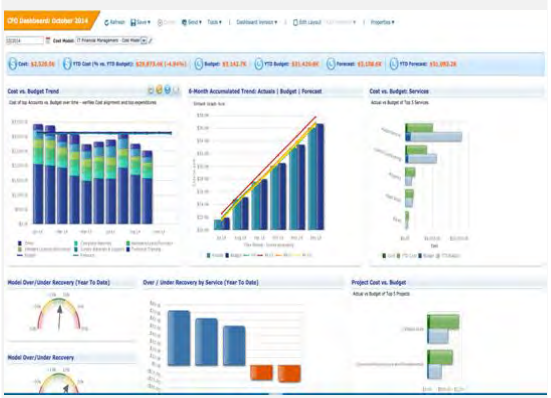
Reporting	Cost Modeling	Data Management
		

Communicating Across The Organization

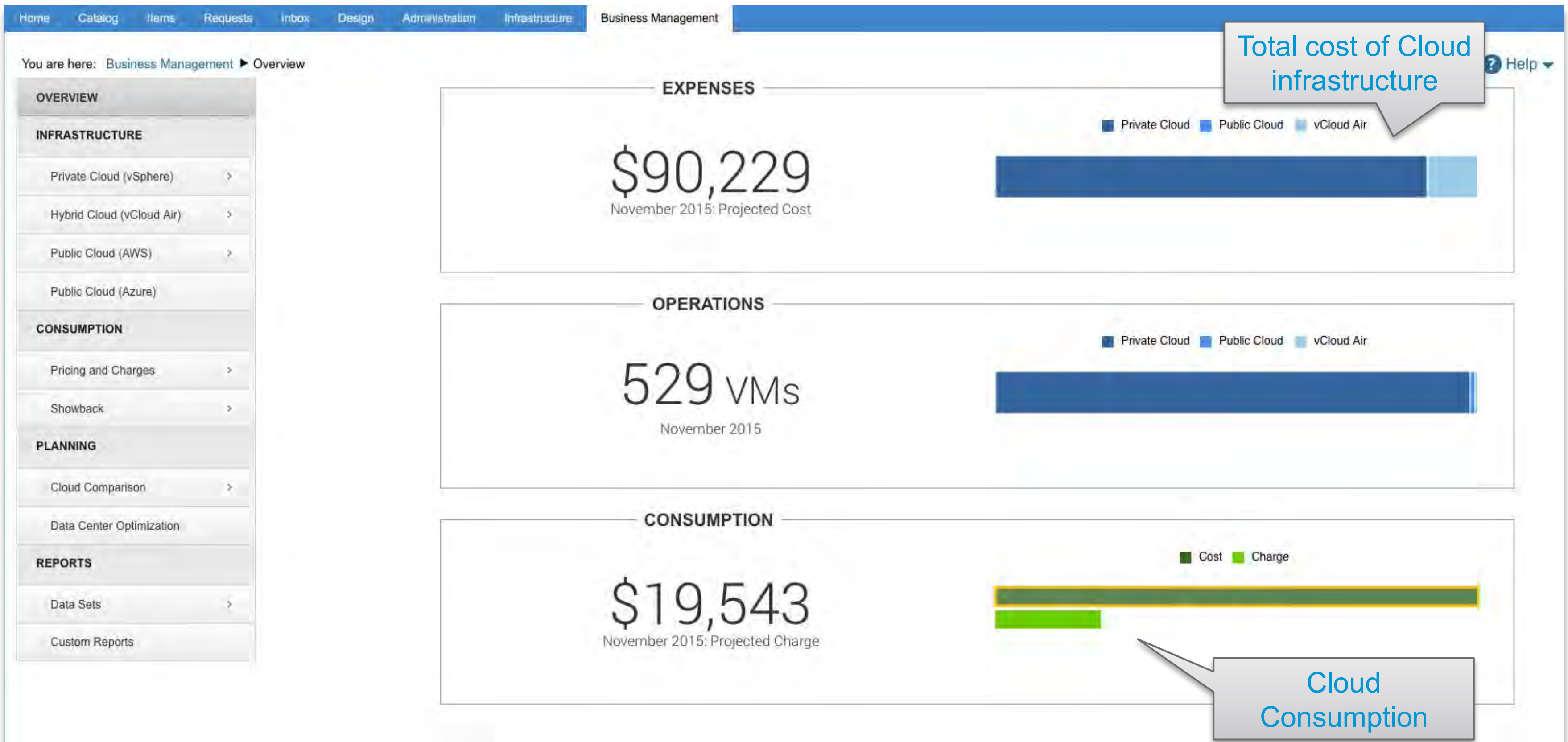


vRealize Business Approach

vRealize Business Primary Use Cases

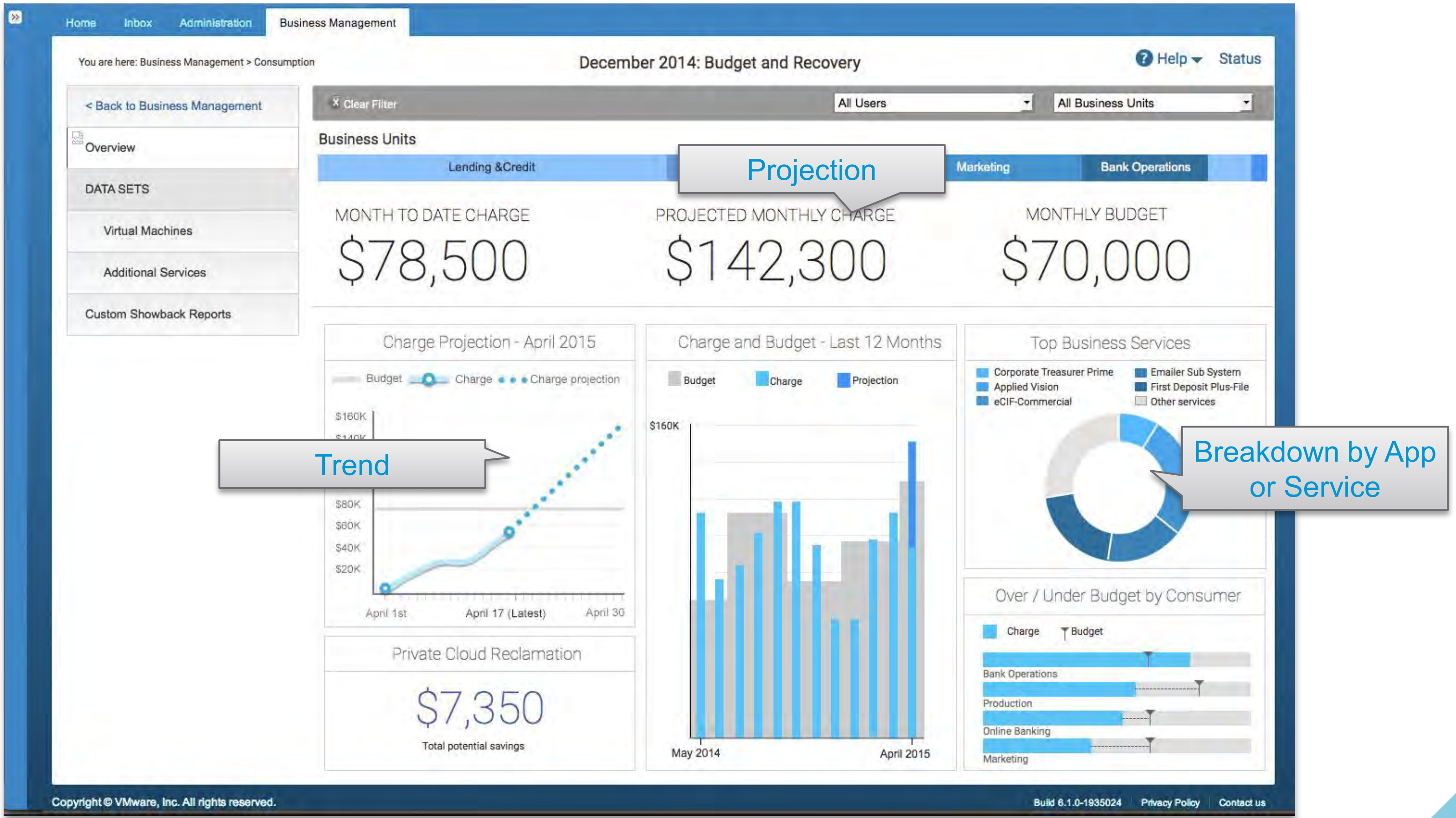
Private Cloud Costing	Cloud Business Management	Enterprise IT Business Management
		
CRAWL	WALK	RUN
vRealize Business for Cloud Standard	vRealize Business for Cloud Advanced	vRealize Business Enterprise

Cost Overview on a Single Pane of Glass



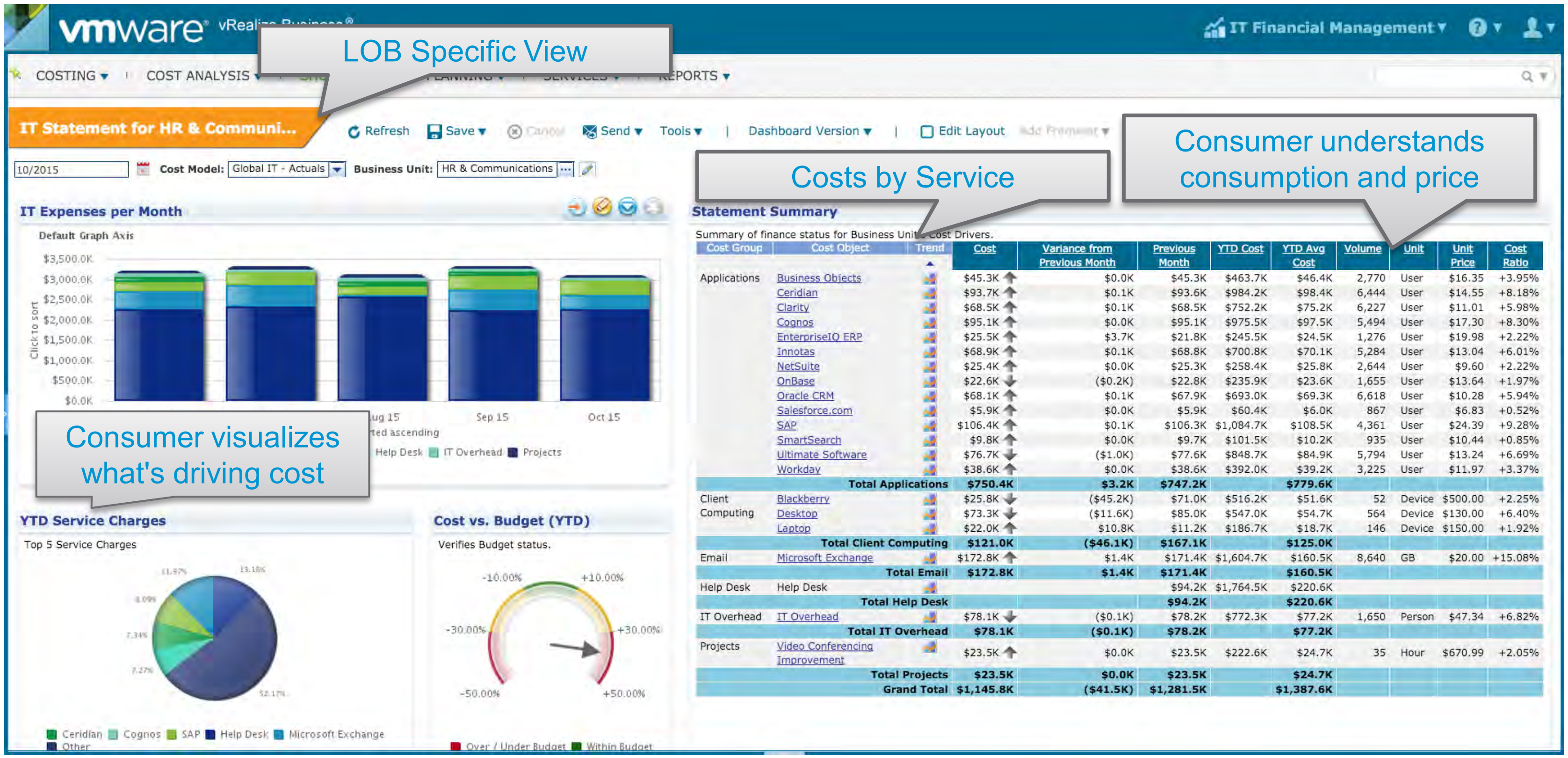
vmware®

Showback by LOB/Customer



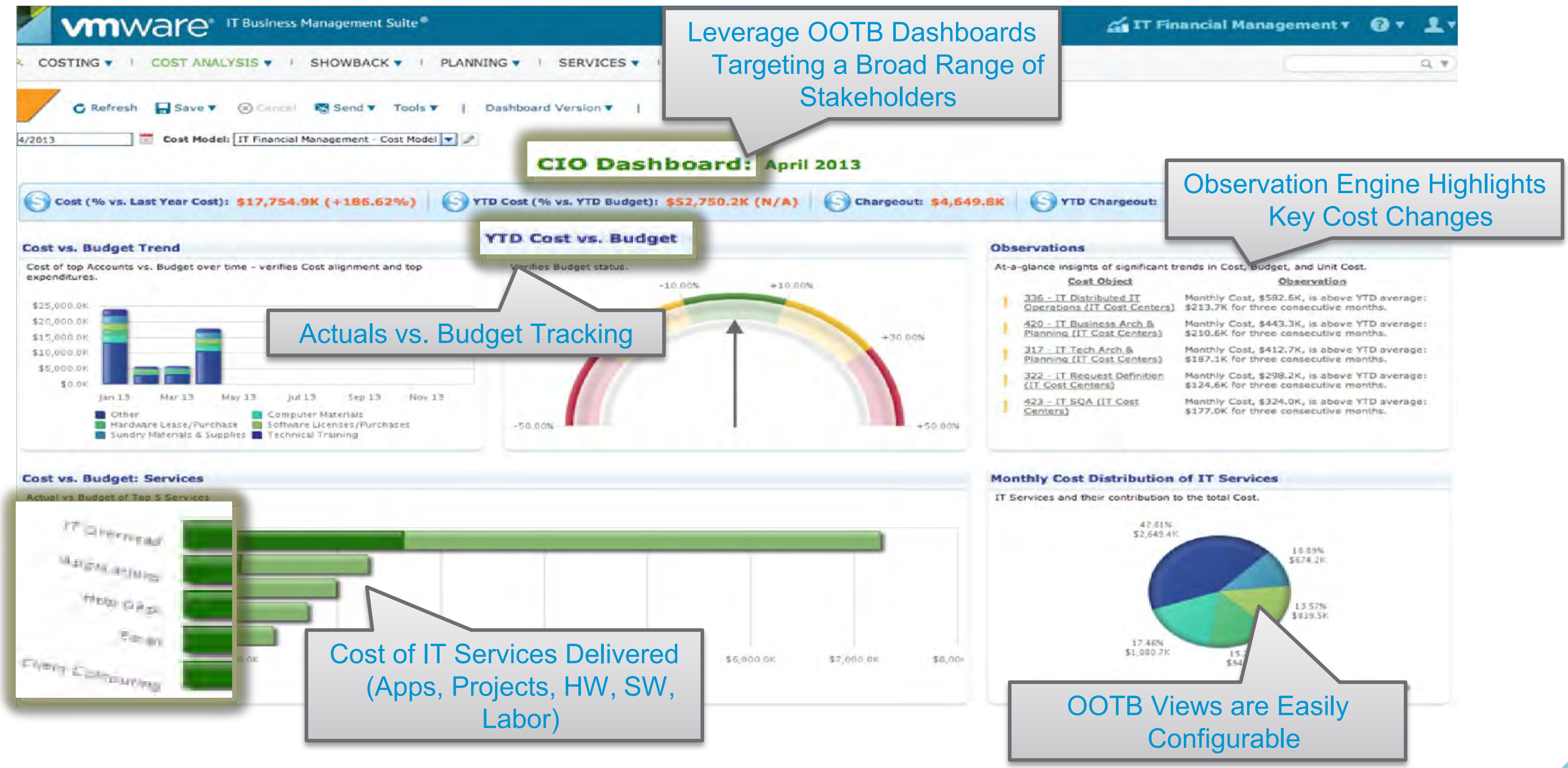
vmware®

Bill of IT for the Full Portfolio of IT Services



Cost Group	Cost Object	Cost	Variance from Previous Month	Previous Month	YTD Cost	YTD Avg Cost	Volume	Unit	Unit Price	Cost Ratio
Applications	Business Objects	$45.3K	$0.0K	$45.3K	$463.7K	$46.4K	2,770	User	$16.35	+3.95%
	Ceridian	$93.7K	$0.1K	$93.6K	$984.2K	$98.4K	6,444	User	$14.55	+8.18%
	Clarity	$68.5K	$0.1K	$68.5K	$752.2K	$75.2K	6,227	User	$11.01	+5.98%
	Cognos	$95.1K	$0.0K	$95.1K	$975.5K	$97.5K	5,494	User	$17.30	+8.30%
	EnterpriseIQ ERP	$25.5K	$3.7K	$21.8K	$245.5K	$24.5K	1,276	User	$19.98	+2.22%
	Innotas	$68.9K	$0.1K	$68.8K	$700.8K	$70.1K	5,284	User	$13.04	+6.01%
	NetSuite	$25.4K	$0.0K	$25.3K	$258.4K	$25.8K	2,644	User	$9.60	+2.22%
	OnBase	$22.6K	($0.2K)	$22.8K	$235.9K	$23.6K	1,655	User	$13.64	+1.97%
	Oracle CRM	$68.1K	$0.1K	$67.9K	$693.0K	$69.3K	6,618	User	$10.28	+5.94%
	Salesforce.com	$5.9K	$0.0K	$5.9K	$60.4K	$6.0K	867	User	$6.83	+0.52%
	SAP	$106.4K	$0.1K	$106.3K	$1,084.7K	$108.5K	4,361	User	$24.39	+9.28%
	SmartSearch	$9.8K	$0.0K	$9.7K	$101.5K	$10.2K	935	User	$10.44	+0.85%
	Ultimate Software	$76.7K	($1.0K)	$77.6K	$848.7K	$84.9K	5,794	User	$13.24	+6.69%
	Workday	$38.6K	$0.0K	$38.6K	$392.0K	$39.2K	3,225	User	$11.97	+3.37%
	Total Applications	**$750.4K**	**$3.2K**	**$747.2K**		**$779.6K**				
Client Computing	Blackberry	$25.8K	($45.2K)	$71.0K	$516.2K	$51.6K	52	Device	$500.00	+2.25%
	Desktop	$73.3K	($11.6K)	$85.0K	$547.0K	$54.7K	564	Device	$130.00	+6.40%
	Laptop	$22.0K	$10.8K	$11.2K	$186.7K	$18.7K	146	Device	$150.00	+1.92%
	Total Client Computing	**$121.0K**	**($46.1K)**	**$167.1K**		**$125.0K**				
Email	Microsoft Exchange	$172.8K	$1.4K	$171.4K	$1,604.7K	$160.5K	8,640	GB	$20.00	+15.08%
	Total Email	**$172.8K**	**$1.4K**	**$171.4K**		**$160.5K**				
Help Desk	Help Desk			$94.2K	$1,764.5K	$220.6K				
	Total Help Desk			**$94.2K**		**$220.6K**				
IT Overhead	IT Overhead	$78.1K	($0.1K)	$78.2K	$772.3K	$77.2K	1,650	Person	$47.34	+6.82%
	Total IT Overhead	**$78.1K**	**($0.1K)**	**$78.2K**		**$77.2K**				
Projects	Video Conferencing Improvement	$23.5K	$0.0K	$23.5K	$222.6K	$24.7K	35	Hour	$670.99	+2.05%
	Total Projects	**$23.5K**	**$0.0K**	**$23.5K**		**$24.7K**				
	Grand Total	**$1,145.8K**	**($41.5K)**	**$1,281.5K**		**$1,387.6K**				

Complete Visibility into All of IT Cost



Software Cost Estimation Meets Software Diversity COCOMO III

Barry Boehm, Brad Clark

Software and IT-CAST Meeting

August 23, 2016

Estimation Meets Diversity

- Sources of Software Diversity
  - A Short History of Software Estimation Accuracy
  - Process, Product, Property, and People Drivers
- Options for Software Cost Estimation
  - Expert Judgment/Consensus; Size-Based; Productivity-Based; Component-Based; Process-Based; Composites
- Best Fits of Estimation-Types to Diversity-Types
  - Extensions of ICSM Common Cases
- Charting Your Path to Improved Estimates
  - Current COCOMO III Strategy Overview

COCOMO III Project

The purpose of the COCOMO® III project is to develop a software cost estimation model for modern software development.

- Address the scope of modern software projects
  - **Subset of the estimation diversity challenges**
- Improve the accuracy and realism of estimates
- Estimate software cost that is complementary with a COSYSMO system engineering cost estimate
- Improve the value of COCOMO® in decision-making
- Create a strategy for maintaining past COCOMO® models

COCOMO III Use Cases

1. Top-level estimate
2. Multiple component estimate
3. Analysis of alternatives
4. Analysis with Size-Effort-Schedule as independent variables
5. Estimation for different processes
6. Lifecycle cost estimates
7. Legacy system transformation
8. Estimate using COCOMO® III and COSYSMO together
9. Alternative size measures
10. Local calibration

COCOMO III Software Cost Estimation

- COCOMO is the most widely used software cost estimation model in the world
  - Registered Trademark for protect the intellectual property
  - Model is open and free for anyone to use
  - Has been commercialized
- It has been 16 years since the model has been updated and calibrated to new Software Engineering data
- What we are looking for:
  - Your ideas on how the new model should be used and new input parameters to estimate software engineering development costs
  - Your chance to influence the new COCOMO III model
- Why you should consider participating:
  - Review of COCOMO III model
  - If you contribute data for model calibration, you will receive:
    - An advanced copy of the new model
    - Comparison of your data with respect to other data points used to calibrate the model

PLEASE JOIN US AT THE WORKSHOP

Process-Related Effort Estimating Relationships for Software Cost Estimating

August 2016

Presenter: Nicholas Lanham

Authors: Corinne Wallshein,
Nicholas Lanham, and Wilson Rosa
Naval Center for Cost Analysis
nicholas.lanham@navy.mil
corinne.wallshein@navy.mil
wilson.rosa@navy.mil

Outline

- Introduction
- Experimental Design
- Data Analysis
- Descriptive Statistics
- Effort Models
- Conclusion

Frequently Used Terms

- **CPAF**: Cost Plus Award Fee (type of contract)
- **CPIF**: Cost Plus Incentive Fee (type of contract)
- **CPFF**: Cost Plus Fixed Fee (type of contract)
- **Development Process:** The process used to develop software (e.g., Waterfall, Incremental, Iterative, Agile, etc.)

Introduction

Study Design

- Highlight elements of project size from paired initial-final Computer Software Configuration Item (CSCI) records to estimate using new categories

- Perform statistical analysis on sizing parameters such as
  - Estimated requirement counts
  - Estimated source lines of code (SLOC)
  - Estimated effort
  - Estimated duration
  - Estimated peak staffing

Problem Statement

- Software effort estimates should take into account requirement, process, and tool evolution
  - **Cost-plus contract types** recommended for exploratory studies, demonstrations, and development
  - **Development processes** also recommended for exploratory studies, demonstrations, and development

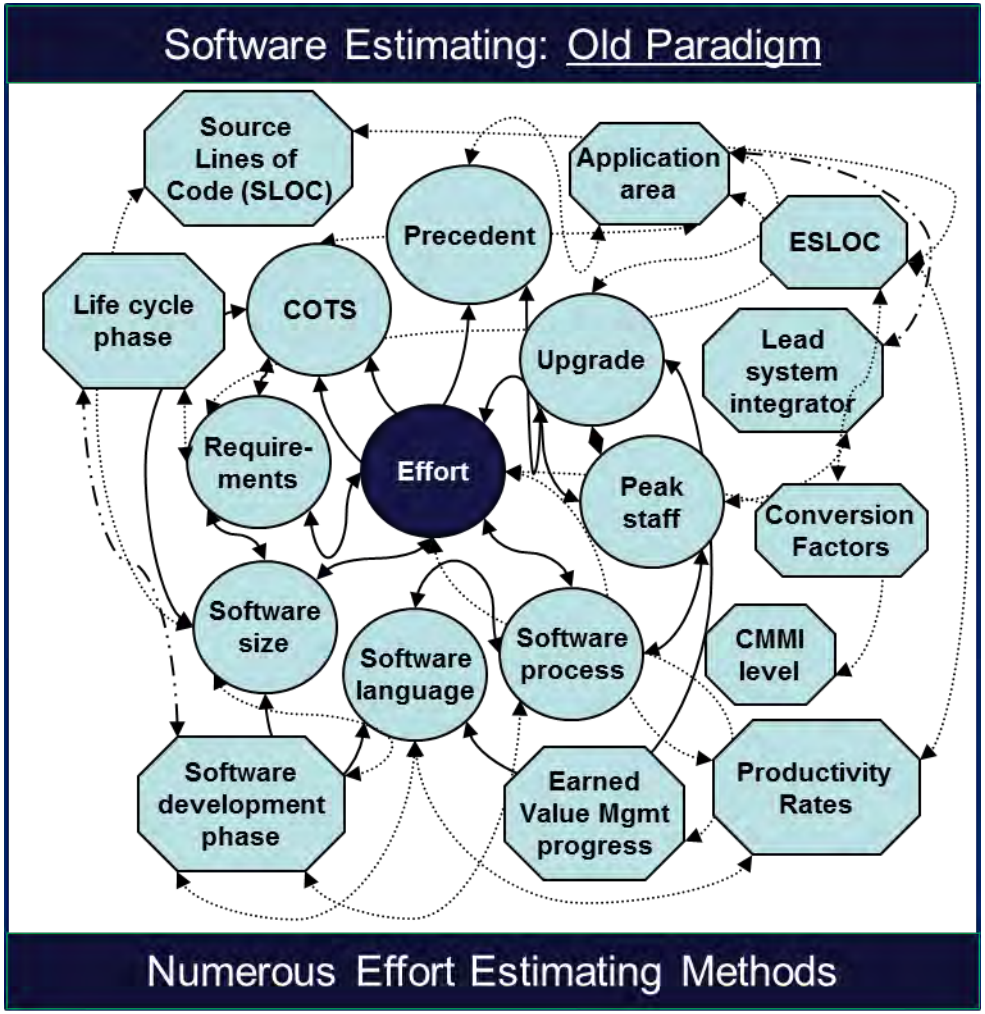
Research Questions

1) By using process-oriented categories, will data available at the initial project stage help predict effort hours?
2) Do growth rates change differently in process-oriented categories ?
3) Are estimated effort hours useful to predict actual effort hours?

Effort Estimation Challenges

- Old paradigm estimates based on:
  - Completed project data
  - Source Lines of Code (SLOC)
  - Subject Matter Experts

- Old paradigm estimates impacted by:
  - Inconsistent SLOC conversions
  - Lack of relevant historical data
  - Lack of quality historical data
  - Lack of subcategory trend analysis



Estimating Paradigm Shift

- This paper attempts to describe an alternate software development effort estimating paradigm
  - Based on actual DoD software development efforts
  - Based on completed development efforts

- Initial analysis indicates very strong relationships between actual and estimated values
  - From analyzing the relationships between initial and final software report submissions
  - From comparing initial (e.g., estimated) variable values to final effort values (e.g., actual)

Model Subcategories

- Model Flexibility Supports New Hypotheses for existing and created subsets

S R D R

- **New vs. Upgrade**
- **Primary Language**
- **CMMI Level**
- **Contractor**
- **Development Process**
- **Application Domain**
- **Super Domain**
- **Operating Environment**
- **Contract Type**
- **Program Type** **(NCCA tag?)**

Software Estimating: New Paradigm

Univariate:
- Distributions for Uncertainty
- Factors
- Parametric CERs
- Percent Change Analysis

Multi-variable:
- Parametric CERs
- Factors

Subset CER/Factor **Comparisons**

Increased Estimating Flexibility

Focusing on Deriving Relationships Less Dependent on SLOC Inputs

Experimental Design

Quantitative Method

- Non-random sample of secondary data
- Projects reported at the CSCI level for early program phases, beginning to elaborate system requirements
- To minimize threats to validity, the analysis framework focused on estimated inputs rather than final inputs

Instrumentation

- Questionnaire:
  - Software Resource Data Report (SRDR) (DD-Form 2630)
- Content:
  - Allows for the collection of project context, responsible company or government information, certified maturity level, requirements, product size, effort, and schedule
- Source:
  - Cost Assessment Data Enterprise (CADE) website:

    http://cade.osd.mil/Files/Policy/Initial_Developer_Report.xlxs

    http://cade.osd.mil/Files/Policy/Final_Developer_Report.xlxs

Sample and Population

- Empirical data from 408 recent records
  - 204 paired initial and final records

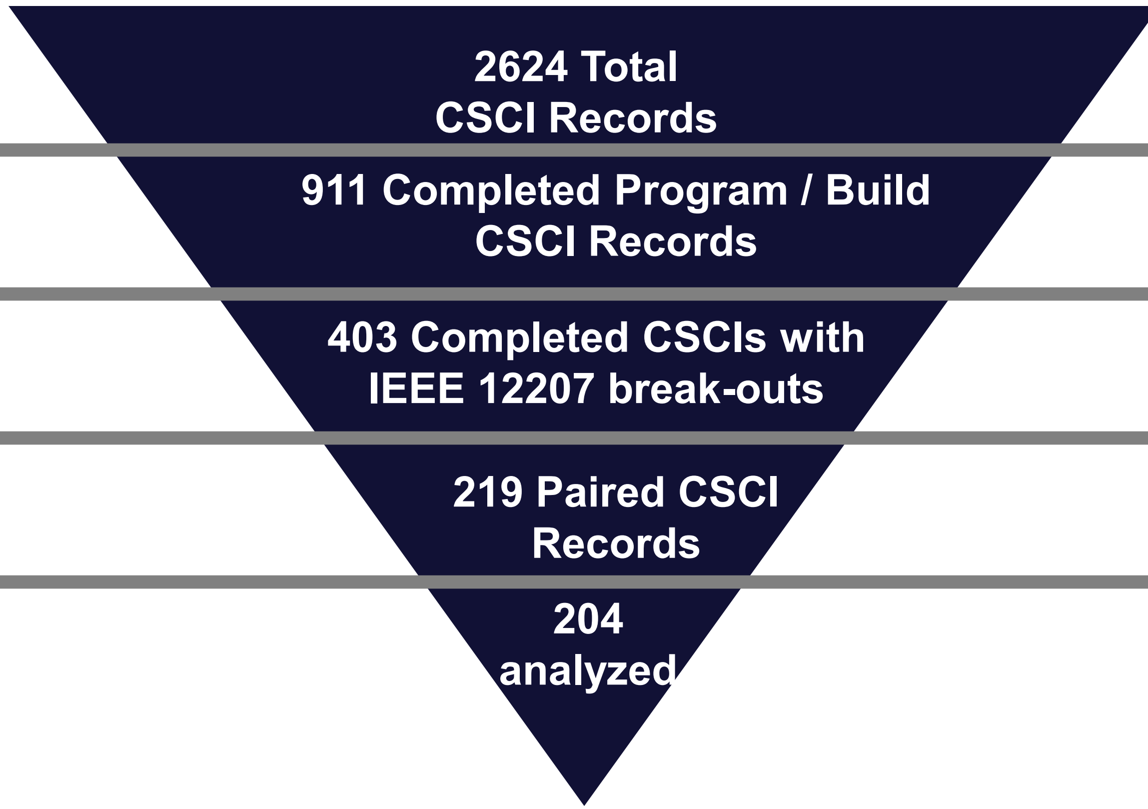
Each program submitted:

SRDR Initial Developer Report (**Estimates**)

&

SRDR Final Developer Report (**Actuals**)

Data Analysis Pedigree

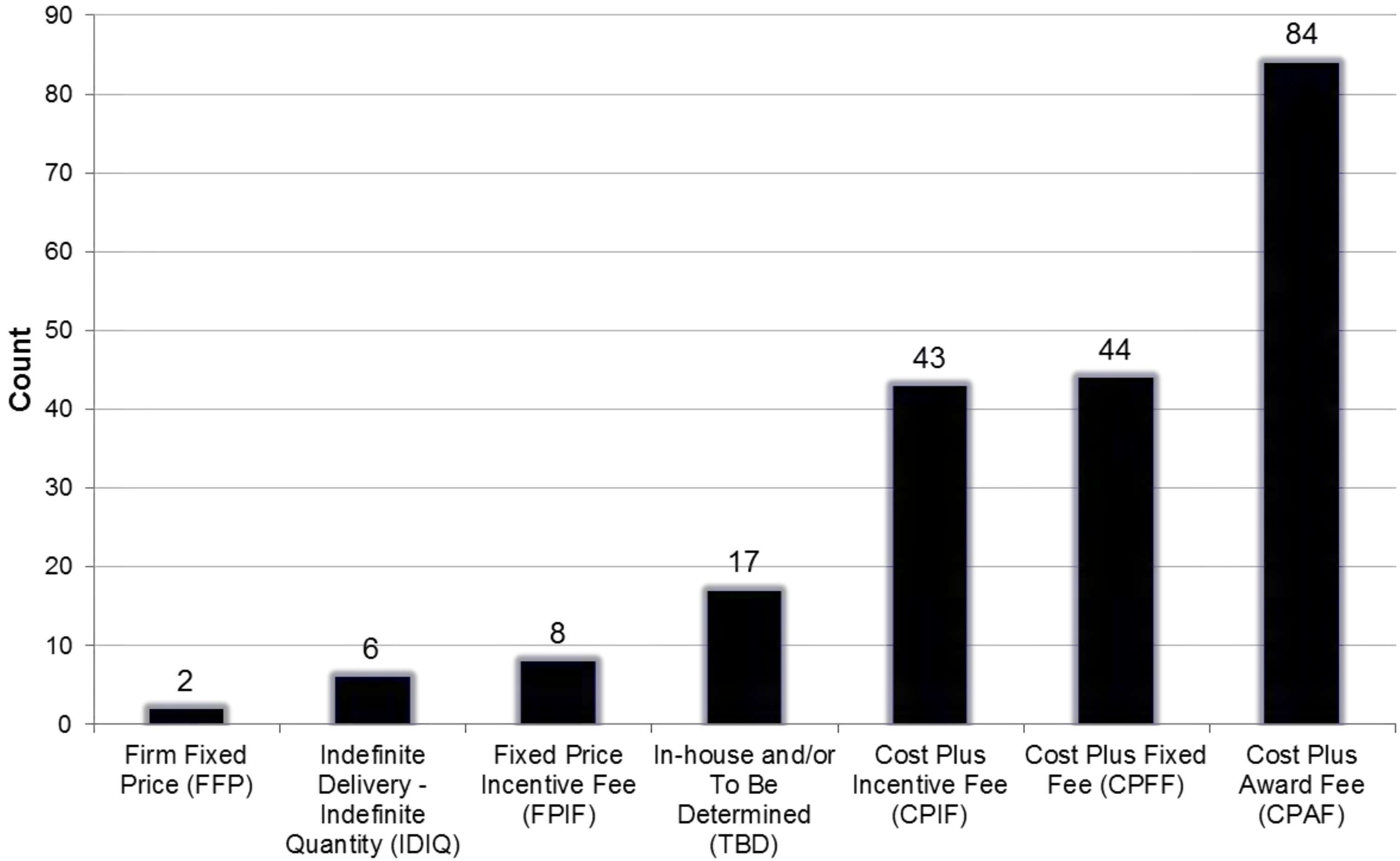


Since last ICEAA (2015)
15 outliers were excluded, accompanied by documented rationales

Contract Type

Contract Types	Symbol	Definition
Cost Plus Award Fee	CPAF	The contract level of effort is uncertain and it is not feasible or effective to negotiate an adjustment formula. The likelihood of meeting objectives can be enhanced by a clear subjective fee plan.
Cost Plus Fixed Fee	CPFF	Cost uncertainty is so great that establishment of predetermined targets and incentive sharing arrangements could result in a final fee out of line with the actual work
Cost Plus Incentive Fee	CPIF	Cost uncertainties are so great that any fixed-price contract would force the contractor to accept an unreasonable risk, but reasonable targets and formulas for sharing costs may be negotiated

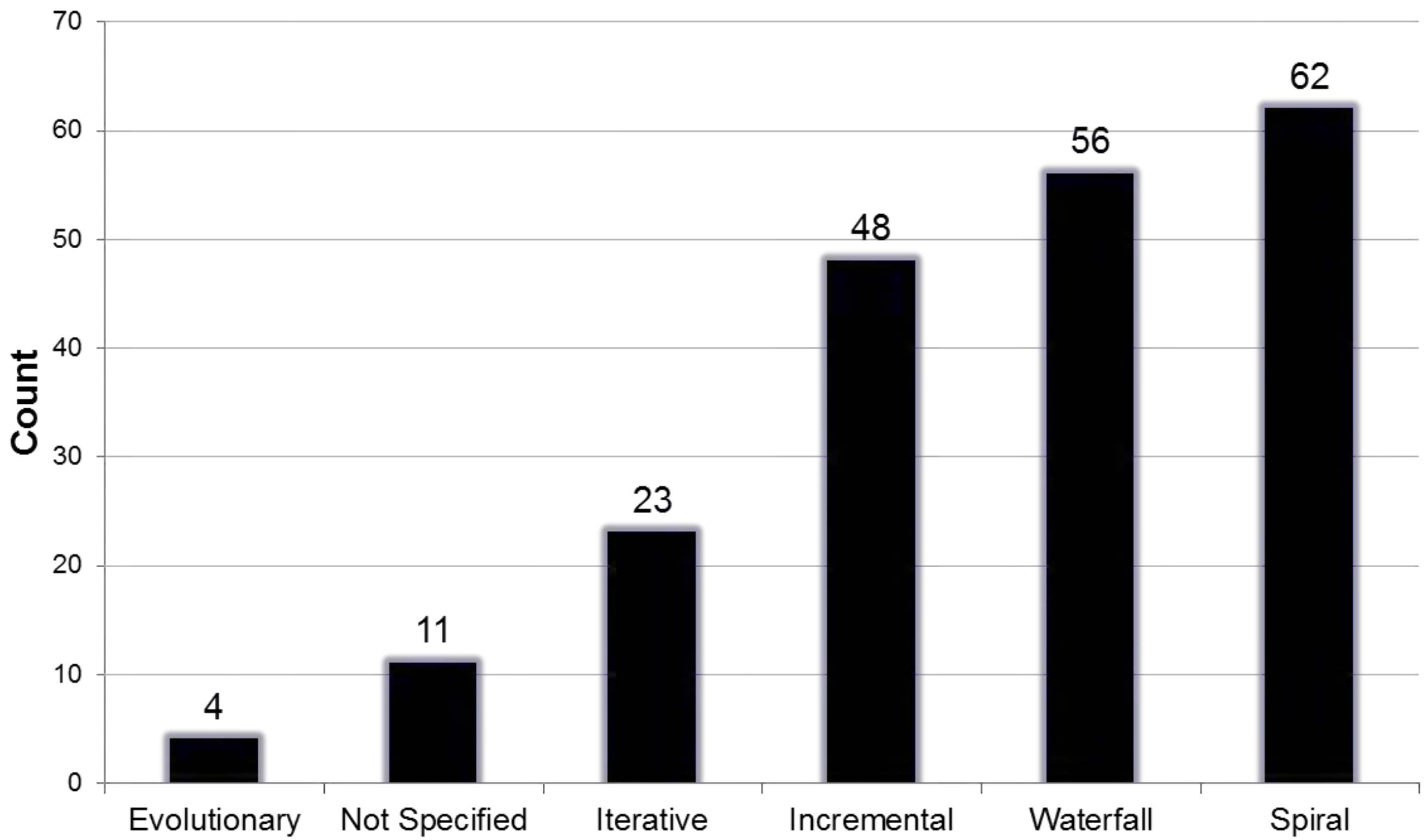
Contract Type



Development Process

Contract Types	Symbol	Definition
Waterfall	Wat	Sequential (non-iterative) design process
Incremental	Incr	Total development effort divided into smaller builds that are designed, coded, and tested in sequence before moving to the next build
Iterative	Iter	Gradual addition of features that are designed, coded, and tested in repeated cycles
Spiral	Spi	Similar to incremental model with increased emphasis on risk analysis
Agile	Agi	Developed in small, rapid cycles with each release building upon the prior build

Development Process



Note: 15 rows excluded

Measures of Model Reliability and Validity

Measure	Symbol	Description
Number of Records	n	The number of records used in the model
Coefficient of Variation	CV	Percentage expression of the standard error compared to the mean of the dependent variable. This is a relative measure allowing direct comparison among models.
P-value	α	Level of statistical significance established through the coefficient alpha ($p \le \alpha$)
Coefficient of Determination	R^2	The Coefficient of Determination shows how much variation in the dependent variable is explained by the regression equation.
t-test	t-stat	Provides a measure of the significance of the predictor variables in the regression model. The variable is significant when the t-stat is greater than the two-tailed value, given the degrees of freedom and coefficient alpha ($\alpha = 0.05$)
Mean Magnitude of Relative Error	MMRE	Mean Magnitude of Relative Error (MRE) measures differences between actual and predicted values relative to the actual value. The mean is computed using every observation.
Prediction Accuracy	PRED(30)	Prediction accuracy is rated as a percentage of the number of records below an individual MRE ≤ 0.30

Data Analysis

Pairwise Correlation Analysis

- Variable selection based on Pairwise Correlation
  - Pairwise correlation chosen over structural equation modeling as the number of observations by subset was below the minimum observations (i.e. 200) needed

From Final Report
Actual Effort
Actual Duration

From Initial Report
Estimated CSCI Requirement Counts
Estimated Peak Staff
Estimated Total SLOC in Logical Statements (LS)
Estimated New Code in LS
Estimated Effort
Estimated Duration

Contract Type Correlation Analysis

- Actual Effort Hours
  - **For all cost plus contracts, Actual Effort Hours are correlated to Estimated Effort Hours**
  - CPAF is correlated to PS, New, and SLOC
  - CPFF is correlated to New
  - CPIF is correlated to PS

- Actual Duration
  - **For all cost plus contracts, Actual Duration is correlated to Estimated Duration (in months)**
  - CPAF is correlated to REQ
  - CPIF is correlated to New and SLOC

CPAF Pearson Correlation	Actual Duration	Estimated Effort	Estimated Requirements	Estimated Duration	Estimated Peak Staff	Estimated New SLOC in LS	Estimated SLOC in LS
Actual Effort	0.10	0.81	0.55	0.22	0.77	0.56	0.47
Actual Duration	1	0.10	-0.16	0.58	-0.09	0.23	0.17
Estimated Effort		1	0.35	0.23	0.71	0.68	0.56
Estimated Requirements			1	-0.14	0.41	0.12	0.06
Estimated Duration				1	0.12	0.24	0.13
Estimated Peak Staff					1	0.52	0.38
Estimated New SLOC in LS						1	0.33
Estimated SLOC in LS							1

CPAF Spearman Correlation	Actual Duration	Estimated Effort	Estimated Requirements	Estimated Duration	Estimated Peak Staff	Estimated New SLOC in LS	Estimated SLOC in LS
Actual Effort	0.16	0.90	0.16	0.24	0.79	0.78	0.72
Actual Duration	1	0.16	0.67	0.71	-0.04	0.22	0.11
Estimated Effort		1	0.10	0.24	0.80	0.85	0.68
Estimated Requirements			1	0.08	0.59	0.51	0.54
Estimated Duration				1	0.15	0.30	0.06
Estimated Peak Staff					1	0.69	0.57
Estimated New SLOC in LS						1	0.68
Estimated SLOC in LS							1

CPFF Pearson Correlation	Actual Duration	Estimated Effort	Estimated Requirements	Estimated Duration	Estimated Peak Staff	Estimated New SLOC in LS	Estimated SLOC in LS
Actual Effort	0.16	0.90	0.60	0.21	0.59	0.76	0.28
Actual Duration	1	0.17	0.13	0.90	-0.06	0.14	0.33
Estimated Effort		1	0.55	0.23	0.54	0.67	0.27
Estimated Requirements			1	0.08	0.62	0.19	0.07
Estimated Duration				1	-0.02	0.20	0.27
Estimated Peak Staff					1	0.34	0.06
Estimated New SLOC in LS						1	0.35
Estimated SLOC in LS							1

CPFF Spearman Correlation	Actual Duration	Estimated Effort	Estimated Requirements	Estimated Duration	Estimated Peak Staff	Estimated New SLOC in LS	Estimated SLOC in LS
Actual Effort	0.22	0.86	0.33	0.28	0.62	0.73	0.70
Actual Duration	1	0.33	0.69	0.88	-0.03	0.08	0.26
Estimated Effort		1	0.21	0.39	0.62	0.75	0.69
Estimated Requirements			1	0.32	0.54	0.54	0.55
Estimated Duration				1	-0.01	0.06	0.21
Estimated Peak Staff					1	0.61	0.55
Estimated New SLOC in LS						1	0.63
Estimated SLOC in LS							1

CPIF Pearson Correlation	Actual Duration	Estimated Effort	Estimated Requirements	Estimated Duration	Estimated Peak Staff	Estimated New SLOC in LS	Estimated SLOC in LS
Actual Effort	0.08	0.88	0.58	-0.06	0.57	0.45	0.51
Actual Duration	1	0.03	0.29	0.88	-0.01	0.12	0.20
Estimated Effort		1	0.41	-0.06	0.54	0.45	0.35
Estimated Requirements			1	0.08	0.56	0.59	0.60
Estimated Duration				1	-0.15	0.02	0.02
Estimated Peak Staff					1	0.60	0.64
Estimated New SLOC in LS						1	0.38
Estimated SLOC in LS							1

CPIF Spearman Correlation	Actual Duration	Estimated Effort	Estimated Requirements	Estimated Duration	Estimated Peak Staff	Estimated New SLOC in LS	Estimated SLOC in LS
Actual Effort	0.14	0.88	0.25	0.04	0.66	0.61	0.55
Actual Duration	1	0.25	0.65	0.85	0.08	0.24	0.22
Estimated Effort		1	0.20	0.13	0.75	0.60	0.59
Estimated Requirements			1	0.03	0.43	0.55	0.46
Estimated Duration				1	-0.14	0.10	-0.02
Estimated Peak Staff					1	0.59	0.80
Estimated New SLOC in LS						1	0.60
Estimated SLOC in LS							1

Development Process Correlation Analysis

- Actual Effort Hours
  - **For all development processes, Actual Effort Hours are correlated to Estimated Effort Hours**
  - Spiral is correlated to New and Mod
  - Waterfall and Iterative are correlated to PS and New
  - Incremental is correlated to PS, New, and REQ
- Actual Duration
  - **For Iterative and Spiral development processes, Actual Durations are correlated to Estimated Durations (in months)**
  - Waterfall and Incremental do not show strong duration correlations for this dataset

Iterative Pearson	Final-Hours	Final-Month	Initial-Hours	Inital-Req	Inital-Peak-Staff	Initial-Month	Init-New-LS	Init-Mod-LS	Init-Reuse-LS	Init-Auto-Gen-LS	Init-SLOC-LS
Final-Hours	1.00	-0.33	0.84	0.42	0.64	-0.35	0.60	0.51	0.15	0.02	0.45
Final-Month		1.00	-0.19	0.03	-0.21	0.92	0.06	-0.09	0.06	0.22	0.10
Initial-Hours			1.00	0.30	0.65	-0.24	0.52	0.70	0.17	-0.05	0.44
Inital-Req				1.00	0.45	0.20	0.39	-0.18	0.26	0.66	0.48
Inital-Peak-Staff					1.00	-0.26	0.64	0.13	0.36	-0.11	0.52
Initial-Month						1.00	-0.03	-0.19	-0.02	0.48	0.06
Init-New-LS							1.00	0.14	0.54	-0.12	0.81
Init-Mod-LS								1.00	-0.01	-0.14	0.17
Init-Reuse-LS									1.00	-0.18	0.88
Init-Auto-Gen-LS										1.00	0.03
Init-SLOC-LS											1.00

Waterfall Pearson	Final-Hours	Final-Month	Initial-Hours	Inital-Req	Inital-Peak-Staff	Initial-Month	Init-New-LS	Init-Mod-LS	Init-Reuse-LS	Init-Auto-Gen-LS	Init-SLOC-LS
Final-Hours	1.00	0.01	0.91	0.25	0.76	0.22	0.66	0.37	0.31	-0.11	0.36
Final-Month		1.00	0.04	0.30	0.16	0.32	0.38	0.32	0.33	-0.10	0.35
Initial-Hours			1.00	0.28	0.82	0.22	0.60	0.38	0.30	-0.15	0.34
Inital-Req				1.00	0.27	0.52	0.27	0.57	0.05	-0.08	0.09
Inital-Peak-Staff					1.00	0.24	0.56	0.54	0.44	-0.11	0.48
Initial-Month						1.00	0.32	0.43	0.05	-0.04	0.08
Init-New-LS							1.00	0.32	0.27	-0.15	0.34
Init-Mod-LS								1.00	0.57	-0.10	0.60
Init-Reuse-LS									1.00	-0.05	1.00
Init-Auto-Gen-LS										1.00	-0.06
Init-SLOC-LS											1.00

Incremental Pearson	Final-Hours	Final-Month	Initial-Hours	Inital-Req	Inital-Peak-Staff	Initial-Month	Init-New-LS	Init-Mod-LS	Init-Reuse-LS	Init-Auto-Gen-LS	Init-SLOC-LS
Final-Hours	1.00	0.11	0.88	0.68	0.67	0.15	0.65	0.42	0.14	0.48	0.36
Final-Month		1.00	0.16	0.01	-0.27	0.49	0.16	0.23	-0.02	0.04	0.03
Initial-Hours			1.00	0.57	0.59	0.18	0.83	0.45	0.12	0.36	0.34
Inital-Req				1.00	0.46	0.04	0.31	0.20	0.04	0.21	0.15
Inital-Peak-Staff					1.00	-0.13	0.35	0.13	0.19	0.46	0.35
Initial-Month						1.00	0.16	0.18	-0.18	-0.09	-0.14
Init-New-LS							1.00	0.36	0.07	0.16	0.25
Init-Mod-LS								1.00	0.02	0.24	0.21
Init-Reuse-LS									1.00	0.40	0.94
Init-Auto-Gen-LS										1.00	0.66
Init-SLOC-LS											1.00

SPIRAL Pearson	Final-Hours	Final-Month	Initial-Hours	Inital-Req	Inital-Peak-Staff	Initial-Month	Init-New-LS	Init-Mod-LS	Init-Reuse-LS	Init-Auto-Gen-LS	Init-SLOC-LS
Final-Hours	1.00	-0.06	0.81	0.46	0.49	-0.10	0.51	0.59	0.14	0.08	0.54
Final-Month		1.00	-0.14	-0.05	0.06	0.82	0.01	-0.13	0.07	0.10	0.00
Initial-Hours			1.00	0.23	0.48	-0.12	0.51	0.44	0.28	0.06	0.58
Inital-Req				1.00	0.27	-0.13	0.00	0.62	0.01	-0.05	0.26
Inital-Peak-Staff					1.00	0.01	0.45	0.30	0.04	0.05	0.34
Initial-Month						1.00	0.00	-0.20	0.28	0.10	0.12
Init-New-LS							1.00	0.16	0.16	-0.08	0.59
Init-Mod-LS								1.00	0.06	-0.06	0.50
Init-Reuse-LS									1.00	0.23	0.79
Init-Auto-Gen-LS										1.00	0.11
Init-SLOC-LS											1.00

New Paradigm

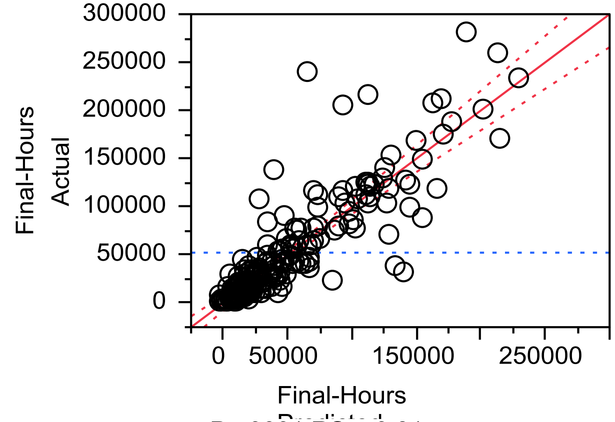
Hours and Requirement Count Estimating Concept

Effort Model Variables

Variable	Type	Definition
Actual Effort	Dependent	Actual software engineering effort (in hours)
Estimated New, Mod, Reuse, and Auto SLOC	Independent	Total Source Lines of Code (SLOC) categorized as New, Modified, Reuse, or Auto
Estimated Total Requirements	Independent	Total Requirements captured in the Software Requirements Specification (SRS). These are the estimated total requirements at contract award.
Estimated Peak Staff	Independent	Actual peak team size, measured in full-time equivalent staff. Only include direct labor.
Estimated Effort	Independent	Estimated software engineering effort (in hours)
Estimated Duration	Independent	Estimated software engineering duration (in months)

Entire Dataset: Predicting Actual Hours with All Estimated Variables

- Multivariate analysis of all "Initial" variables indicates summary level fit
- Several independent variables were not significant (**Prob > |t| above .05**)
  - Initial New
  - Initial Mod
  - Initial Reuse
- Several independent variables were significant (**Prob > |t| below .05**)
  - Initial Hours
  - Initial Requirement counts
  - Initial Peak Staff counts
  - Initial Duration (in months)



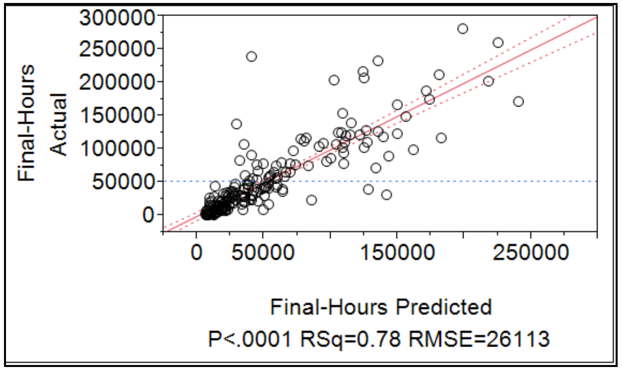
P<.0001 RSq=0.81

RSquare	0.815
RSquare Adj	0.807
Root Mean Square Error	24,572
Mean of Response	52,380
Observations (or Sum Wgts)	200

Term	Estimate	Std Error	t Ratio	Prob>|t|	VIF
Intercept	-3,233.87	3,954.495	-0.82	0.4145	.
Initial-New	0.0715735	0.039668	1.80	0.0728	1.7278008
Initial-Mod	0.0516804	0.050073	1.03	0.3033	1.7583416
Initial-Reuse	0.005661	0.006708	0.84	0.3997	1.1211566
Initial-Auto	0.1391352	0.03714	3.75	0.0002*	1.0483384
Initial-Hours	0.8101929	0.0569	14.24	<.0001*	2.3353118
Inital-Req	7.134068	2.124851	3.36	0.0009*	1.7140274
Initial-Peak-Staff	318.95122	146.3109	2.18	0.0305*	2.0139019
Initial-Month	264.23516	107.6958	2.45	0.0150*	1.3089695

Entire Dataset: Predicting Actual Hours with Selected Estimated Variables

- Analysis indicates good fit with the statistically significant independent variables below:
  - Initial Hours
  - Initial Requirements Count
- Cost analysts may generate software estimates without explicit SLOC/ESLOC counts or productivity-based metrics
- Based on April 2014 paired initial and final SRDR data points, excluding four records with multiple missing initial values



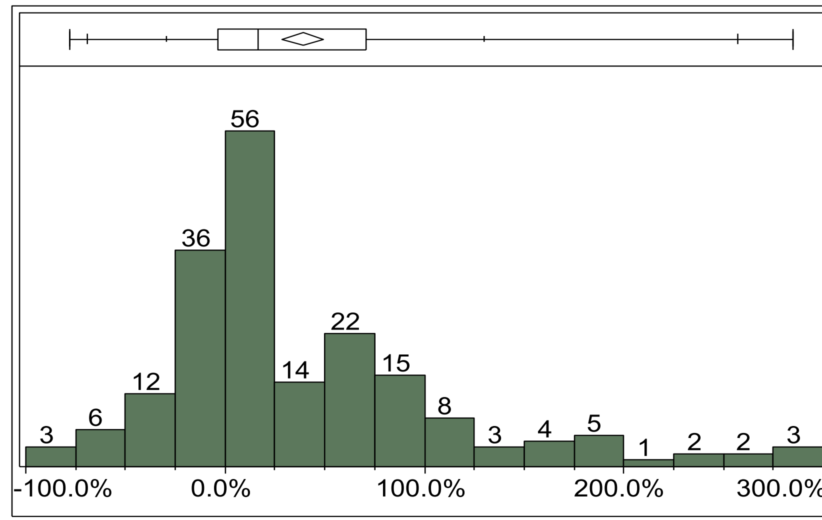
RSquare	0.784236
RSquare Adj	0.782045
Root Mean Square Error	26113.47
Mean of Response	52380.44
Observations (or Sum Wgts)	200

Term	Estimate	Std Error	t Ratio	Prob>|t|	VIF
Intercept	5870.9039	2550.668	2.30	0.0224*	.
Initial Hours	0.9915144	0.040649	24.39	<.0001*	1.055324
Initial Req. Count	9.0827304	1.771865	5.13	<.0001*	1.055324

New Paradigm
Univariate Statistics
Percent Change by Contract Type

Entire Dataset: Percent Change (PC) Analysis

- Compares percent growth in effort hours from initial to final reporting events



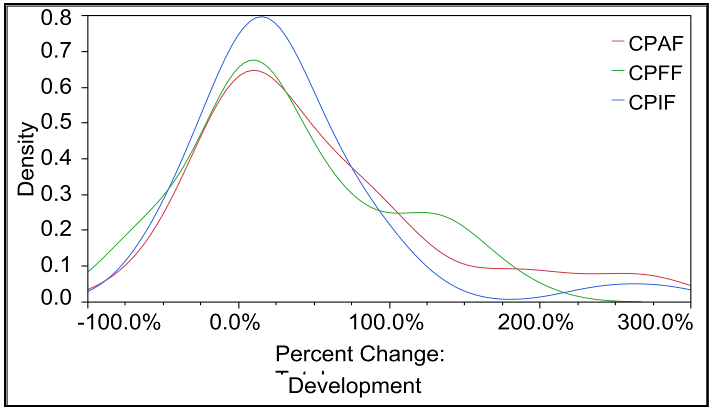
Mean	0.3890538
Std Dev	0.7172871
Std Err Mean	0.0517657
Upper 95% Mean	0.4911597
Lower 95% Mean	0.2869479
N	192
Sum Wgt	192
Sum	74.69833
Variance	0.5145007
Skewness	1.3838684
Kurtosis	2.1103624
CV	184.36706
N Missing	0

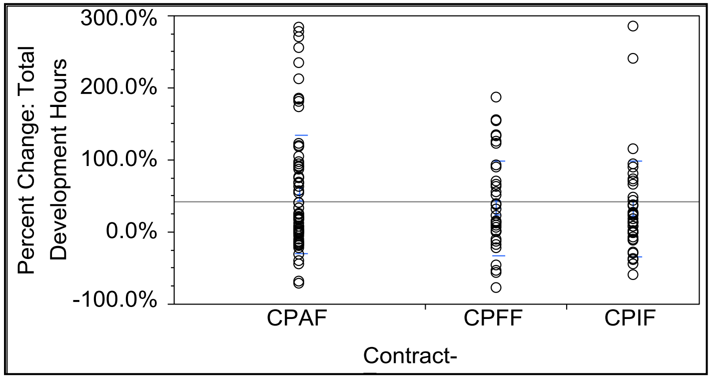
100.0%	maximum	2.84982
99.5%		2.84982
97.5%		2.57157
90.0%		1.30207
75.0%	quartile	0.70528
50.0%	median	0.16843
25.0%	quartile	-0.039
10.0%		-0.2956
2.5%		-0.6914
0.5%		-0.7786
0.0%	minimum	-0.7786

Filter: All data, less than +300%

Contract Type: Percent Change (PC) Analysis

- Contract types result in similar distribution shapes
- Higher variance for CPAF
- When contract type is known, PC effort hour uncertainty may be explicitly modeled, based on empirical data





Level	Number	Mean	Std Dev	Std Err
CPAF	76	0.522318	0.814732	0.09346
CPFF	43	0.332074	0.659618	0.10059
CPIF	40	0.324001	0.665465	0.10522

Filter: Contract Type, less than +300%

Effort Models

By Contract Type

Subset	Records	Equation	R2 in Fit Space	R2 in Unit Space	SE	RMS of % Errors	MAD	CV (MAD Res/ Avg Act)	MMRE	PRED(30)	Minimum Value	Maximum Value
											eEH	
CPAF	74	aEH = 8.53 * eEH ^ 0.82	0.79	0.64	28741.6	0.78	0.51	0.35	0.51	0.46	575	169583
CPFF	40	aEH = 1.27 * eEH ^ 0.99	0.71	0.78	15281.0	1.11	0.58	0.31	0.58	0.50	1896	101665
CPIF	43	aEH =18.8 * eEH ^ 0.75	0.66	0.70	35749.2	0.84	0.49	0.28	0.49	0.60	2235	191013

Conclusion

- New paradigm subset analysis increases estimating flexibility and accuracy
- Allows analysts to develop estimates without reliance on SLOC, ESLOC, or productivity-rate metrics
- Based on actual historical DoD software development efforts

Relationships Can be Derived Using Any Available Subset or Category

Future Work

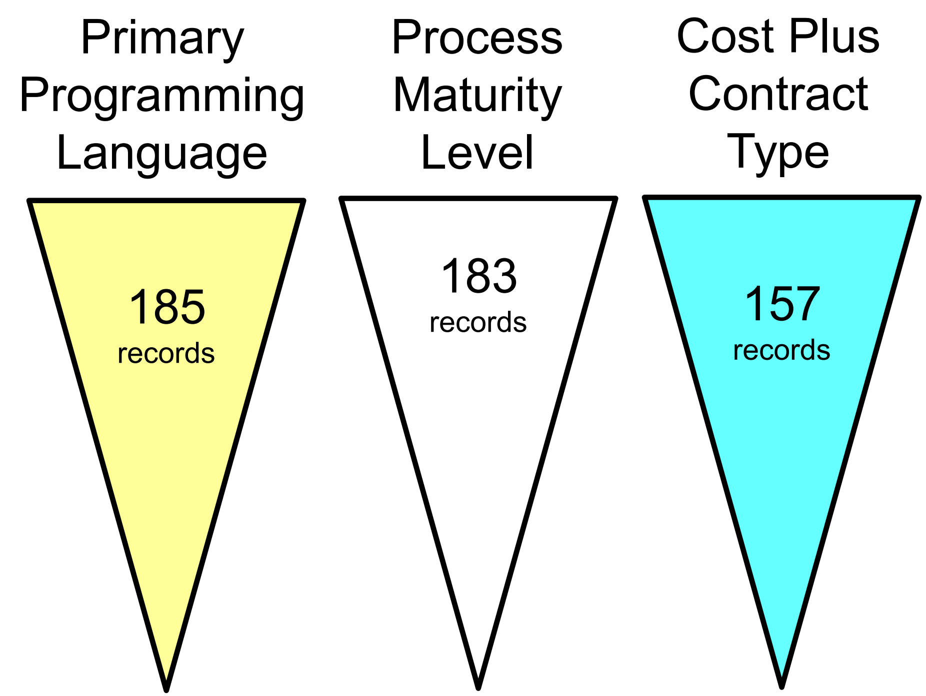
- Contacts:
  - Nicholas Lanham, NCCA
  - Dr. Corinne Wallshein, NCCA
  - Dr. Wilson Rosa, NCCA

- Future Work:
  - Analyzing software development phasing
  - Improving SRDR data quality via SURF
  - Updating SRDR dataset from April 2014 on

Backup

Model Acceptance Criteria

Measure	Criterion
MAD	≤ 45%
CV	≤ 45%
R^2	≥ 55%
t-test	> Two tailed critical value ($\alpha = 0.05$)



- Zero / blank input values were excluded to generate comparative CERs and SERs

Development Process

Spearman Correlation Matrixes

Waterfall Spearman	Final-Hours	Final-Month	Initial-Hours	Inital-Req	Inital-Peak-Staff	Initial-Month	Init-New-LS	Init-Mod-LS	Init-Reuse-LS	Init-Auto-Gen-LS	Init-SLOC-LS
Final-Hours	1.00	0.04	0.95	0.56	0.87	0.04	0.85	0.32	0.18	-0.06	0.65
Final-Month		1.00	0.03	0.26	0.16	0.83	0.26	0.30	0.39	-0.07	0.34
Initial-Hours			1.00	0.52	0.87	0.06	0.86	0.32	0.16	-0.18	0.62
Inital-Req				1.00	0.57	0.14	0.55	0.36	0.34	-0.03	0.65
Inital-Peak-Staff					1.00	0.17	0.78	0.45	0.36	-0.12	0.71
Initial-Month						1.00	0.28	0.33	0.32	-0.07	0.25
Init-New-LS							1.00	0.28	0.18	-0.30	0.66
Init-Mod-LS								1.00	0.53	-0.23	0.49
Init-Reuse-LS									1.00	0.11	0.77
Init-Auto-Gen-LS										1.00	0.05
Init-SLOC-LS											1.00

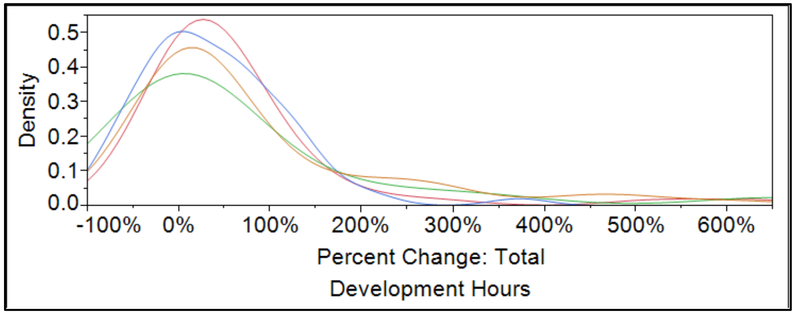
Spiral Spearman	Final-Hours	Final-Month	Initial-Hours	Inital-Req	Inital-Peak-Staff	Initial-Month	Init-New-LS	Init-Mod-LS	Init-Reuse-LS	Init-Auto-Gen-LS	Init-SLOC-LS
Final-Hours	1.00	0.11	0.85	0.66	0.68	0.11	0.64	0.41	0.34	0.13	0.71
Final-Month		1.00	0.11	0.22	0.04	0.88	-0.16	-0.19	0.25	0.09	0.05
Initial-Hours			1.00	0.59	0.67	0.16	0.69	0.32	0.34	0.13	0.69
Inital-Req				1.00	0.63	0.10	0.44	0.32	0.34	-0.09	0.55
Inital-Peak-Staff					1.00	0.02	0.56	0.31	0.28	0.12	0.61
Initial-Month						1.00	-0.14	-0.23	0.20	0.15	0.01
Init-New-LS							1.00	0.23	-0.11	-0.15	0.47
Init-Mod-LS								1.00	0.39	-0.14	0.62
Init-Reuse-LS									1.00	0.21	0.67
Init-Auto-Gen-LS										1.00	0.16
Init-SLOC-LS											1.00

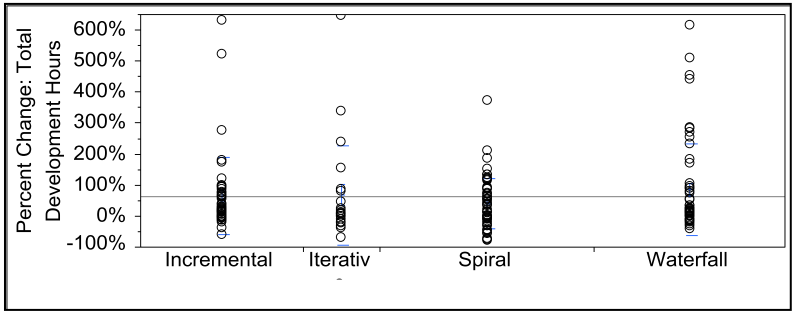
Incremental Spearman	Final-Hours	Final-Month	Initial-Hours	Inital-Req	Inital-Peak-Staff	Initial-Month	Init-New-LS	Init-Mod-LS	Init-Reuse-LS	Init-Auto-Gen-LS	Init-SLOC-LS
Final-Hours	1.00	0.06	0.90	0.62	0.70	0.09	0.71	0.36	0.21	0.01	0.45
Final-Month		1.00	0.15	-0.09	-0.30	0.44	0.30	-0.30	-0.18	0.03	0.01
Initial-Hours			1.00	0.59	0.69	0.19	0.76	0.32	0.16	0.07	0.45
Inital-Req				1.00	0.51	-0.02	0.41	0.34	0.20	-0.01	0.37
Inital-Peak-Staff					1.00	-0.10	0.45	0.29	0.24	0.14	0.38
Initial-Month						1.00	0.29	-0.16	-0.29	-0.27	-0.10
Init-New-LS							1.00	0.23	0.16	0.08	0.49
Init-Mod-LS								1.00	0.16	0.03	0.35
Init-Reuse-LS									1.00	0.27	0.79
Init-Auto-Gen-LS										1.00	0.21
Init-SLOC-LS											1.00

Iterative Spearman	Final-Hours	Final-Month	Initial-Hours	Inital-Req	Inital-Peak-Staff	Initial-Month	Init-New-LS	Init-Mod-LS	Init-Reuse-LS	Init-Auto-Gen-LS	Init-SLOC-LS
Final-Hours	1.00	-0.31	0.64	0.45	0.39	-0.49	0.46	0.31	-0.11	0.10	0.23
Final-Month		1.00	0.17	0.00	0.11	0.83	0.14	0.10	0.20	0.23	0.20
Initial-Hours			1.00	0.39	0.83	-0.16	0.81	0.51	0.17	0.06	0.54
Inital-Req				1.00	0.45	0.03	0.34	-0.12	0.30	0.35	0.55
Inital-Peak-Staff					1.00	-0.17	0.73	0.54	0.34	-0.10	0.58
Initial-Month						1.00	-0.10	-0.07	0.16	0.36	0.13
Init-New-LS							1.00	0.43	0.45	-0.13	0.62
Init-Mod-LS								1.00	0.10	-0.31	0.21
Init-Reuse-LS									1.00	-0.23	0.83
Init-Auto-Gen-LS										1.00	0.13
Init-SLOC-LS											1.00

Development Process: Percent Change (PC) Analysis

- Development processes result in similar distribution shapes
- Lowest standard deviation for Spiral development process
- When development process is known, PC effort hour uncertainty may be explicitly modeled, based on empirical data





Level	Number	Mean	Std Dev	Std Err
Incremental	47	0.653326	1.24921	0.18222
Iterative	22	0.674825	1.60697	0.34261
Spiral	62	0.390656	0.80368	0.10207
Waterfall	55	0.853038	1.46428	0.19744

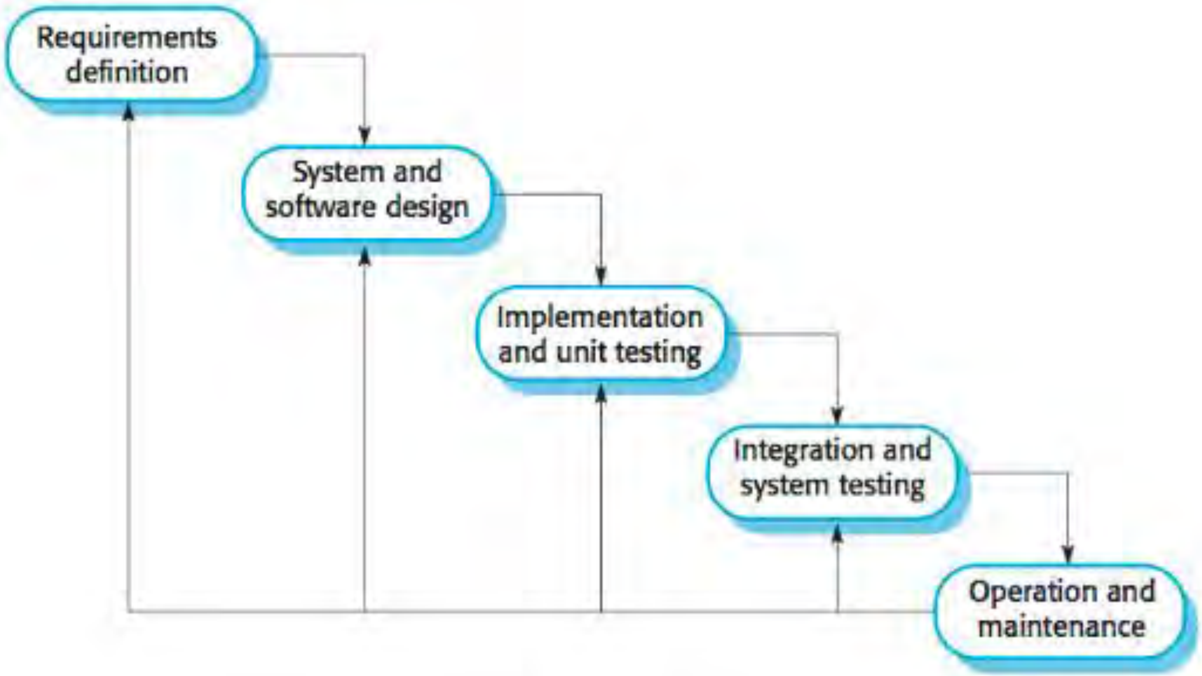
Filter: Development Process, less than +700%

Effort Models

Subset	Records	Equation	R2 in Fit Space	R2 in Unit Space	SE	RMS of % Errors	MAD	CV (MAD Res/Avg Act)	MMRE	PRED(30)	Min Value	Max Value	Min Value	Max Value
											eEH		eREQ	
Waterfall	56	aHr = 8.66 * eHr ^ 0.82	0.87	0.85	21717.62	0.74	0.49	0.24	0.49	0.50	184	99520		
Waterfall	56	aHr = 10957.7 + 0.98 * eHr	0.83	0.83	22983.37	7.33	2.13	0.27	2.13	0.43	184	99520		
Waterfall	55	aHr = 7.3 * eHr ^ 0.8 * eREQ ^ 0.1	0.88	0.83	23180.96	0.67	0.46	0.24	0.46	0.49	184	99520	2	407
Waterfall	55	aHr = 11120.1 + 0.98 * eHr + 0.08 * eREQ	0.83	0.83	23406.36	7.51	2.18	0.27	2.18	0.44	184	99520	2	407
											eEH		eREQ	
Spiral	56	aHr = 3.16 * eHr ^ 0.9	0.70	0.62	32202.62	1.14	0.66	0.42	0.66	0.38	1496	191013		
Spiral	56	aHr = 7592 + 1.02 * eHr	0.65	0.65	31455.92	1.90	0.99	0.43	1.22	0.36	1496	191013		
Spiral	56	aHr = 4.8 * eHr ^ 0.77 * eREQ ^ 0.17	0.74	0.69	29286.93	1.04	0.59	0.37	0.59	0.50	1496	191013	8	8850
Spiral	56	aHr = 4207 + 0.94 * eHr + 8.77 * eREQ	0.73	0.73	27873.17	1.41	0.78	0.37	0.78	0.39	1496	191013	8	8850
											eEH		eREQ	
Iterative	23	aHr = 77 * eHr ^ 0.6	0.47	0.61	21850.02	0.86	0.62	0.41	0.62	0.30	2235	156650		
Iterative	23	aHr = 14042.6 + 0.75 * eHr	0.70	0.70	19551.64	1.07	0.76	0.39	0.76	0.35	2235	156650		
Iterative	23	aHr = 52.8 * eHr ^ 0.5 * eREQ ^ 0.2	0.52	0.62	21526.76	0.72	0.55	0.37	0.55	0.30	2235	156650	55	2400
Iterative	23	aHr = 10415.75 + 0.7 * eHr + 10.4 * eREQ	0.73	0.73	18949.39	0.92	0.67	0.35	0.67	0.39	2235	156650	55	2400
											eEH		eREQ	
Incremental	44	aHr = 2 * eHr ^ 0.96	0.83	0.76	28330.60	0.56	0.37	0.27	0.37	0.59	1104	141000		
Incremental	44	aHr = 2544 + 1.4 * eHr	0.78	0.78	27260.82	0.77	0.52	0.31	0.52	0.45	1104	141000		
Incremental	44	aHr = 2.55 * eHr ^ 0.88 * eREQ ^ 0.11	0.84	0.79	26091.05	0.51	0.35	0.25	0.35	0.59	1104	141000	7	2521
Incremental	44	aHr = (-1521) + 1.189 * eHr + 26.5 * eREQ	0.82	0.82	24631.07	0.86	0.50	0.27	0.50	0.57	1104	141000	7	2521

Development Process

Waterfall

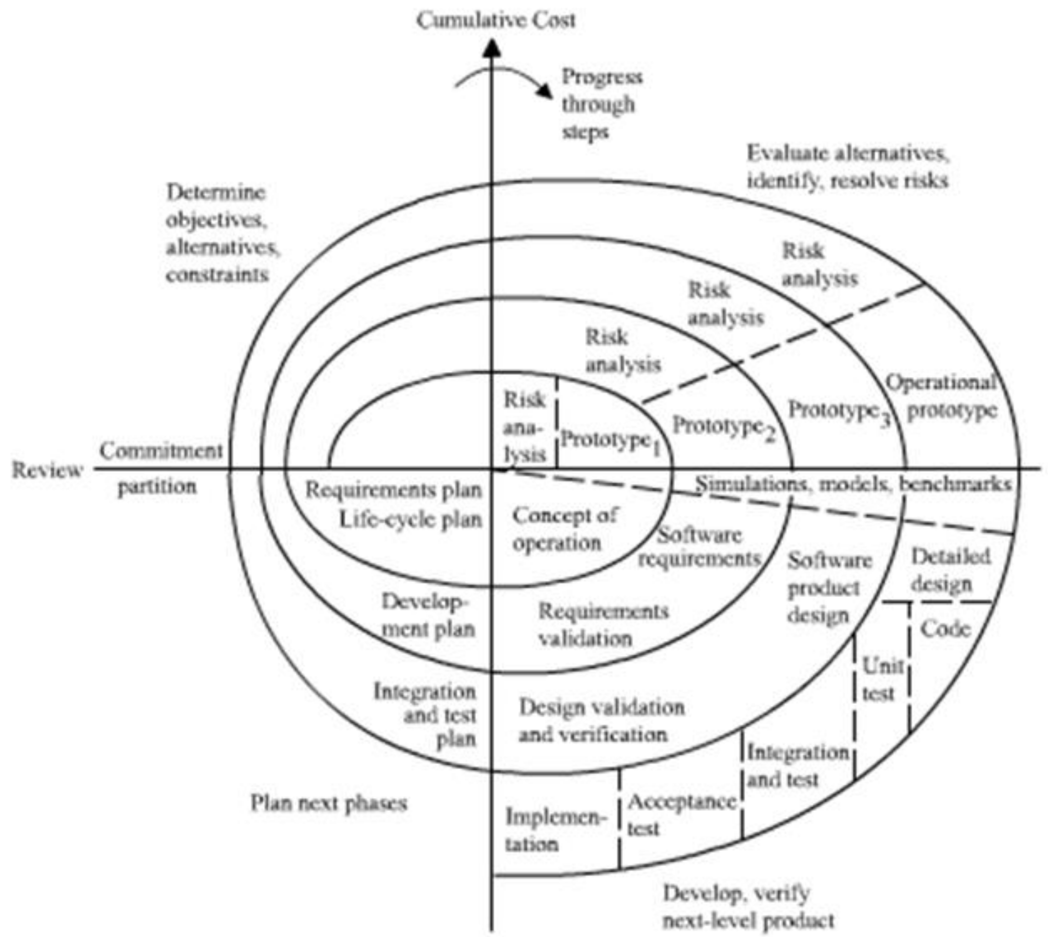


Kurkovsky, S. Central Connecticut State University (CCSU), Department of Computer Science

Ref: http://www.cs.ccsu.edu/~stan/classes/CS530/Notes14/02-SoftwareProcesses.html

Development Process

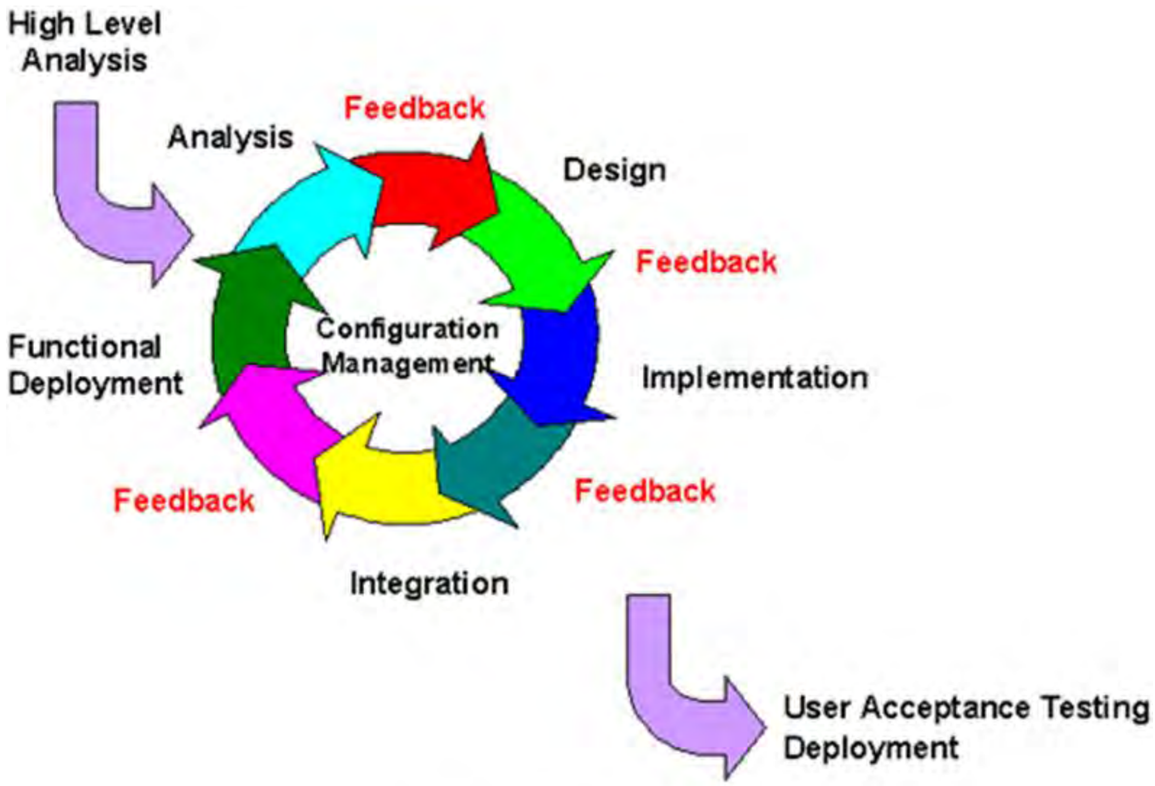
Spiral



Boehm, B. (1988), "A Spiral Model of Software Development and Enhancement"
Ref: https://courses.cs.vt.edu/csonline/SE/Lessons/Spiral/Lesson.html

Development Process

Iterative



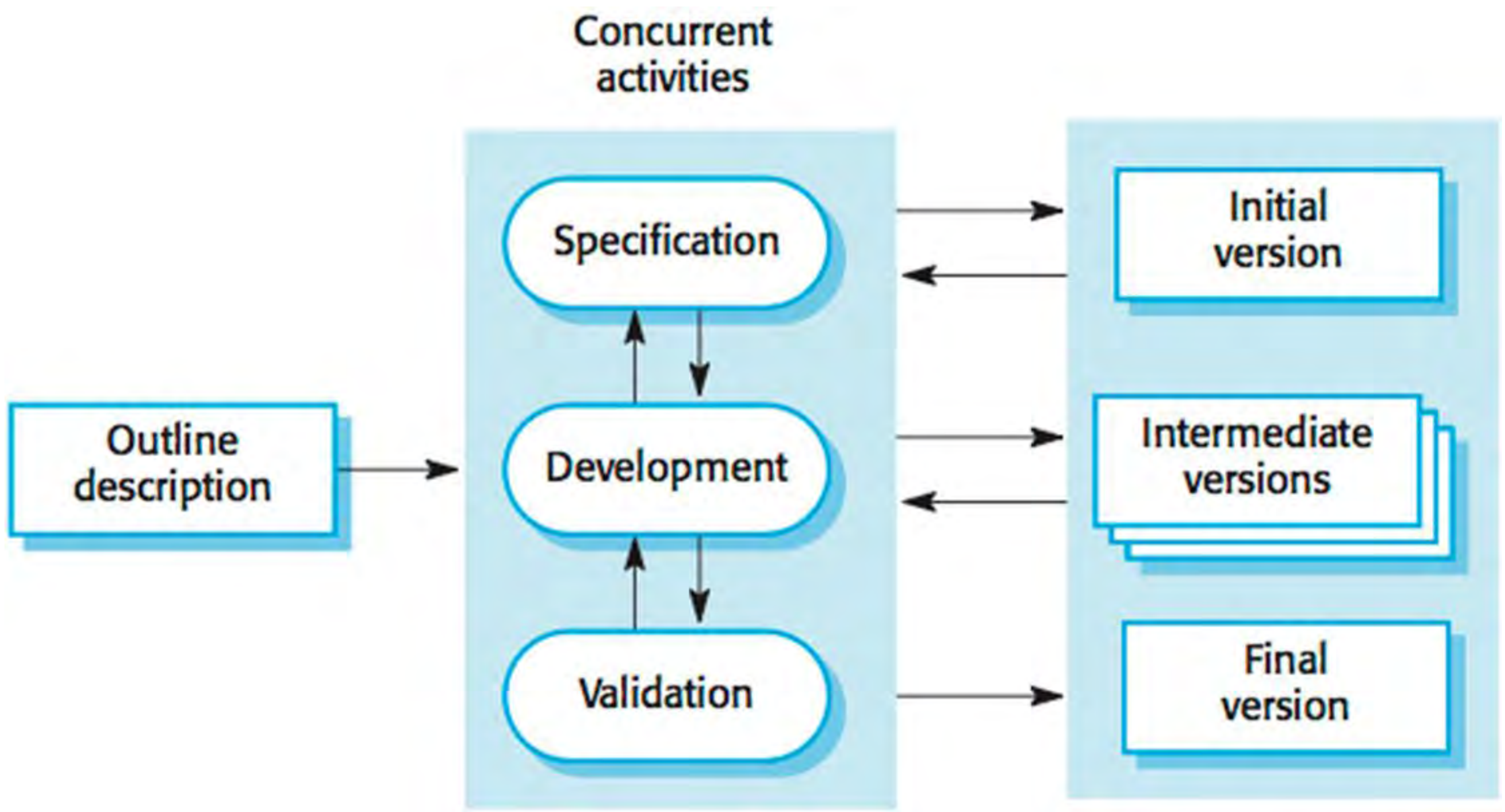
Project Iteration Flow

NIH, National Cancer Institute, Wiki page

Ref: https://wiki.nci.nih.gov/display/CommonProjects/Iterative+Software+Development+Approach

Development Process

Incremental



Kurkovsky, S. Central Connecticut State University (CCSU), Department of Computer Science

Ref: http://www.cs.ccsu.edu/~stan/classes/CS530/Notes14/02-SoftwareProcesses.html

PRICE®

TruePlanning® Risk & Uncertainty Analysis:

Best Practices

August 2016

Agenda

- Overview of PRICE® Approach
- Risk and Uncertainty
- Parameters used in Cost Uncertainty Analysis
  - Size
  - Functional Complexity
  - Reuse
  - Technology
- Uncertainty Options with TruePlanning

Overview of PRICE® Approach

One View of Risk

- "… All projects should be budgeted at a 70% confidence level"

2008 NASA Cost Estimating Handbook

Section 2. Cost Risk Approaches

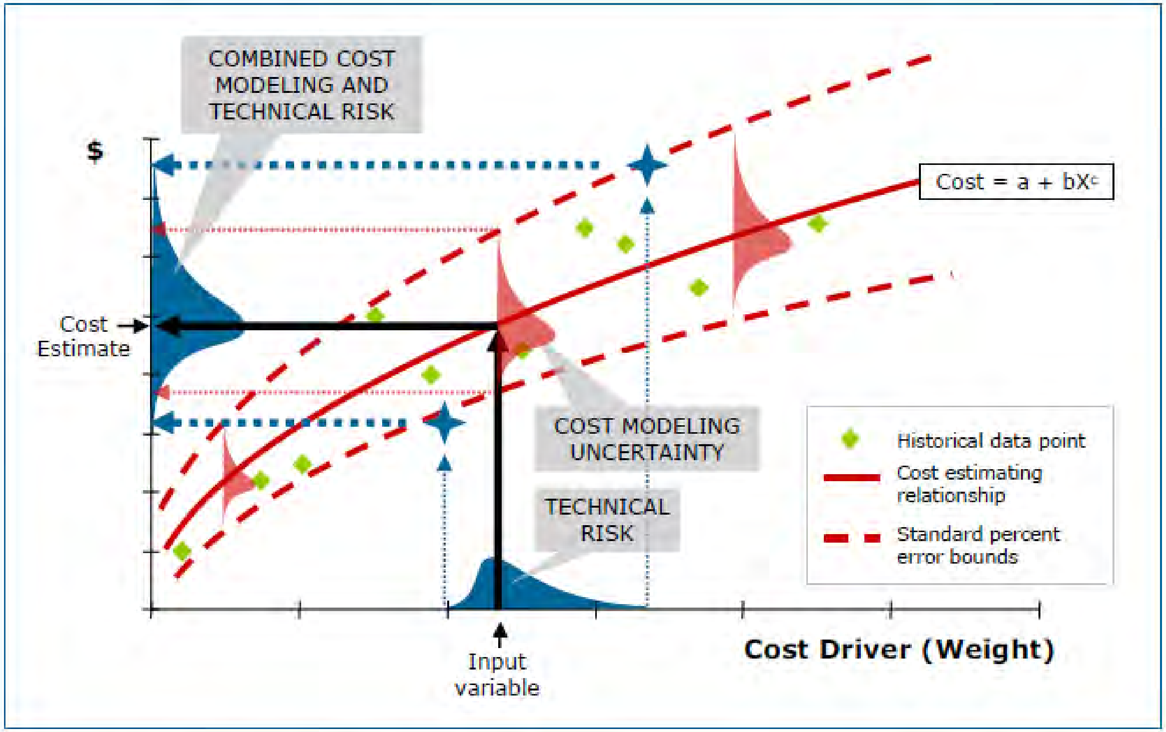


Figure 2-11. Culmination of CER and Technical Risk

Source: 2008 NASA Cost Estimating Handbook (CHE), http://www.ceh.nasa.gov

Risk and Uncertainty

Risk versus Uncertainty

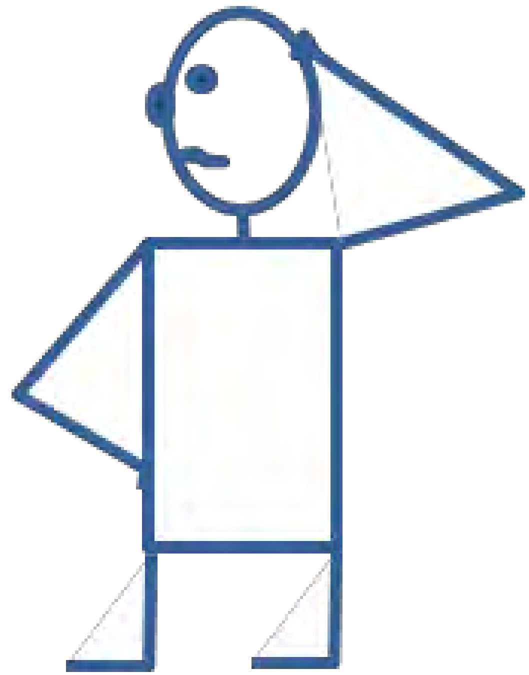
- Risk
  - Known Unknowns
  - Implies that Probabilities can be Assigned to an Event
  - Risk exists when the decision maker is in a position to assign probabilities to various outcomes. This occurs when there is historical data on the basis in which you can assign probability to other projects of the same nature.
- Uncertainty
  - Unknown Unknowns
  - Implies that Probabilities Cannot be Assigned to an Event
  - Uncertainty exists when the decision maker has no historical data from which to develop a probability distribution and must make intelligent guesses in order to develop a subjective probability distribution.

 | *Decades of Cost Management Excellence*

Risk versus Uncertainty

"Risk is measurable uncertainty"

"Uncertainty is unmeasurable risk"



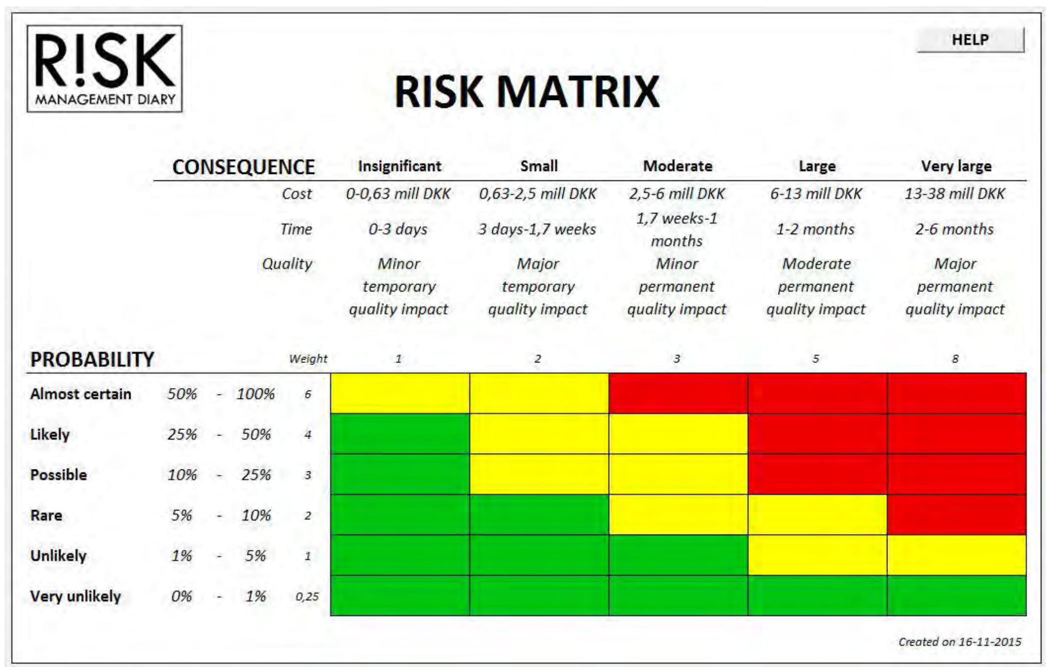
Source:http://beyonddisruptions.blogspot.com/2014/07/risk-vs-uncertainty-and-how-to-make.html

Risk Components

Performance Risk	• The degree of uncertainty that the product will meet its requirements and be fit for its intended use.
Cost Risk	• The degree of uncertainty that the project budget will be maintained.
Support Risk	• The degree of uncertainty that the resultant product will be easy to maintain, correct, and enhance.
Schedule Risk	• The Degree of uncertainty that the project will be maintained and that the product will be delivered on time.

Risk Analysis Projection

- Risk Projection, also called Risk estimation, attempts to rate each risk in two ways
  - The likelihood or probability that the risk is real
  - The consequences of the problems associated with the risk, should it occur



 | *Decades of Cost Management Excellence*

General Types of Uncertainty

Requirements

- Imperfect Knowledge about State of the World
- System Requirements Change over the Course of its Development, Which are Beyond the Control of the Program Manager

Cost Estimating Uncertainty

- Statistical Uncertainty
- Data can be very Subjective
- Sampling Techniques have Random Error

Technological Uncertainty

- Is the Technology Mature Enough to Allow Development and Integration Into the System?

Techniques for Uncertainty Analysis

Sensitivity Analysis

Requirements Uncertainty

Approximation

Cost Estimating Uncertainty

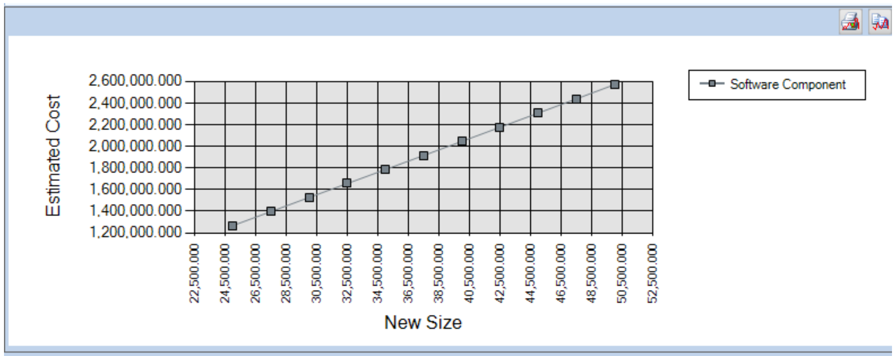
Simulation

Technical Uncertainty

 | ***Decades of Cost Management Excellence***

Sensitivity Analysis

- Compute Baseline System Point Estimate
- Select Equipment Parameters that will have the greatest impact on Total Project Cost
- Determine Parameter Ranges



- Re-Estimate System Costs
  - Change one Parameter Value at a Time
  - Compare new System Estimate to Baseline

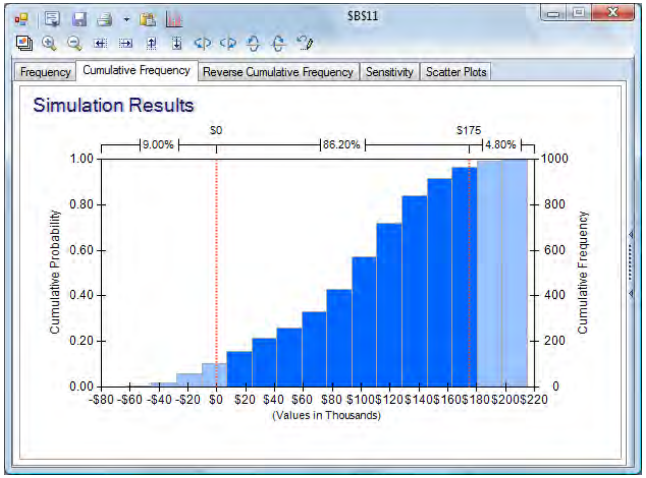
 | ***Decades of Cost Management Excellence***

Approximation

- Normally Accomplished at System Level
  - Determine or Identify Items containing Uncertainty
  - Rate the Uncertainty as Nominal, Low, or High
  - Based on Rating, Apply Uncertainty Percentage to System Level Cost

 | ***Decades of Cost Management Excellence***

Simulation



- Best Methodology for Developing Probabilistic Estimate
- More Time and Effort Consuming
- Requires Identification of Uncertain Elements
  - Input Parameter Ranges Entered for Each Uncertainty Element
    - *Triangular, Normal, Uniform, Beta*
- Monte Carlo Simulation Should then be Used for Calculation of Probabilistic Estimate

Parameters used in Uncertainty Analysis

Parameters used in Uncertainty Analysis (SW)

- Software Size (SLOC, Function Points, COSMIC)
  - For Uncertainty Analysis, uncertainty in software size should be considered a first order driver
  - Uncertainty values can include:
    - *Contingency*
    - *Expected code growth based on similar past projects*
    - *Factors by subsystem (based on contingencies or default percentage) from Optimistic (e.g., Current Best Estimate mass), to Most Likely, to Pessimistic*
  - Code Growth set specifically by Component based on knowledge of technology readiness

 | ***Decades of Cost Management Excellence***

Parameters used in Cost Uncertainty Analysis(SW)

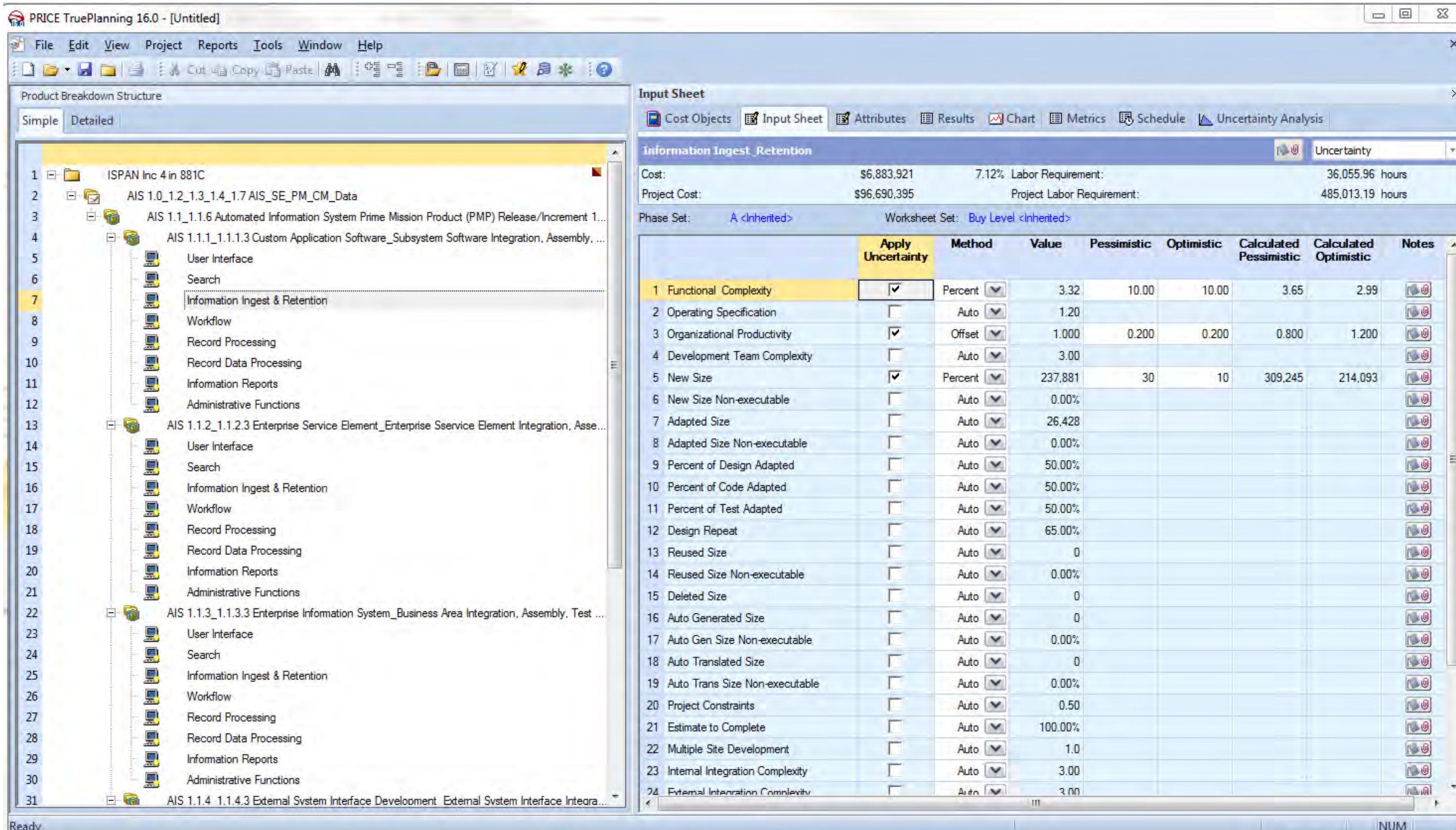
- Reuse
  - The assignment of design reuse is typically a judgment call. Often, the level of cost savings associated with reuse that is realized in development is different from that expected based on the proposal. Cost risk due to reuse should be assessed for each component
  - In True S Reuse is indicated by the amount of Adapted and Reused Code
  - Other factors typically associated with Reuse
    - *Scope of Design*
    - *Experience of the Team*
- Technology Changes/Functional Complexity
  - Technology changes may occur after the initial proposal. The cost risk of technology changes can be estimated by identifying elements that have a possibility of changing and using relevant calibration values to represent possible best/worst case scenarios.

 | *Decades of Cost Management Excellence*

Types of Cost Uncertainty Analyses Supported by TruePlanning® Framework

- Method of Moments
  - FRISK methodology
  - Performed within TruePlanning® framework
  - Simple, fast, accurate
  - Allows estimator to model correlation amount components
- Monte Carlo Simulation
  - Crystal Ball® and @Risk®
  - Performed outside of TruePlanning® framework
  - TruePlanning® interfaces with applications through “companion apps”
  - Third-party software license required

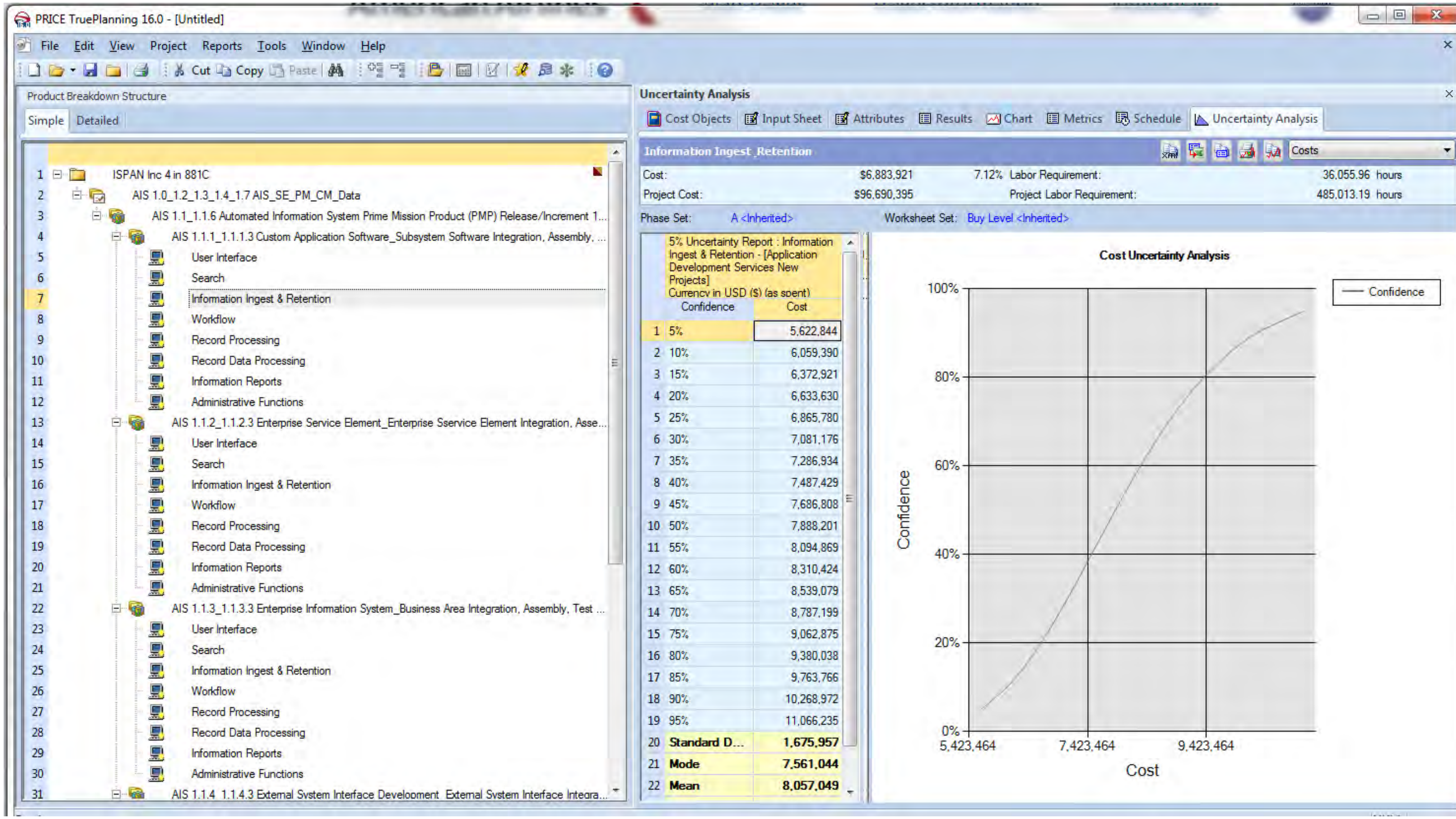
FRISK: Set Cost Uncertainty Inputs



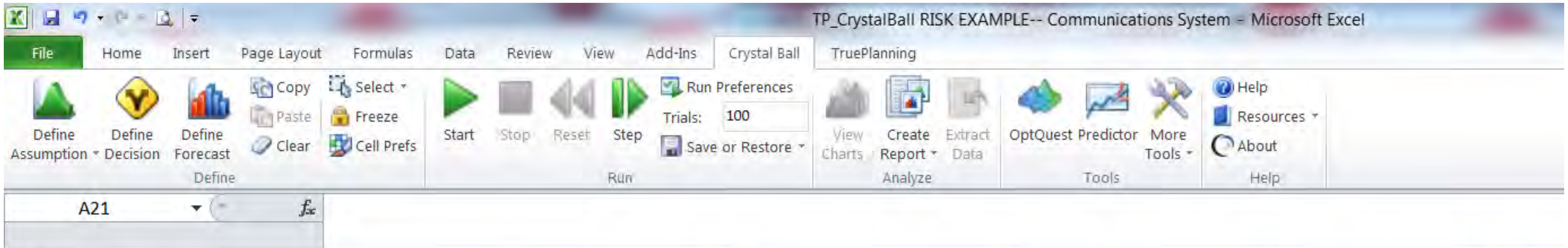
		Apply Uncertainty	Method	Value	Pessimistic	Optimistic	Calculated Pessimistic	Calculated Optimistic	Notes
1	Functional Complexity	☑	Percent	3.32	10.00	10.00	3.65	2.99	
2	Operating Specification	☐	Auto	1.20					
3	Organizational Productivity	☑	Offset	1.000	0.200	0.200	0.800	1.200	
4	Development Team Complexity	☐	Auto	3.00					
5	New Size	☑	Percent	237,881	30	10	309,245	214,093	
6	New Size Non-executable	☐	Auto	0.00%					
7	Adapted Size	☐	Auto	26,428					
8	Adapted Size Non-executable	☐	Auto	0.00%					
9	Percent of Design Adapted	☐	Auto	50.00%					
10	Percent of Code Adapted	☐	Auto	50.00%					
11	Percent of Test Adapted	☐	Auto	50.00%					
12	Design Repeat	☐	Auto	65.00%					
13	Reused Size	☐	Auto	0					
14	Reused Size Non-executable	☐	Auto	0.00%					
15	Deleted Size	☐	Auto	0					
16	Auto Generated Size	☐	Auto	0					
17	Auto Gen Size Non-executable	☐	Auto	0.00%					
18	Auto Translated Size	☐	Auto	0					
19	Auto Trans Size Non-executable	☐	Auto	0.00%					
20	Project Constraints	☐	Auto	0.50					
21	Estimate to Complete	☐	Auto	100.00%					
22	Multiple Site Development	☐	Auto	1.0					
23	Internal Integration Complexity	☐	Auto	3.00					
24	External Integration Complexity	☐	Auto	3.00					

 | ***Decades of Cost Management Excellence***

FRISK: View Cost Uncertainty Outputs at Object-level



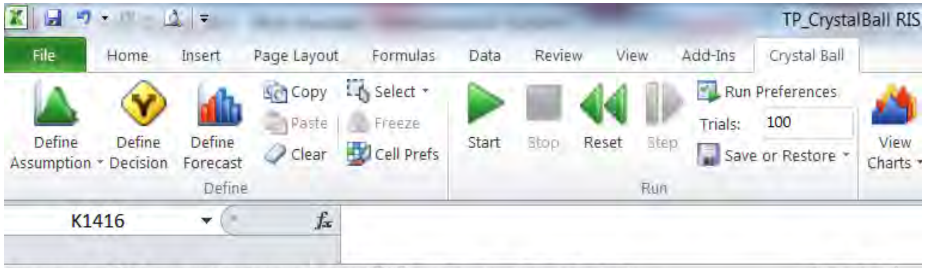
MONTE CARLO: Simulation in Crystal Ball



	A	B	C	D	E	F	G
1	**Risk Example-- Communication System - TruePlanningAdmin**						
2							
9		**Cost Object Name**	**Cost Object Type**	**Cost**	**Input Name**	**Value**	**Unit**
10		Risk Example-- Communication System	System Folder	$ 78,241,377			$
11							
12		Program Management	System	$ 78,241,377			$
13			Project Initiation and Planning for Development	$ 716,990			$
14			Project Management and Control for Development	$ 3,452,238			$
15							
16		Assembly, Integration & Test	Assembly	$ 60,854,459			$
17			System Design	$ 935,740			$
18			Software Integration and Test	$ 1,885,016			$
19							
20							
21		Datalink Antenna	Hardware Component	$ 175,538			$
22			Development Engineering	$ 103,427			$
23			Production Manufacturing	$ 47,100			$
24					Weight of Structure	5	lbs
25					Weight of Electronics		lbs
26					Manufacturing Complexity for Structure	4.71	
27							
28		Pre-Amplifier	Hardware Component	$ 12,642,313			$
29			Development Engineering	$ 4,153,941			$
30			Production Manufacturing	$ 5,847,789			$
31					Weight of Structure	5	lbs
32					Weight of Electronics	3	lbs
33					Manufacturing Complexity for Structure	6.16	
34							
35		Communications Mgt Unit	Hardware Component	$ 9,930,876			$
36			Development Engineering	$ 3,975,417			$

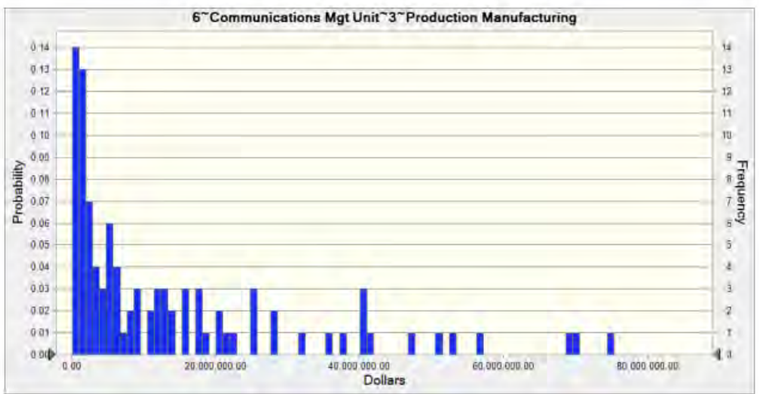
 | ***Decades of Cost Management Excellence***

Crystal Ball: View Cost Risk Outputs at Any Level



Forecast: 6˜Communications Mgt Unit˜3˜Production Manufacturing Cell: D42

Summary:
- Entire range is from 94,944.52 to 138,515,241.77
- Base case is 4,177,469.71
- After 100 trials, the std. error of the mean is 2,494,776.64



Statistics:	Forecast values
Trials	100
Base Case	4,177,469.71
Mean	16,859,827.72
Median	6,514,355.45
Mode	---
Standard Deviation	24,947,766.36
Variance	################
Skewness	2.65
Kurtosis	11.11
Coeff. of Variability	1.48

Forecast: 6˜Communications Mgt Unit˜3˜Production Manufacturing (cont'd)

Percentiles:	Forecast values
0%	94,944.52
10%	675,921.92
20%	1,379,911.20
30%	2,176,221.74
40%	4,041,551.55
50%	6,399,890.31
60%	11,920,821.00
70%	16,254,782.92
80%	25,564,193.55
90%	41,321,603.02
100%	138,515,241.77

 | *Decades of Cost Management Excellence*

Observations/ Q&A

Backup

Parameters used in Cost Uncertainty Analysis (HW)

- Mass (aka, Weight)
  - For Risk Analysis, uncertainty in mass is used as a first order driver
  - Uncertainty values can include:
    - *Contingency Mass (proposed)*
    - *Expected weight growth based on similar past projects*
    - *Factors by subsystem (based on contingencies or default percentage) from Optimistic (e.g., Current Best Estimate mass), to Most Likely, to Pessimistic*
  - Mass Growth set specifically by Component/ Instrument
- Example distributions for Mass Growth

	Approach #1			Approach #2		
	Optimistic	Most-Likely	Pessimistic	Optimistic	Most-Likely	Pessimistic
Weight of Structure/ Electronics	CBE	CBE*(1+ Growth)	Likely*1.3	CBE	(CBE + (CBE*Cont)) *1.3	(CBE + (CBE*Cont)) *1.3

- Mass-growth contingencies, by subsystem, are assigned individually for each Spacecraft Bus element and Payload instrument, via a Dictionary in the Crystal Ball solution below

 | ***Decades of Cost Management Excellence***

Parameters used in Cost Uncertainty Analysis (HW)

- Heritage
  - The assignment of design heritage is typically a judgment call. Often, the level of heritage cost saving that is realized in development is different from that expected based on the proposal. Cost risk due to heritage may be addressed by bounding the best/worst case heritage for specific components or the system in general.
  - Heritage affects Percent New Design (Structure and Electronics):
  - It also affects Engineering Complexity
    - *Scope of Design*
    - *Experience of the Team*
- Manufacturing Complexities
  - Technology changes may occur after the initial proposal. The cost risk of technology changes can be estimated by identifying hardware elements that have a possibility of changing and using relevant calibration values to represent possible best/worst case scenarios.

Uncertainty Ranges

- Example distributions for TruePlanning parameters

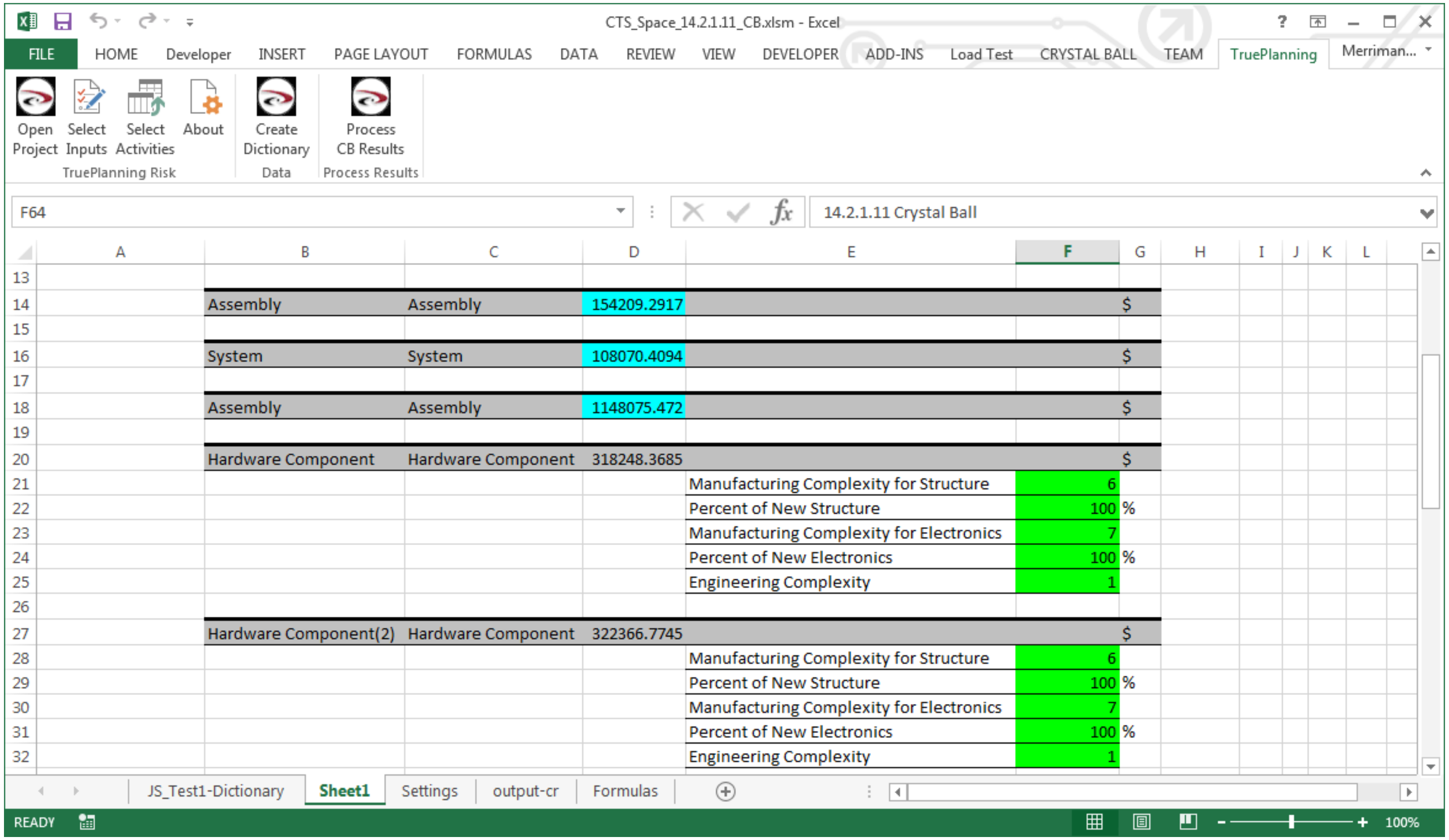
	Approach 1			Approach 2		
	Optimistic	Most Likely	Pessimistic	Optimistic	Most Likely	Pessimistic
Functional Complexity	.85*CBD	CBE	1.15*CBE	.95*CBE	CBE	1.05*CBE
Reused Size	.95*CBE	CBE	1.3*CBE	.95*CBE	CBE	1.15*CBE
Organizational Productivity	.98*CBE	CBE	1.02*CBE	.95*CBE	CBE	1.05*CBE

 | ***Decades of Cost Management Excellence***

Step-By-Step Process:
Crystal Ball Methodology to Assess Risk

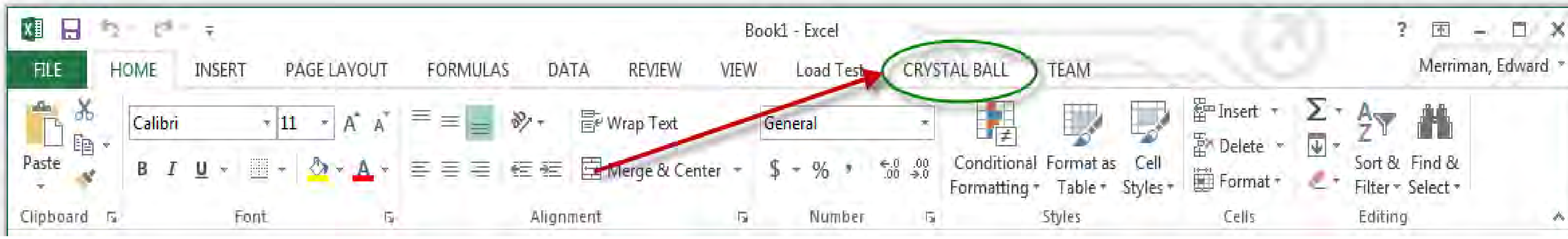
TruePlanning / Crystal Ball Solution Usage

1. Launch the latest version of the TruePlanning/Crystal Ball Solution.
2. Select the TruePlanning ribbon.
3. Click the "Create Dictionary" button and provide a name for the dictionary and a path to the target project.
4. Click the "Open Project" button.
5. Select the inputs to be used in the analysis.
6. For each selected input, select a min/mid/max formula, growth value and indicate if the dictionary should be used.
7. Select the dictionary to be used.
8. Click OK and provide the path to the TruePlanning project to be used.



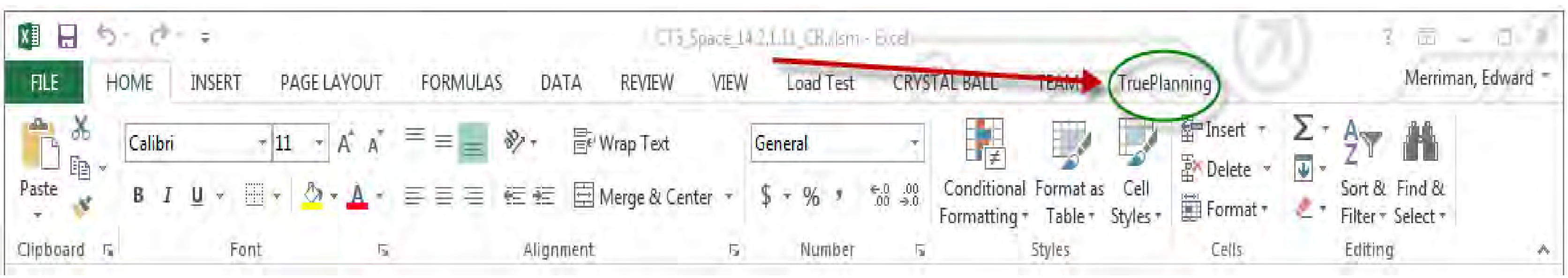
TruePlanning / Crystal Ball Solution Usage

1. Launch Crystal Ball. This will result in Excel opening with the Crystal Ball ribbon.



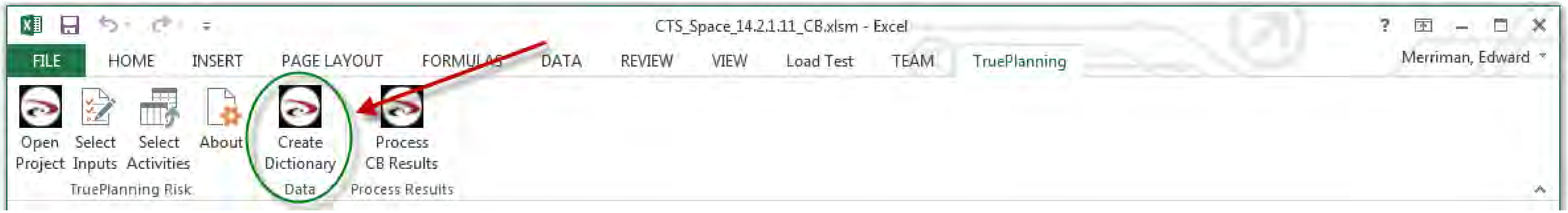
2. Open a copy of the TruePlanning / Crystal Ball Solution in the instance of Excel that opened when Crystal Ball was launched.

3. Click on the TruePlanning ribbon.

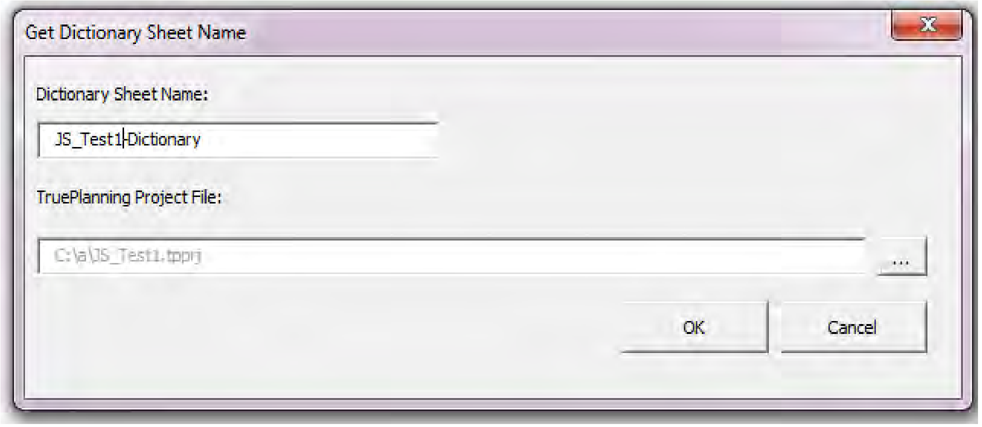


TruePlanning / Crystal Ball Solution Usage

4. Create a dictionary by clicking the “Create Dictionary” button. Note: If the dictionary to be used already exists, skip to step 9.



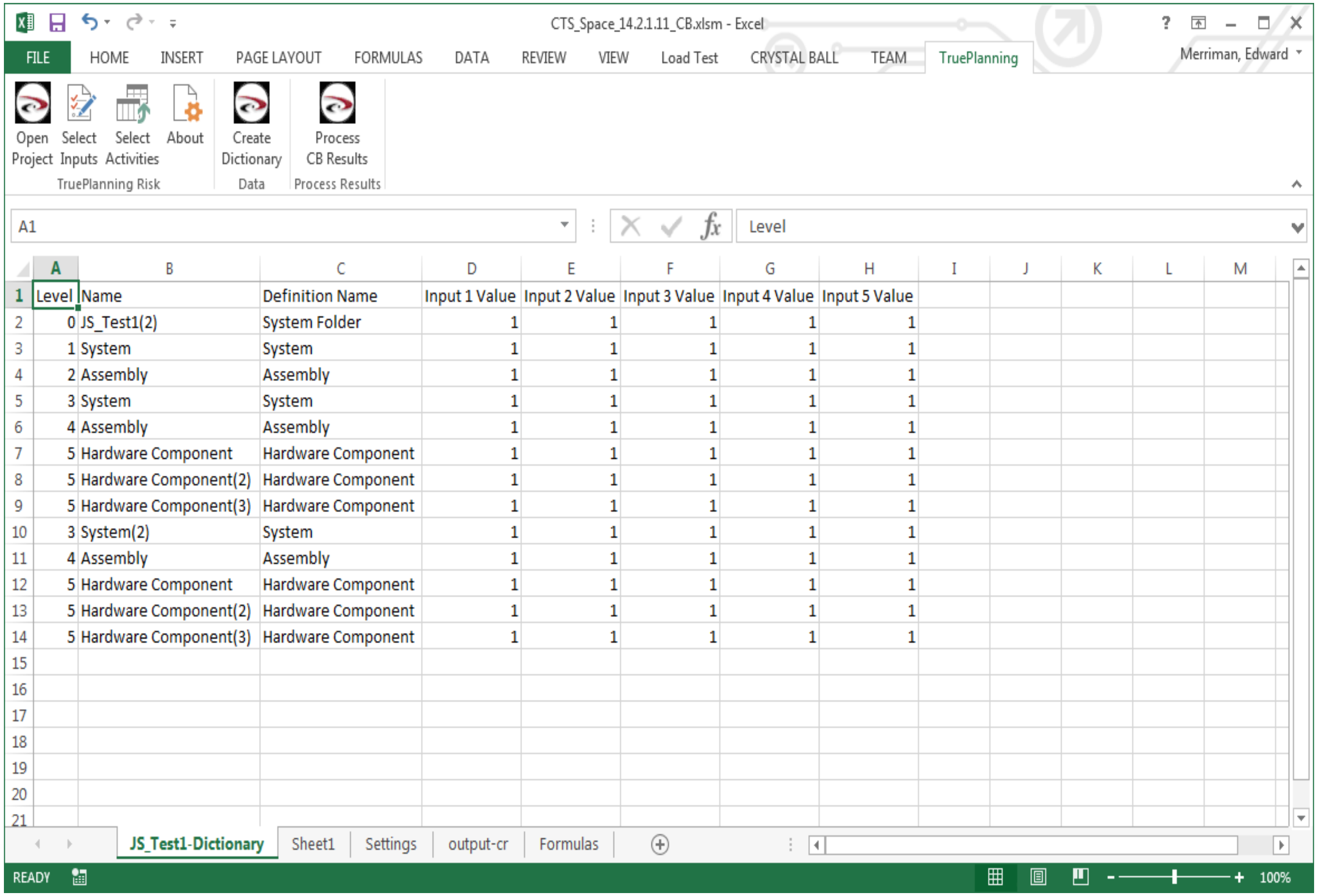
5. Provide a name for the Excel sheet that will receive the dictionary information and provide the path to the TruePlanning *.tpprj file that contains the TruePlanning project to be analyzed. Click “OK”.



6. If prompted, select the appropriate connection name. The default connection name is “(local)”. It is unlikely that users will be prompted to select a connection name.

TruePlanning / Crystal Ball Solution Usage

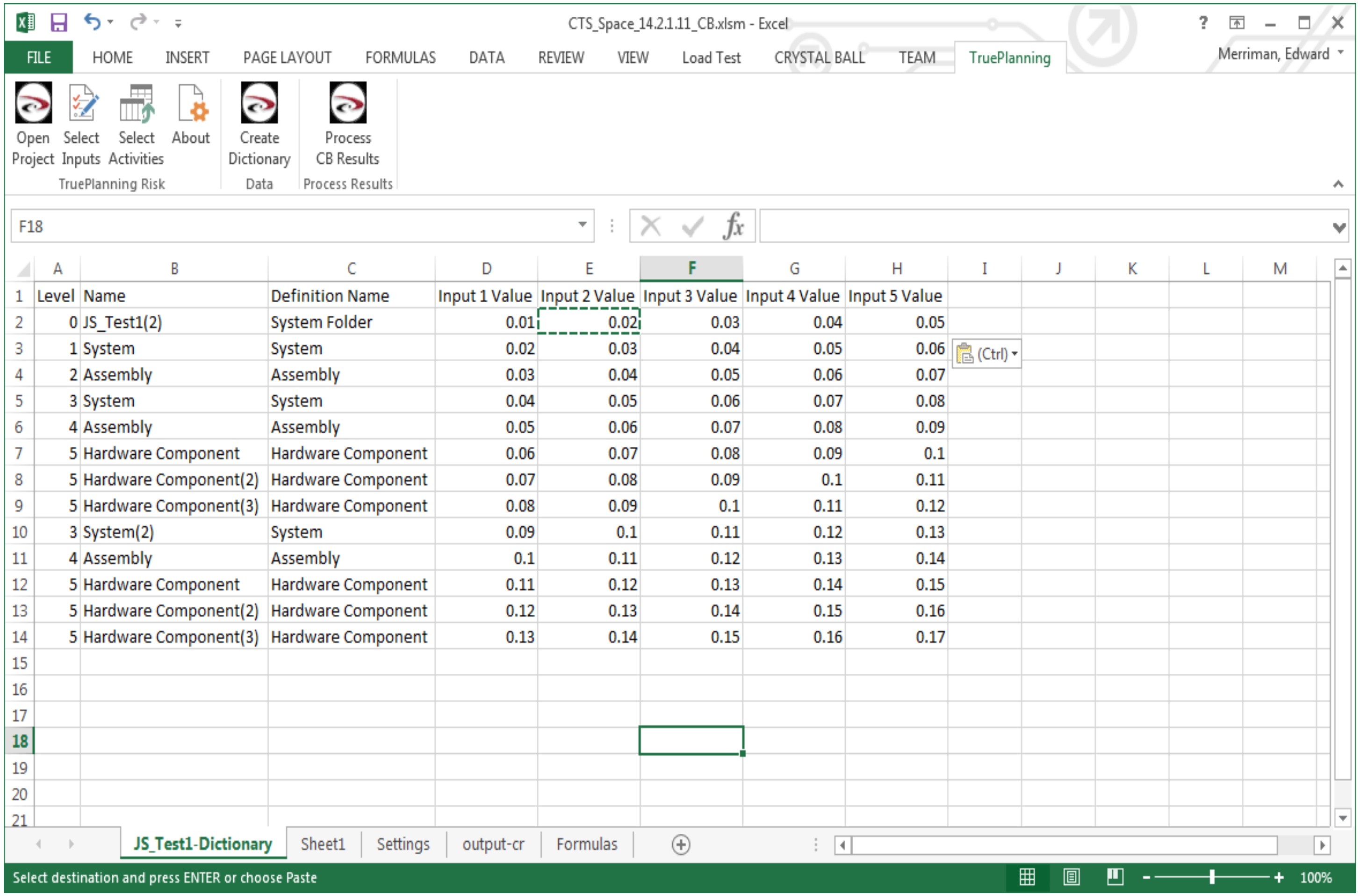
7. A sheet with the name provided in the above dialog will have been created and populated with the PBS from the selected TruePlanning project. Additionally 5 columns are created to represent the 5 inputs that can be set up as Crystal Ball assumptions.



Level	Name	Definition Name	Input 1 Value	Input 2 Value	Input 3 Value	Input 4 Value	Input 5 Value
0	JS_Test1(2)	System Folder	1	1	1	1	1
1	System	System	1	1	1	1	1
2	Assembly	Assembly	1	1	1	1	1
3	System	System	1	1	1	1	1
4	Assembly	Assembly	1	1	1	1	1
5	Hardware Component	Hardware Component	1	1	1	1	1
5	Hardware Component(2)	Hardware Component	1	1	1	1	1
5	Hardware Component(3)	Hardware Component	1	1	1	1	1
3	System(2)	System	1	1	1	1	1
4	Assembly	Assembly	1	1	1	1	1
5	Hardware Component	Hardware Component	1	1	1	1	1
5	Hardware Component(2)	Hardware Component	1	1	1	1	1
5	Hardware Component(3)	Hardware Component	1	1	1	1	1

TruePlanning / Crystal Ball Solution Usage

8. On this sheet, growth or contingency values can be entered for specific PBS elements. They will be used later in the setup of Crystal Balls' assumption distributions.



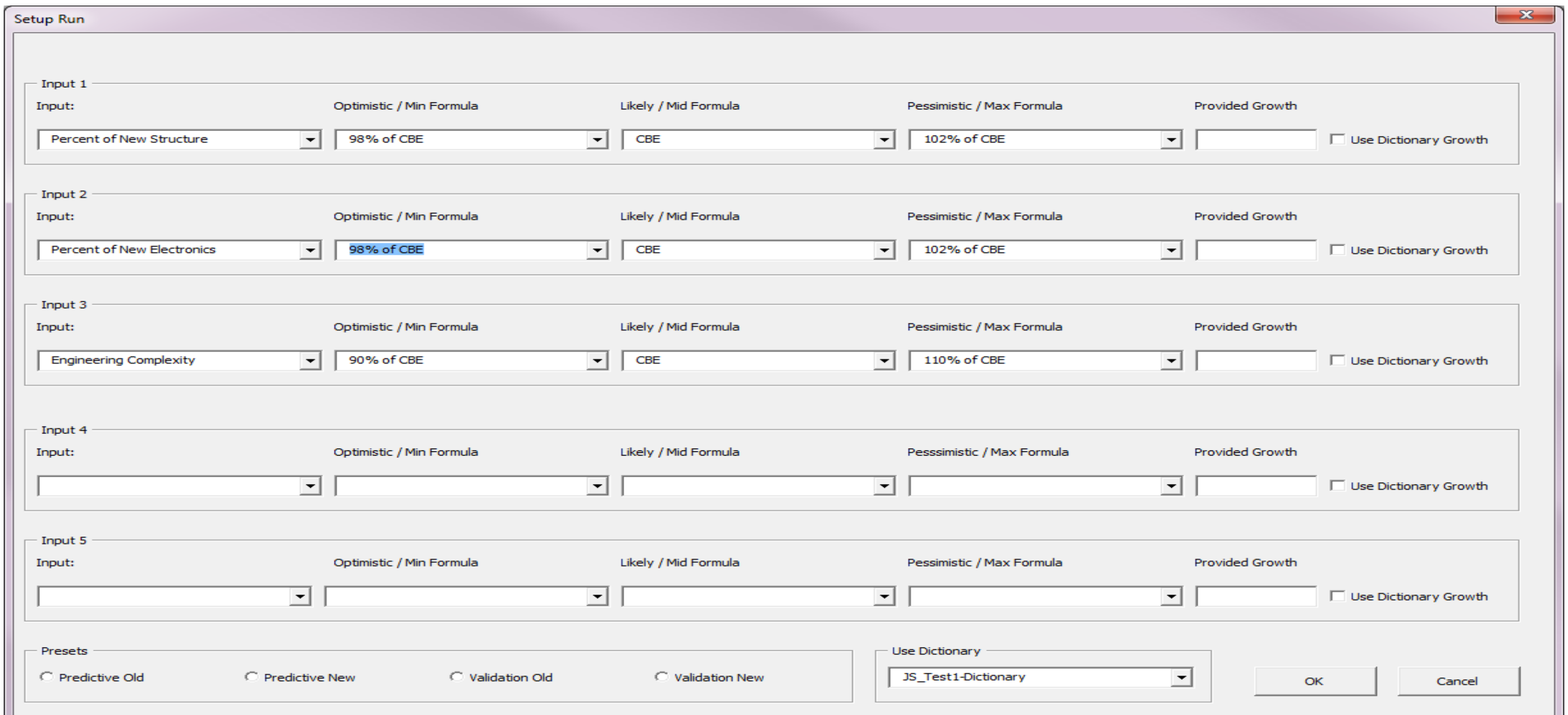
Level	Name	Definition Name	Input 1 Value	Input 2 Value	Input 3 Value	Input 4 Value	Input 5 Value
0	JS_Test1(2)	System Folder	0.01	0.02	0.03	0.04	0.05
1	System	System	0.02	0.03	0.04	0.05	0.06
2	Assembly	Assembly	0.03	0.04	0.05	0.06	0.07
3	System	System	0.04	0.05	0.06	0.07	0.08
4	Assembly	Assembly	0.05	0.06	0.07	0.08	0.09
5	Hardware Component	Hardware Component	0.06	0.07	0.08	0.09	0.1
5	Hardware Component(2)	Hardware Component	0.07	0.08	0.09	0.1	0.11
5	Hardware Component(3)	Hardware Component	0.08	0.09	0.1	0.11	0.12
3	System(2)	System	0.09	0.1	0.11	0.12	0.13
4	Assembly	Assembly	0.1	0.11	0.12	0.13	0.14
5	Hardware Component	Hardware Component	0.11	0.12	0.13	0.14	0.15
5	Hardware Component(2)	Hardware Component	0.12	0.13	0.14	0.15	0.16
5	Hardware Component(3)	Hardware Component	0.13	0.14	0.15	0.16	0.17

 | ***Decades of Cost Management Excellence***

TruePlanning / Crystal Ball Solution Usage

9. Click the “Open Project” button on the TruePlanning ribbon.

10. The Setup Run dialog is launched.



TruePlanning / Crystal Ball Solution Usage

11. The Setup Run dialog allows users to configure up to 5 inputs from the Hardware Component Cost Object to be used as assumptions in the Crystal Ball analysis.

12. To setup an input 6 attributes must be set:
    a. Name of input
    b. Optimistic / Min formula
    c. Likely / Mid formula
    d. Pessimistic / Max formula
    e. Provided Growth/Contingency value
    f. Use Dictionary Growth/Contingency checkbox

13. List of available inputs:
    a. Weight of Structure
    b. Weight of Electronics
    c. Percent of New Structure
    d. Percent of New Electronics
    e. Engineering Complexity
    f. Manufacturing Complexity of Structure
    g. Manufacturing Complexity of Electronics

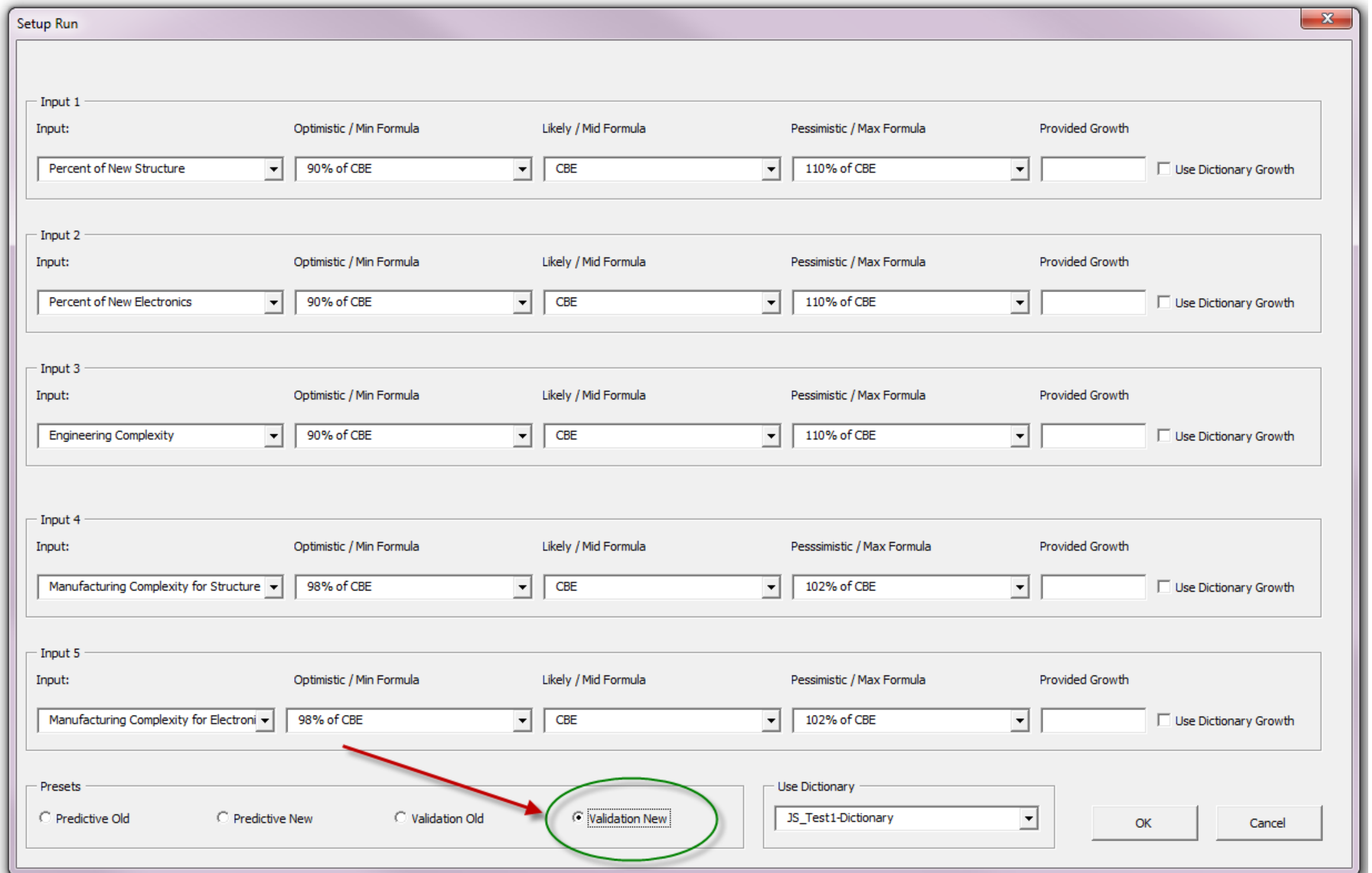
TruePlanning / Crystal Ball Solution Usage

14. List of formulas:
    a. 80% of CBE
    b. 90% of CBE
    c. 95% of CBE
    d. CBE
    e. 102% of CBE
    f. 105% of CBE
    g. 110% of CBE
    h. 120% of CBE
    i. CBE * (1 + Growth)
    j. Likely * 1.3
    k. CBE * 1.3
    l. (CBE + (CBE*Contingency))*1.3

15. Growth / Contingency values: The “Setup Run” dialog has two settings for suppling growth/contingency values to the formulas used to set up the Crystal Ball assumptions.
    a. Provided Growth input: this input allows users to set a single value for all assumptions created for an input.
    b. Use Dictionary Growth: this check box indicates that formulas for this input should obtain their growth/contingency values from the selected dictionary. This allows unique growth/contingency values to be used for each PBS element.

TruePlanning / Crystal Ball Solution Usage

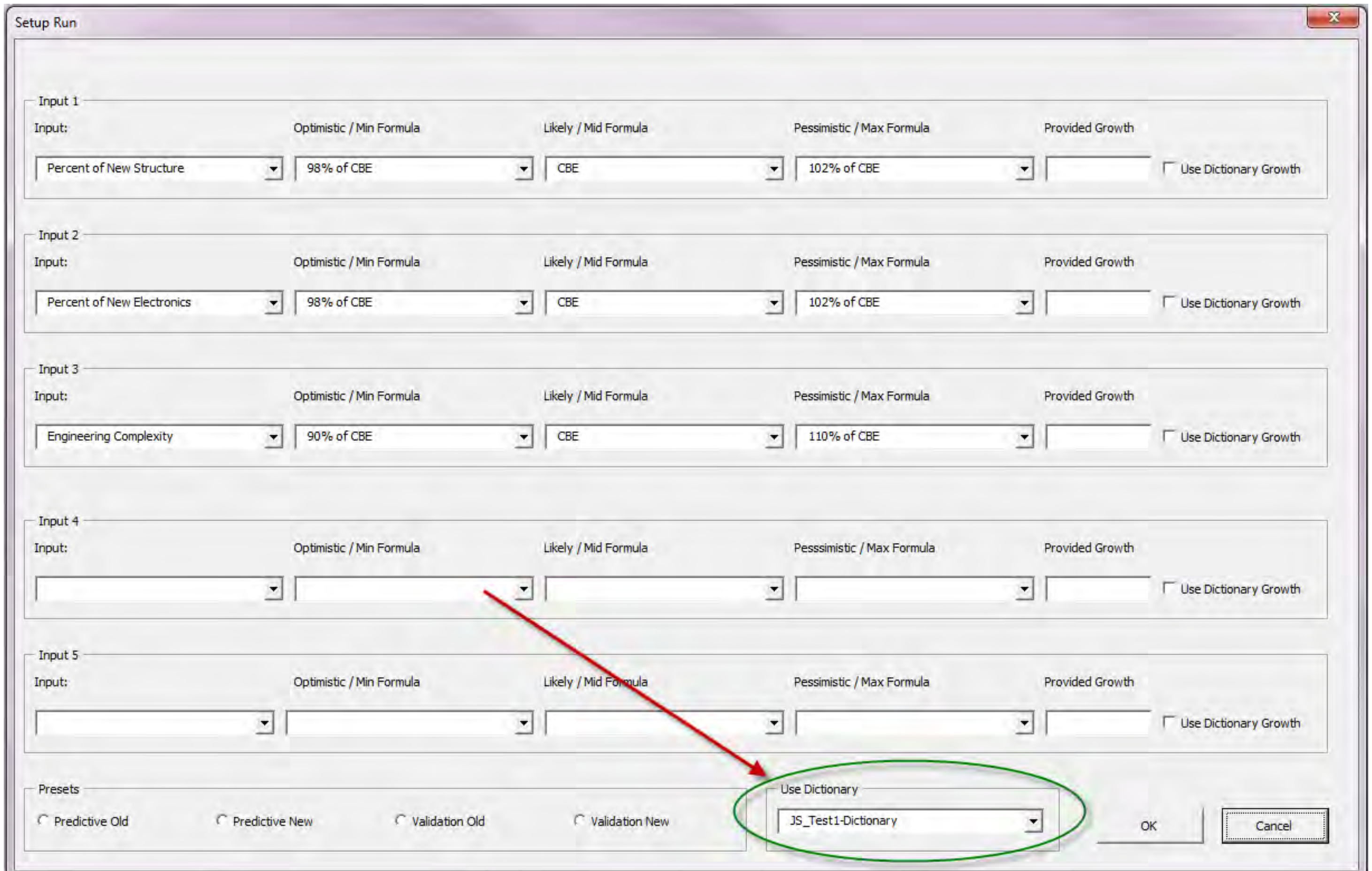
16. Presets: There are four radio buttons in a group box at the bottom of the Setup Run dialog. Selecting one of these radio buttons will fill in the settings of the dialog for a preconfigured setup. This allows users to quickly reproduce the same setup. The settings defined by a preset can be configured.



 | *Decades of Cost Management Excellence*

TruePlanning / Crystal Ball Solution Usage

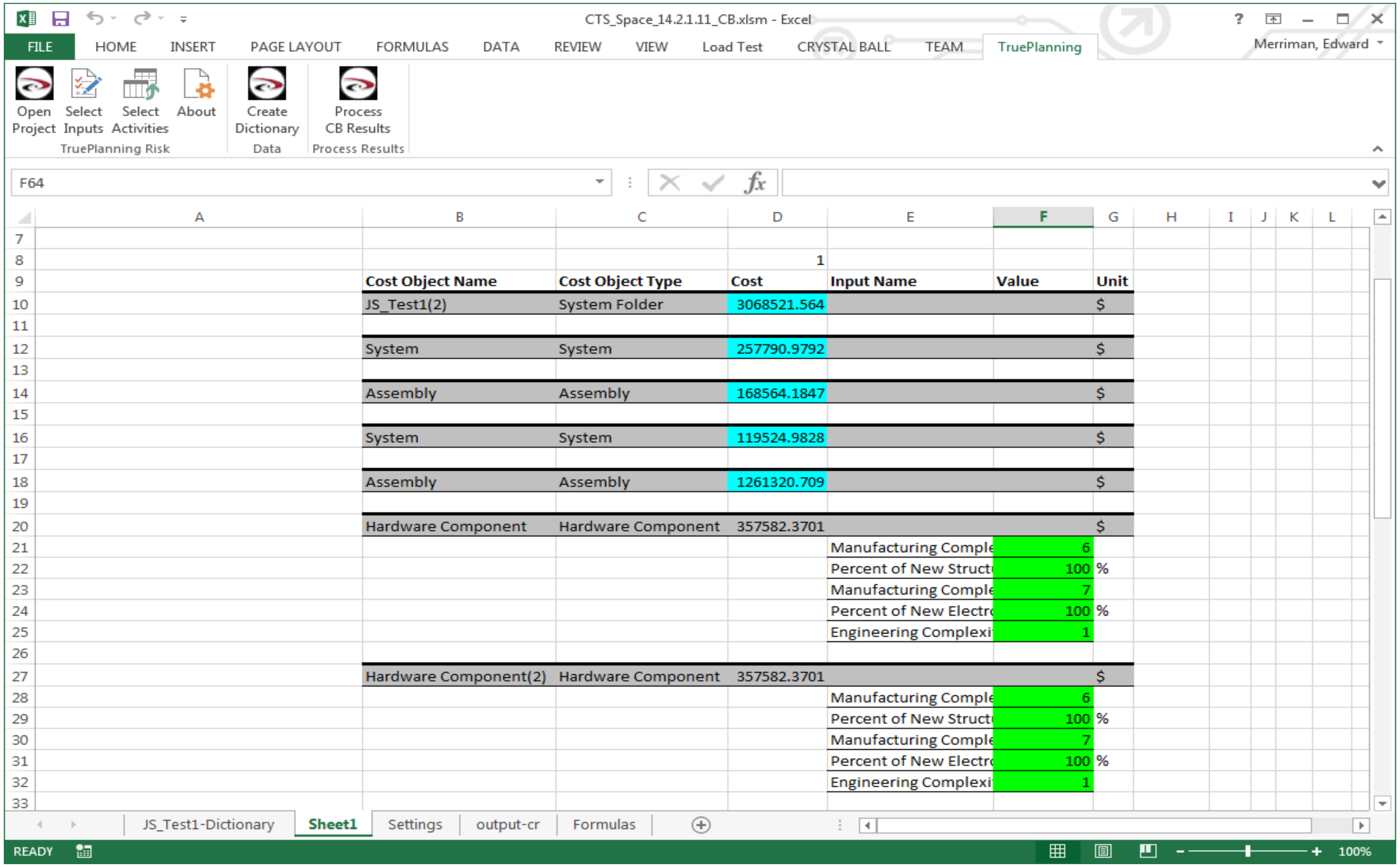
17. Select a dictionary to be used. Note: A dictionary must be selected. The selected dictionary needs to have a PBS structure that matches the TruePlanning project that will be selected for use with the Crystal Ball analysis. Use the pulldown at the bottom of the dialog.



 | ***Decades of Cost Management Excellence***

TruePlanning / Crystal Ball Solution Usage

18. Click “OK” on the Setup Run dialog to setup the Crystal Ball forecasts and assumptions. Users will be prompted to select a TruePlanning project file (*.tpprj). When complete “Sheet 1” will contain the PBS of the targeted TruePlanning project with the appropriate Cost Objects’ estimate costs set as forecasts and the selected inputs set as assumptions.



 | ***Decades of Cost Management Excellence***

Observations/ Q&A

 | ***Decades of Cost Management Excellence***

Using Functional Size and Source Code to estimate ERP and Cloud Based Big Data Analytics

Software & IT CAST

NGA

August 2016

David P. Seaver
Senior Technical Analyst
National Security Agency

Outline

- Counting before Estimating
- Size Estimation
- Functional Size Example
- NSA Sizing Customizations
  - Analytics
  - ERP/COTS
- Wrap up

Before you estimate you need to Count!

Count what you have done before, so you can estimate using historical data.

Count what you are going to do, so you can have configuration control of you cost estimate and your program

Count, Calculate, Judge and Advise

Count if possible

- Earlier is always better than latter
- Must be correlated to the scope of what you are estimating

Calculate when done counting

- Convert what you have counted into an estimate using history from somewhere. Local history is always best.
- History has to relate to what you count

Judge and Advise only as a last resort

- Only when you don't have data
- Pressure will be to be optimistic away from reality

NSA Counts.....

Source code for every possible project.

- Use a NSA tailored version of University of Southern California UCC
  - Customization identifies
    - GOTS, COTS and FOSS
    - Test Code
    - Duplicate code
    - Auto-generated Code

Functional size of Requirements using a streamlined Function Point technique

- Simple Function Points

NSA Does Not Count

Delivered Function Points

- Information is not accessible
- Difficulty in obtaining access to working applications
- Not enough resources to perform manual counts

We are evaluating an automated function point capability that we will be piloting in the Fall of 2016

- CAST Software's implementation of the CISQ functional size standard
- www.cisq.org & www.castsoftware.com
- Automated Counts will be done in tandem with USC UCC code counts on projects that are at or near completion

Size: Source Lines of Code Issues and Solutions

- No defined counting rules of standards organization
- Inconsistent rules means there is no reliable and verifiable industry data
- Penalizes efficient software writing and incentivizes poor coding
- Heavily dependent on developer skills and style
- Difficult to estimate early in life cycle.
- Not all development creates source code

- Use USC UCC to standardize rules
- Don't pay for lines of code, We purchase systems not code
- Agree, and most developers don't use code to **plan** projects
- Easy to count at completion
- Use functional size too

Functional Size

Why does NSA use Functional Size?

What is Functional Size?

What are Simple Function Points (SFP)?

Software Size

Source Lines of Code (SLOC)

- Difficult for now SW engineers to estimate source code early in a program (and most for most SW engineers too)
- Only becomes really viable during the design phase
- Complications
  - DBMS
  - COTS, FOSS and GOTS
  - Difficult to link code back to requirements
- Can be measured on a completed project
- **Code metrics not meaningful to developers**

Functional Size

- Can be estimated early in the program
- Easy to link to requirements
- Can be used to estimate code size
- Can be used to estimate effort and cost per requirement
- **Function Point metrics not meaningful to anyone who is not a cost estimator**

Function Point Analysis

Classic Function Point analysis (IFPUG version) is based on 5 components

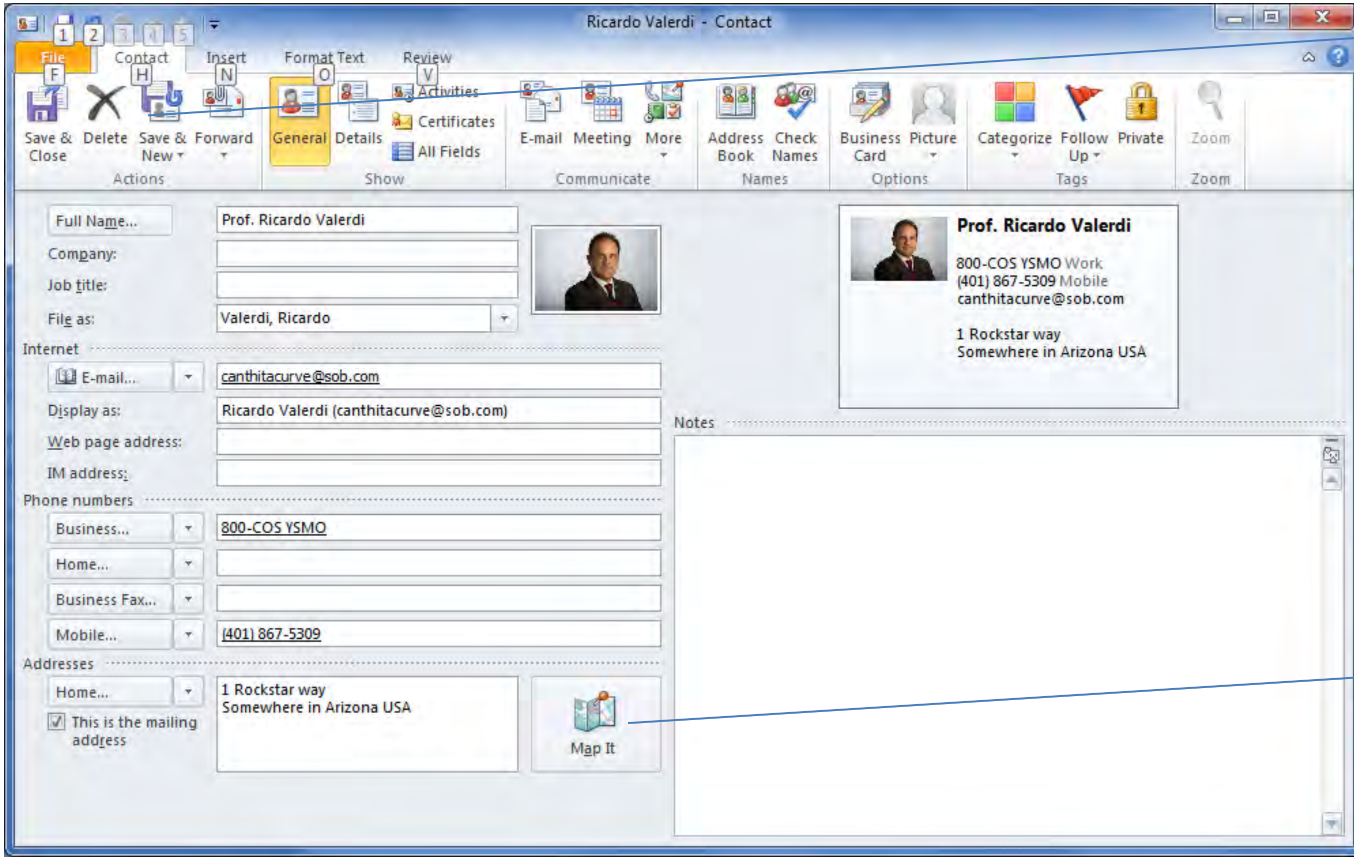
- Inputs (adding, updating, deleting data in an application)
- Outputs/Reports that displays or sends data that is processed
- Queries that retrieve and display data (no processing of the data)
- Internal Logical Files which is data maintained by your application by either an input or a report
- External Interface Files which is data maintained in another application that your application interfaces with for to accomplish user required functionality

Function Point Analysis

Function points are calculated by assigning a complexity to the transaction or data entities based on the number of unique fields in each, the files referenced to meet user requirements, or the data types for each of the logical file categories.

- Based on scoring complexity are Low, Average or High
- Function Points scores can range between 3, 4, 5, 7, 10 and 15 function points depending on the complexity
- www.ifpug.org

Input Transaction: New Contact



Can input, edit or delete data on this screen

3 transactions one data entity (save)

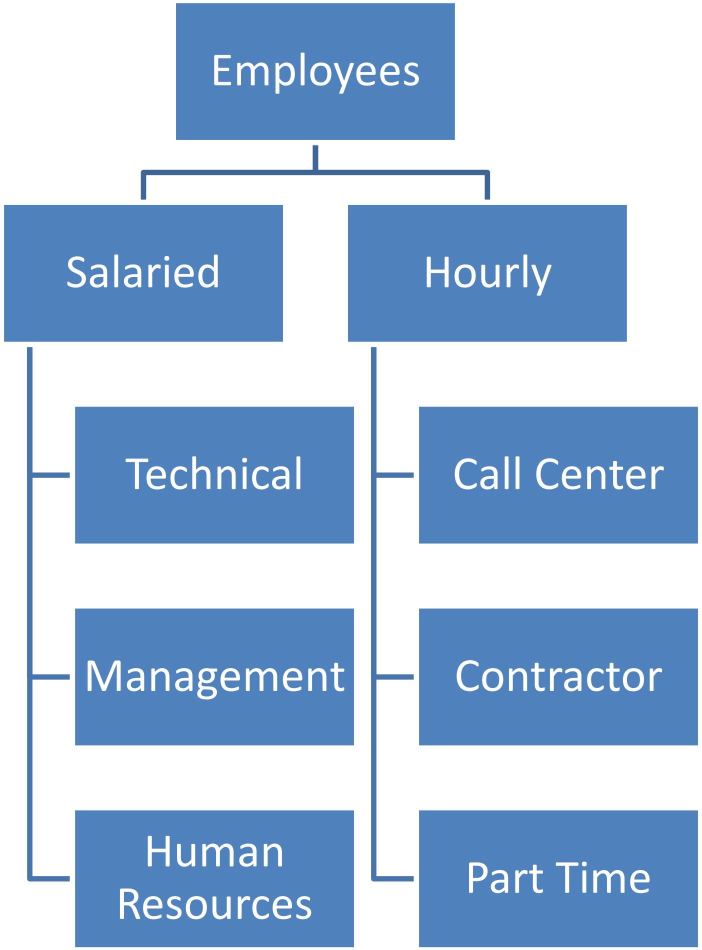
26 fields 9(detail button)

Map It could be a query

Buttons are for navigation?

Inputs process information that enters the application, the information maintains a logical file

Logical File



External
Vendor Data Read-Only

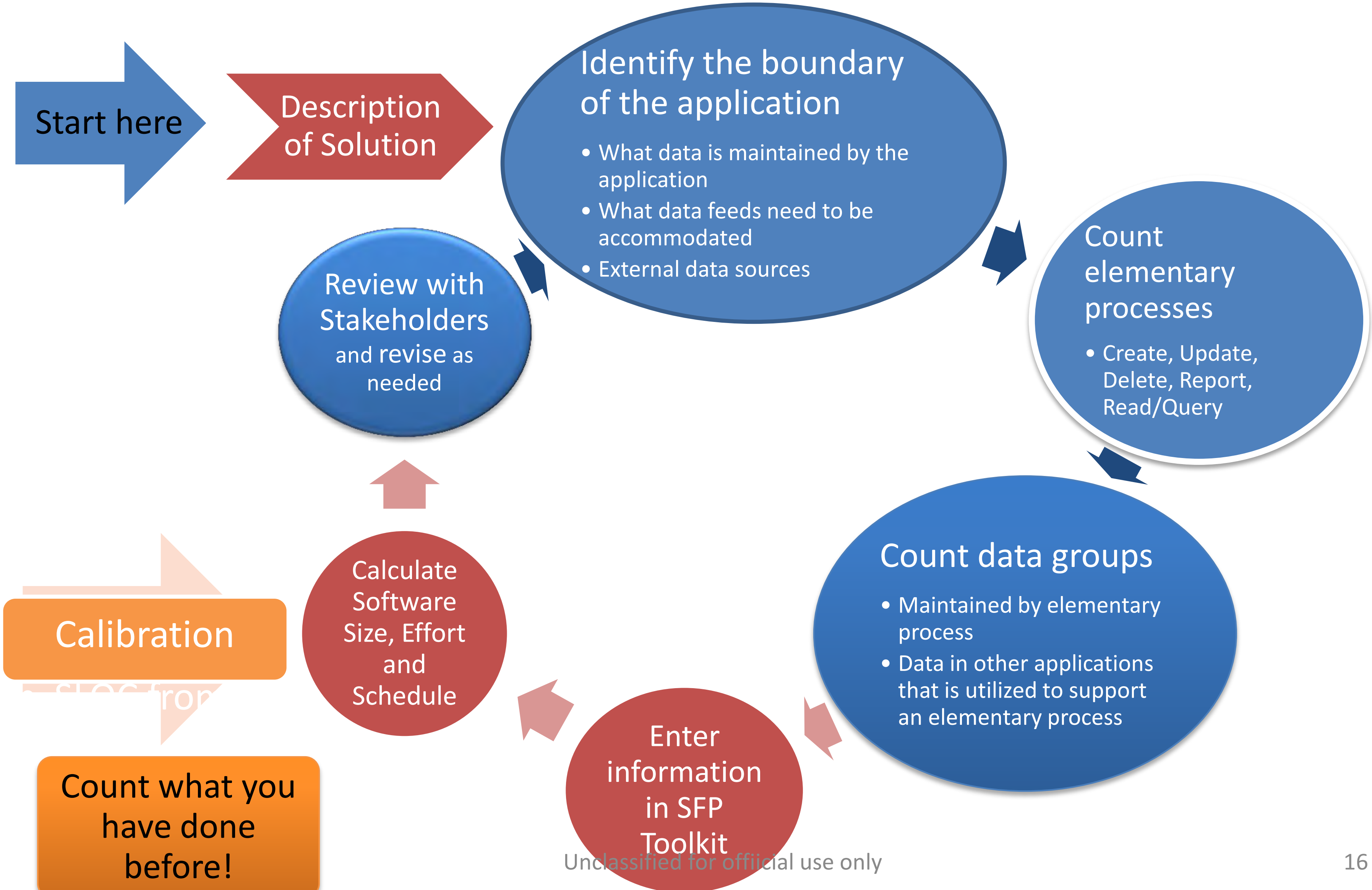
NSA Reality

- Estimates are required very early in the life cycle
- We don’t have design artifacts
- We estimate from a variety of requirements documents
  - Do not have level of detail of fields on screens
  - Cant count or estimate fields in a database table
- We have adopted a streamlined Functional Size technique called Simple Function Points.

Simple Function Points

- Simple Function Points (SFP)
- SFP counts two components
  - Elementary Process: which is defined as the smallest level of activity that is meaningful to the user
  - Logical Data Groups: A user identified group of data or control information maintained by an application
  - *www.sifpa.org/en/index.htm* for more information
- Since we are counting requirements it is not possible to perform an IFPUG function point count.
- User group meetings are in Italy

Process to Estimate Software



The RV Baseball Analytics

This is based on a real life example, the words have been changed to protect the innocent

EXAMPLE

The RV Baseball Analytics

Prof. R. Valerdi our hero

- **Program Description:** The RV Baseball Analytic (RVBA) Ingest and process all available data from MLB ballparks and Television Broadcasts to provide state of the art analytics for MLB teams and MLB.
- 5 RVBA Requirements were identified
  - Ingest and Automation
  - Balls and Strikes Analytics
  - Instant Replay Analytics
  - Umpire Analytics
  - Baseball Commissioner Analytics

Functional Size Ingest and Processing

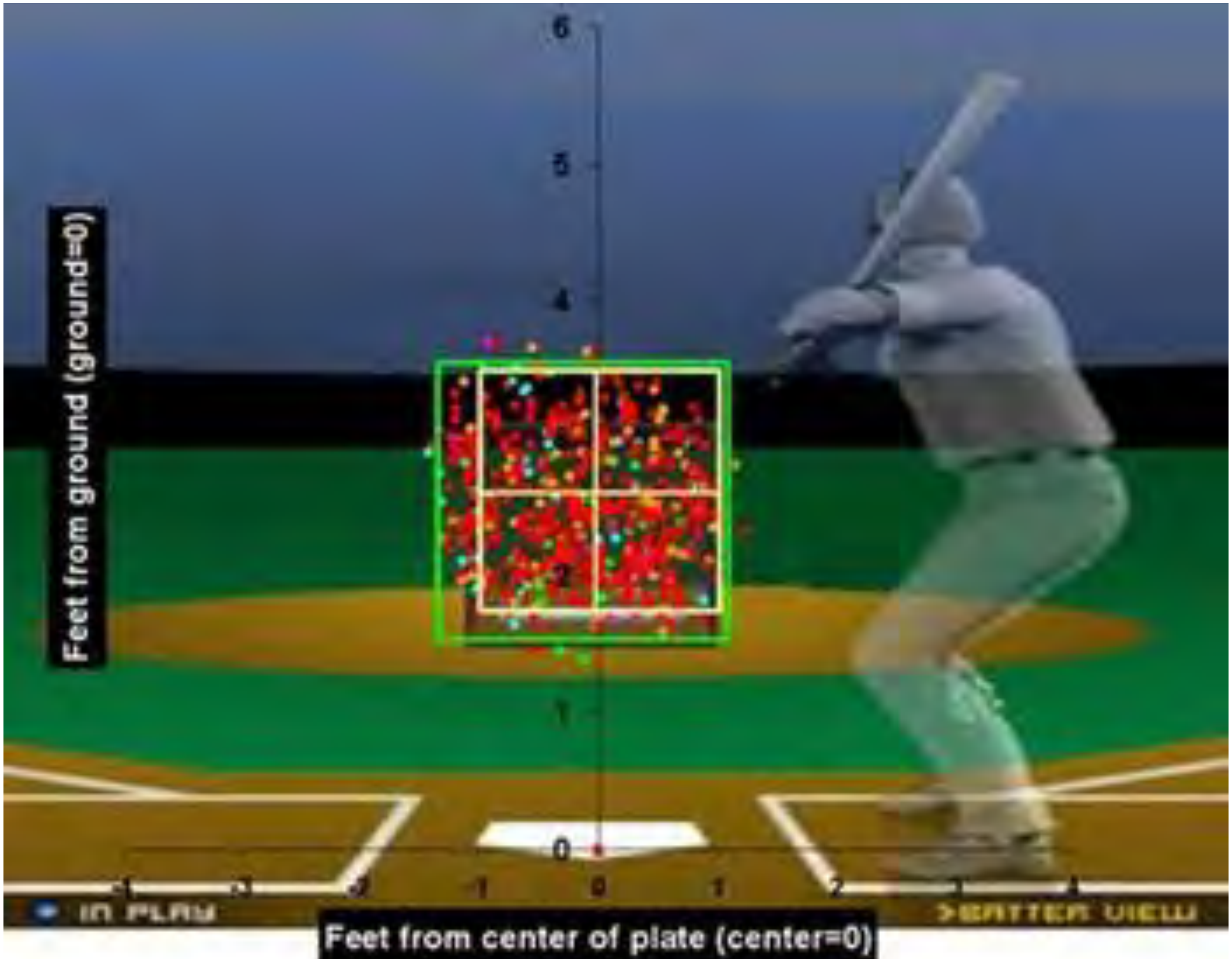
Sort	Requirement	Requirement Description	Note	Create	Update	Delete	Read	Report	Save	Multiplier	Data Multiplier	Transaction Count	Data Count	Function Points	ESLOC
1	Data Ingest & Processing	> 70 different data types		1	1		1		1	75	75	225	75	1,560	85,800
1	Data Ingest & Processing	meta data associated with core data types		1	1		1		1	75	75	225	75	1,560	85,800
1	Data Ingest & Processing	Monitoring and screening of the data ingest stream		1	1		1		1	1	1	3	1	21	1,144
	Subtotal											453	151	3,141	172,744

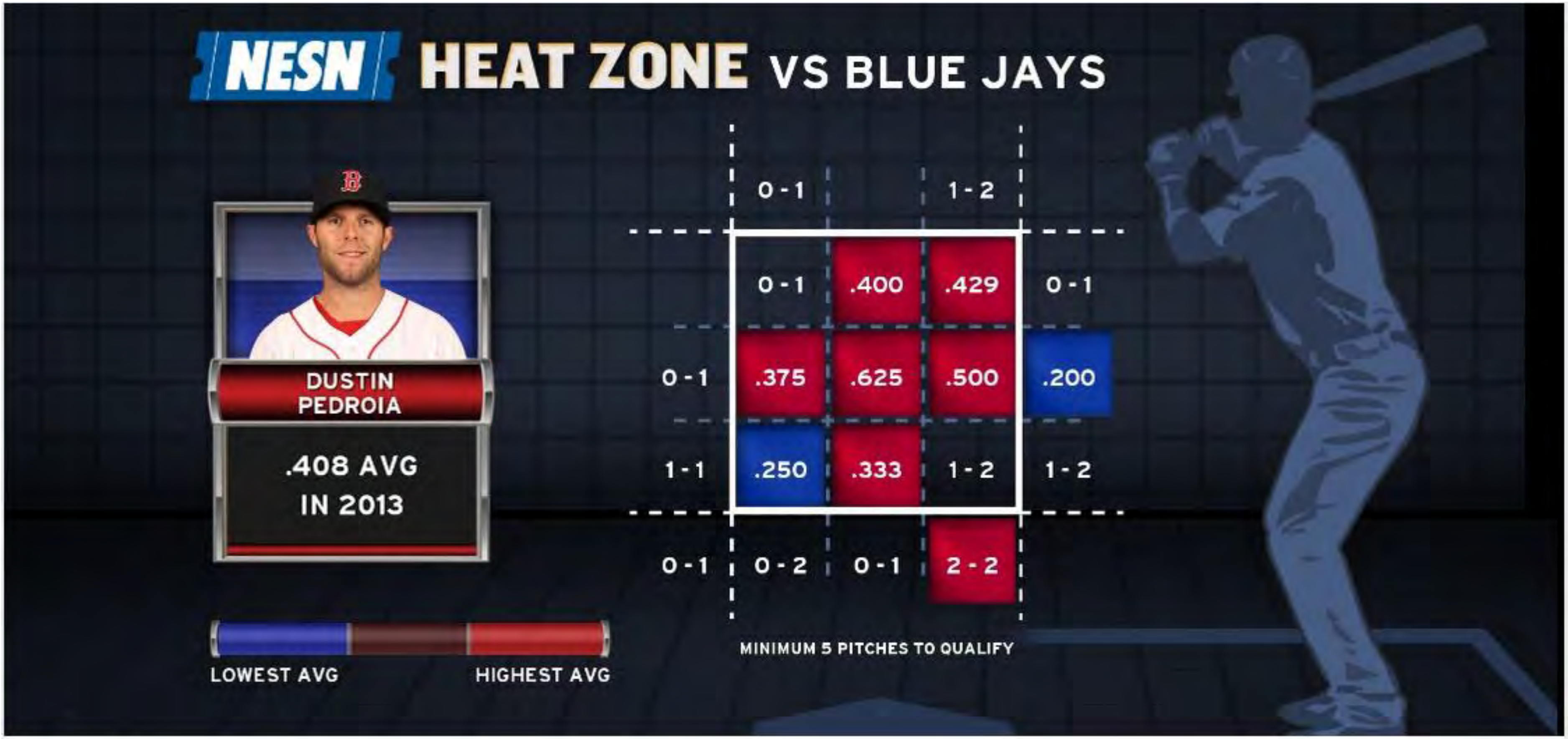
Functional Size for Analytics

Sort	Requirement	Requirement Description	Note	Create	Update	Delete	Read	Report	Save	Multiplier	Data Multiplier	Transaction Count	Data Count	Function Points	ESLOC
2	Balls and Strikes Analytics	GUI for configuration, 5 data toggles per analytic		1	1		5		1	4	4	28	4	157	8,624
2	4 Analytics	2 reports per analytic (table and graphic) 4 additional configuration options in the report itself					4	4		4		32	0	147	8,096
2	Instant Replay Analytics	GUI for configuration, 8 data toggles		1	1		1		1	1	1	3	1	21	1,144
2	2 Analytics	two different video options, same controls					1		1	2		2	0	9	506
2	Umpire Analytics	GUI for configuration, 8 data toggles		1	1		1		1	1	1	3	1	21	1,144
2	5 Analytics						5	1	1	5	0	30	0	138	7,590
2	Baseball Commissioner Analytics			1	1		1		1	1	1	3	1	21	1,144
2	15 Analytics						5	1	1	5	0	30	0	138	7,590
	Subtotal											131	7	652	35,838


Lefty
Pitch Frequency
Least
Most
Righty


FIELDER STATS
#7 Reed Johnson
First step
Top speed
Acceleration
82.7 ft
80.9 ft
FIELDER STATS
#22 Jason Heyward
First step
Top speed
Acceleration


Feet from ground (ground=0)
Feet from center of plate (center=0)
IN PLAY
>BATTER VIEW


NESN HEAT ZONE VS BLUE JAYS
DUSTIN PEDROIA
.408 AVG IN 2013
0 - 1
1 - 2
0 - 1
.400
.429
0 - 1
0 - 1
.375
.625
.500
.200
1 - 1
.250
.333
1 - 2
1 - 2
0 - 1
0 - 2
0 - 1
2 - 2
MINIMUM 5 PITCHES TO QUALIFY
LOWEST AVG
HIGHEST AVG

Sizing Frameworks for Analytics

Data Ingest and processing

- Content data - the basic data you want to analyze
- Metadata - data that provides context for your content data
  - Weather
  - Time of Day
- Reference Data – other data sources that may be referenced
  - Ball park dimensions and features

Object Creation – process data to prepackage information of interest and make it readily accessible

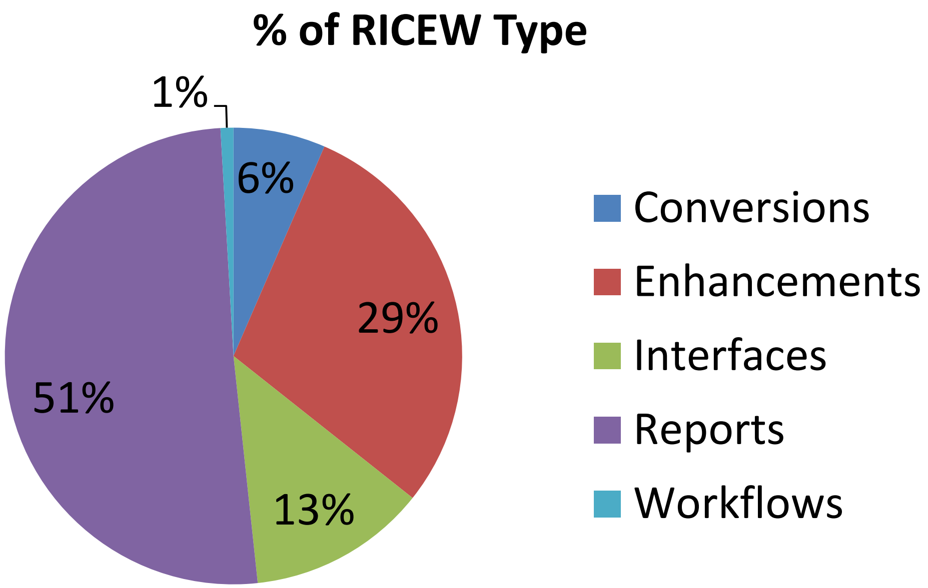
- Player objects
  - Pitcher
  - Fielder
  - Batter
  - Catcher

A brief discussion on how this works for COTS and ERP Applications

COTS AND ERP APPLICATIONS

ERP Perspective

- ERP platform is PEOPLESOFT
- Financials and HR
- RICEW work unit
  - Reports 390
  - Interfaces 97
  - Conversions 50
  - Enhancements 224
  - Workflow 7



Ongoing Work COTS and ERP

- Developed SFP size for all COTS estimates to date, on average 24 Function Points/Work Unit
- Attempting to derive a Function Point/Person Month metric
- Working with PEOPLETOOLS to update size metrics
  - Outputs CEMLI metrics (next slide) and RICEW object counts
- Running CAST analysis on PEOPLESOFT Code
  - Automated Function Points does not work on PEOPLESOFT
- Attempting to incorporate configuration activities into sizing and estimation of COTS. Have been using $ per work unit for configuration work to date
- Developing a framework to estimate the O&M costs for PEOPLESOFT suite

COTS Sizing Framework-CEMLI

Sizing Categorization	Sizing comment
Configurations : Configure the existing, pre-built application features according to your client's requirement. Changing setups and profile values can be the example of configurations.	Typically no code developed, use applications existing screens to implement capability
Customization : Customization means altering/changing the standard objects or creation of custom object to meet client's business need. It may be Extensions or Modifications.	Custom Code
Extensions : Extension means creating custom code from scratch, existing objects (views, packages and java classes etc) can be used. It is having different behavior from seeded one.	Custom Code
Modifications : Modifications is enhancing/changing the existing code to meet the client's requirements. It is the modification of seeded behavior.	Custom Code
Localization : It is to define the different legislative support provided by oracle Applications based on country/region/language requirements.	Typically no code developed, use applications existing screens to implement capability
Integration : It can be Data Integration or Application Integration, options for these two are Open Interface tables, APIs, EAI(Enterprise Application Integration Tools), BPEL, AQ, EDI etc.	Can be either a configuration activity or require code development.
Personalization : Tailoring the layout or visibility of page content to meet client requirements is Personalization. Changing the user interface (UI) look-and-feel, making any field visible/enabled/disabled/mandatory/non mandatory comes under Personalization.	Typically no code developed, use applications existing screens to implement capability

Conclusions

- We are pleased with SFP results to date
- Estimates are produced more quickly with less overhead
- Workload includes more projects with little or no code developed focus mostly in the business side of the agency
- Will update on progress with CAST at a later date. Install was completed August 16. Will be running some cloud based analytics and PEOPLESOFT as our initial test cases

Back up slides

NGA IT CAST 8.2016

Words

Word	IFPUG Map	Transaction	Data	Comment	Data Comment	Column1	Create	Update	Delete	Report	Read	Save	AFP	JAVA SLOC
Accept	Input	2	1	receive, process and store data			1				1	1	16	891
Add	Input	2	1	receive, process and store data			1				1	1	16	891
Adjust	Input	1		Update				1					5	253
Allocate	Report	1		Process Data for Report	possible save					1		0	5	253
Analyze	Report	1		Process Data for Report	possible save					1		0	5	253
Apply	Input	2	1	receive, process and store data			1				1	1	16	891
Assign	Input	2		Update, and read				1			1		9	506
Associate	Input	2		Update, and read				1			1		9	506
Browse	Query	1		query/read							1		5	253
Change	Input	1		Update, and read				1					5	253
Combine	input	2		Update, and read				1			1		9	506
Correlate	Report	1		Process Data for Report	possible save					1		0	5	253
Create	Input	2	1	receive, process and store data			1				1	1	16	891
Data Source	Input	2	1	receive, process and store data			1				1	1	16	891
Delete	Input	1		delete					1				5	253
Detect	Report	1		Process Data for Report	possible save					1		0	5	253
Display	Report	1		Process Data for Report	possible save					1		0	5	253
Distribution	Report	1		Process Data for Report	possible save					1		0	5	253
Enquire	Query	1		query/read							1		5	253
Enrich	Input	2	1	receive, process and store data			1				1	1	16	891
Enter	Input	2	1	receive, process and store data			1				1	1	16	891
Export	Report	1		Process Data for Report	possible save					1		0	5	253
Extract	Query	1		query/read							1		5	253
Generate	Report	1		Process Data for Report	possible save					1		0	5	253
Identify	Report	1		Process Data for Report	possible save					1		0	5	253
Import	Input	2	1	receive, process and store data			1				1	1	16	891
Inform	Report	1		Process Data for Report	possible save					1		0	5	253
Ingest	Input	2	1	receive, process and store data			1				1	1	16	891
Inputs	Input	2	1	receive, process and store data			1				1	1	16	891
Inquire	Query	1		query/read							1		5	253
Interface	Input/Reports	5	1	potential to send and receive data	potential to save data		1	1	1	1	1	1	30	1650
Knowledge	Report	1		Process Data for Report	possible save					1		0	5	253
Link	Input	2		receive, process and store data			1				1	1	16	891
List	Query	1		query/read							1		5	253
Log	Input	2	1	receive, process and store data			1				1	1	16	891
Maintain	Input	4	1	Create, Update Status, Delete Query and Save			1	1	1		1	1	25	1397
Make Inactive	Input	1		update				1					5	253
Manage	Input	2	1	receive, process and store data			1				1	1	16	891
Measure	Report	1		Process Data for Report	possible save					1		0	5	253
Modify	Input	1		update							1		5	253
Outputting	Report	1		Process Data for Report	possible save					1		0	5	253
Pick List	Query	1		query/read							1		5	253
Provenance	Input	3	1	Create, Update Status, Query and Save			1	1			1	1	21	1144
Provide	Output	1		Report						1			5	253
Purge	Input	1		Delete					1				5	253
Report	Report	1		Process Data for Report	possible save					1		0	5	253
Smart Data Tagging	Input	2	1	receive, process and store data			1				1	1	16	891
Store	Input	2	1	receive, process and store data			1				1	1	16	891
Tabulate	Report	1		Process Data for Report	possible save					1		0	5	253
Track	Output	1		Report						1			5	253
View	Query	1		query/read							1		5	253

NSA UCC output

File Path	File Type	Physical SLOC	Logical Sloc	Comment Lines	Blank Lines	Total Lines	Suspected GOTS, COTS or FOSS	Suspected Auto Generated (Greater than 1000 SLOC, less than 5 comments)	Test	Duplicate File Name-Conc	Duplicate Search	Row Number	Suspected C/G/F Yes - No	Developed vs. N/A
C:\Boston|Fenway|World Champions\javascript	javascript	100	65	12	3	115						2		Developed
C:\Boston|Fenway|World Champions\javascript	javascript	100	65	12	3	115				World Champions\javascript	duplicate	3		N/A
C:\Boston|Fenway|World Champions\javascript	javascript					0						4		Developed
C:\Boston|Fenway|World Champions\java	java					0						5		Developed
C:\Boston|Fenway|World Champions\java	java					0						6		Developed
C:\Boston|Fenway|World Champions\java	java					0						7		Developed
C:\Boston|Fenway|World Champions\Test\java	java					0			test			8		N/A
C:\Boston|Fenway|World Champions\Test\java	java					0			test			9		N/A
C:\Boston|Fenway|World Champions\Test\java	java					0			test			10		N/A
C:\Boston|Fenway|World Champions\Nate	MongoDB					0	MongoDB					11		N/A
C:\Boston|Fenway|World Champions\Nate	MongoDB					0	MongoDB					12		N/A
C:\Boston|Fenway|World Champions\Nate	MongoDB					0	MongoDB					13		N/A

Perl shell reads file path and flags GOTS, COTS or FOSS, Identifies duplicate files, test code and flags code as developed or not developed.

Analyst filters code > 1000 Logical SLOC and Comments < 5, and manually flags it as auto generated. (Based on NSA data analysis)

UCC Report

Developed vs. N/A	(All)

Row Labels	Sum of Physical SLOC	Sum of Logical Sloc	Sum of Comment Lines	Sum of Blank Lines	Sum of Total Lines
java	1090	908	24	16	1130
javascript	255	157	27	7	289
MongoDB	719	545	10	13	742
Grand Total	**2064**	**1610**	**61**	**36**	**2161**

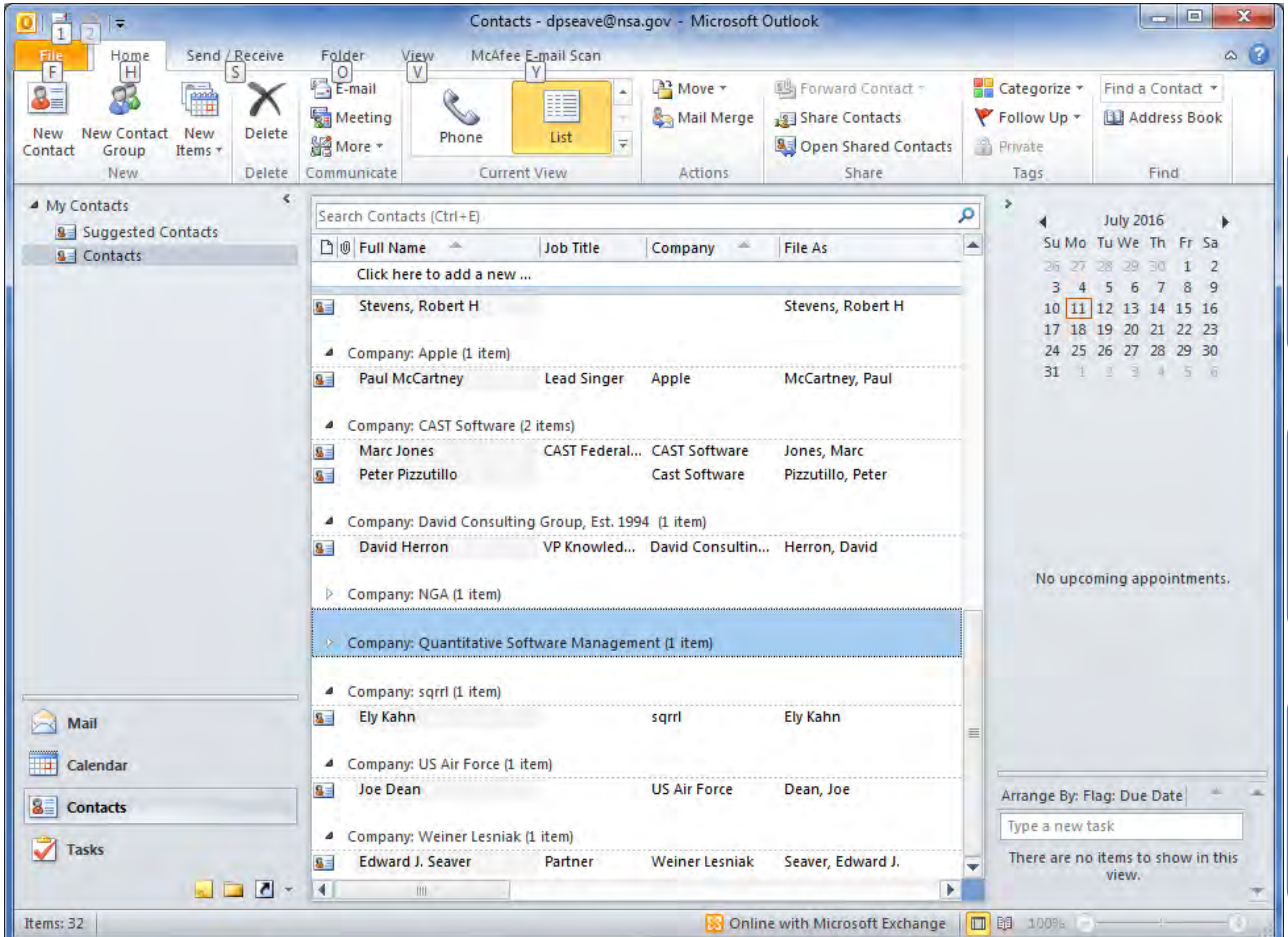
Developed vs. N/A	Developed

Row Labels	Sum of Physical SLOC	Sum of Logical Sloc	Sum of Comment Lines	Sum of Blank Lines	Sum of Total Lines
java	887	755	15	7	909
javascript	155	92	15	4	174
Grand Total	**1042**	**847**	**30**	**11**	**1083**

Developed vs. N/A	N/A

Row Labels	Sum of Physical SLOC	Sum of Logical Sloc	Sum of Comment Lines	Sum of Blank Lines	Sum of Total Lines
java	203	153	9	9	221
javascript	100	65	12	3	115
MongoDB	719	545	10	13	742
Grand Total	**1022**	**763**	**31**	**25**	**1078**

Query Transaction



A Query lets you retrieve data

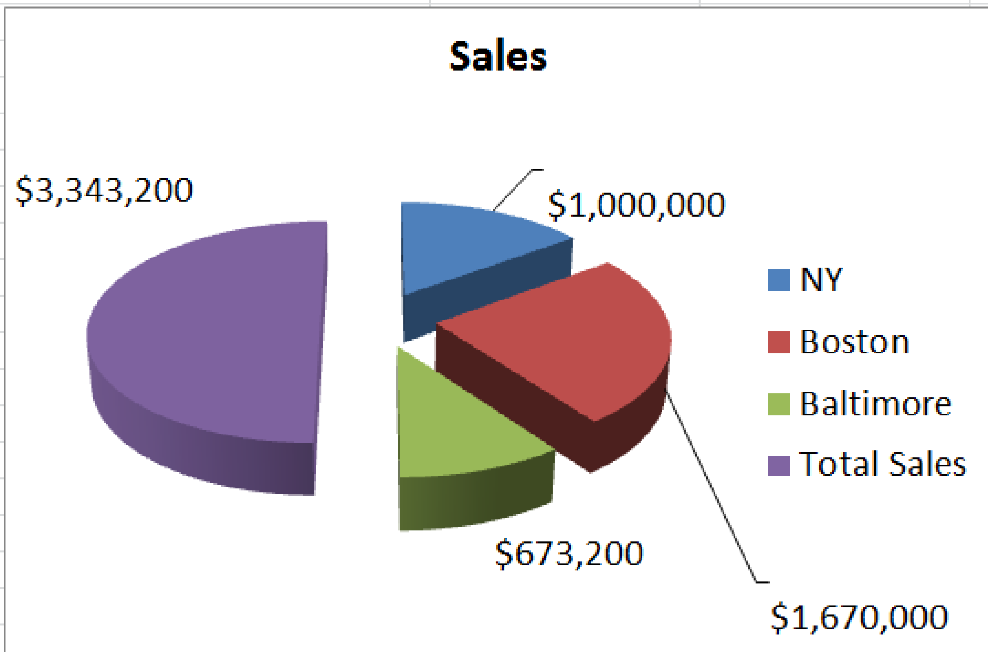
This example has 4 search variables

1 transaction

A Query retrieves and displays information already in the system, or another system

Report Transaction

Month to date sales of PEEPS Candies Aug-16				
Distribution Source	NY	Boston	Baltimore	Total Sales
Sales	$ 1,000,000	$ 1,670,000	$ 673,200	$ 3,343,200
% of total Sales	30%	50%	20%	



- A report displays or transmits data that is processed when the report runs. Typically this is some kind of mathematical processing
- This example would be 2 reports one for the table one for the chart
- 7 data elements

Function Points +/- for NSA

- Pros:
  - Consolidated by several decades of use
  - Many benchmarks available (ISBGS)
  - Detailed documentation
  - Training
  - Certification
- Cons
  - Time consuming
  - High level of detail required for IFPUG count
  - The wealth of rules are not always easily applicable
  - Limited DoD or US Government data

Count before you Estimate

- How many Jelly Beans in the Jar?
- Closest guess gets a candy bar?

NATIONAL RECONNAISSANCE OFFICE

Software Development Agility Scale

Aaron Gregory, Michelle Jones, Geoff Pierce, Ryan Timm, Jenny Murrill

August 2016

SUPRA ET ULTRA

Agenda

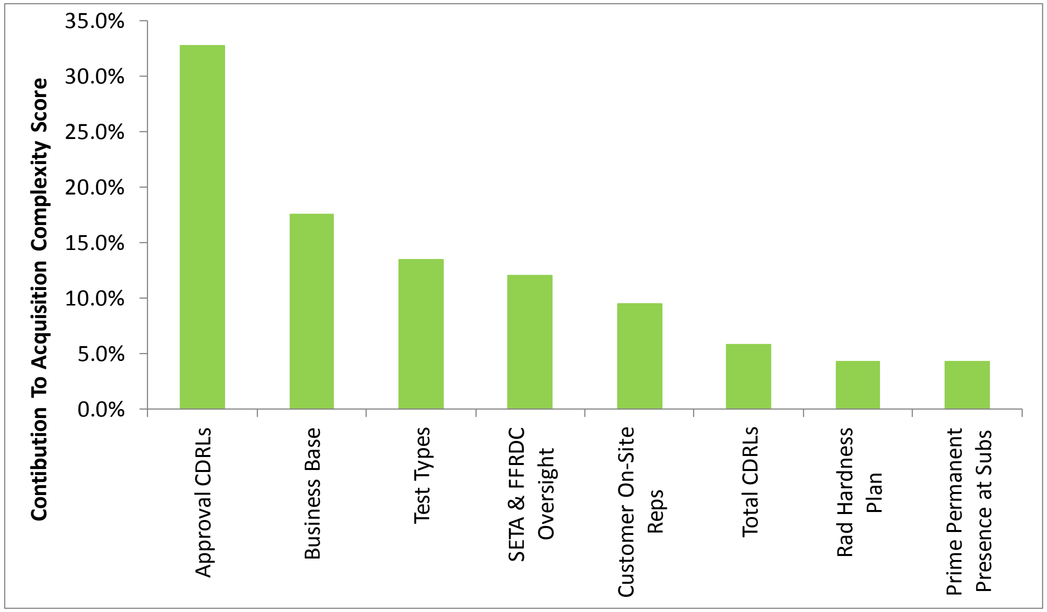
- Informative References
- Objectives of our Study
- Our Survey
- Our Process for Analyzing Results
- Cross-Agency Participation
- Discussion

Background and Objectives

- The NRO CAAG is encountering more and more programs using agile software methodology, or Scaled Agile Framework (SAFe) in software development
- Government program managers can benefit from the software development team's responsiveness to change and increased insight into status and backlog compared to programs employing waterfall development methodology
- Programs using agile claim the potential for lower cost: reduce integration cost due to continuous testing, reduce sustainment cost because defects are corrected sooner. These claims are hard to quantify or substantiate, and often involve an up-front investment in automated testing or other infrastructure.
- Agile sizing metrics are not standard: the amount of work that equates to a "story point" varies by company, by project, and by team.

The objective of this research project is to better understand business practices associated with Agile, so we can improve cost estimates and Integrated baseline review support for programs applying agile

Commercial-Like Acquisitions: Practices and Costs
2013 Updated Model



$$\text{Adj. Factor} = 0.11 + 0.63 \times (\text{AC Score})$$

Parameter x_i	Weight (W_i)	x_{min}	x_{max}
# Approval CDRLs	32.8%	0	75
Plant Business Base at ATP	17.5%	Varies By Plant	Varies By Plant
Types of Testing	13.5%	6	10
3rd Party Oversight Types	12.1%	0	2
# Customer On-Site Reps	9.5%	0	35
Total # CDRLs	5.9%	15	175
Rad Hardness Assurance Plan (y/n)	4.3%	0	1
Prime Presence Permanent on Subcontractors' Sites (y/n)	4.3%	0	1

The CAPs model identifies commercial practices which reduce costs and provides an adjustment factor for estimating the associated savings.
Could this approach apply to estimating Agile Software Development efforts?

Informative References

Measuring Agility in Software Development. Dan Houston & Steve Rosemergy. Aerospace Corporation, Software Acquisition and Modeling Department

A Model for Estimating Agile Project Process and Schedule Acceleration. Dan Ingold, Barry Boehm, Supannika Koolmanojwong Center for Systems and Software Engineering University of Southern California

Commercial-Like Acquisitions: Practices and Costs. Cost Analysis Improvement Group, National Reconnaissance Office

Defense Agile Acquisition Guide, Tailoring DoD IT Acquisition Program Structures and Processes to Rapidly Deliver Capabilities. Pete Modigliani and Us Chang.

Agile Metrics: Progress Monitoring of Agile Contractors. Will Hayes, Suzanne Miller, Mary Ann Lapham, Eileen Wrubel, Timothy Chick. Software Engineering Institute

A Primer on Agile Software Development for Cost Analysts. Qualis Corporation

Maturing the Economics of Agile Development. Jennifer Manring. MITRE.

Our approach is to learn from current research efforts

Maturing the Economics of Agile Development

Cost Increases / Cost Decreases

Life Cycle Cost Element	Cost Increases	Cost Decreases
PM/SE	* Start-up learning inefficiencies * Some new roles, such as Product Owner * Higher level of engagement for the Government	None
Software Development	* Higher labor rates * Additional unplanned rework during development	* Less unplanned work after development * Higher productivity * Less software growth due to requirements uncertainty
Integration and Test	* Iniital costs to set-up automated testing * Continuous integration and testing performed * Regression testing on each release	* Less complex, smaller batch testing * High use of automated testing * Earlier detection of software defects
Deployment	* Initial costs to develop deployment scripts	* High use of automated deployment and DevOps
Training	* More frequent training deliveries	* Greater degree of built-in usability
Sustainment	* Some increases from above elements continue	* Reduction in # of software defects post-deployment * Reduction in enhancement post-deployment

MITRE

Life Cycle Cost Element	Cost Impact Range	
	Best Case	Worst Case
Program Management/ System Engineering	=	+
Software Development	-	=
Integration and Test	=	+
Fielding/Deployment	=	++
Training	+	++
Sustainment	--	-

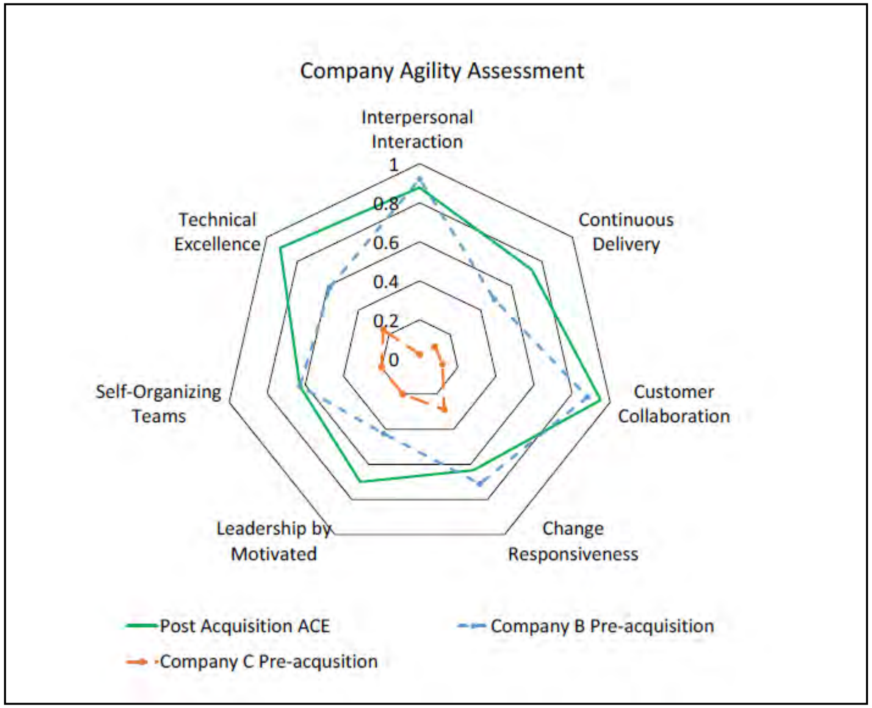
++ significant increase, + increase, = no impact, – decrease, – – significant decrease

The paper identifies potential for cost impact of using agile, and concludes the largest area of potential decreased cost lies in sustainment

Measuring Agility in Software Development

General Agile Characteristics

- Interpersonal interaction
- Working product or service
- Customer/user collaboration
- Responsiveness to change
- Continual delivery of customer value
- Self-organizing, multifunctional collaboration
- Leadership by the motivated
- Technical excellence and simplicity



Similarity: Measuring Agility model surveys projects to measure agility to help programs become more agile

Difference: Our study has a neutral perspective and does not advocate Agile over Plan Driven

A Model for Estimating Agile Project Process and Schedule Acceleration

- Effort is constant and duration can be impacted by a series of factors relative to agility

Category	Factor in Table 1 (Fi)
Product	Simplicity
	Element Reuse
	Low-Priority Deferrals
	Models vs Documentation
	Key Technology Maturity
Process Factors	Concurrent Operational Concept Requirements, Architecture, V&V
	Process Streamlining
	General SE tool support CIM (Coverage, Integration, Maturity)
Project Factors	Project size (peak # of personnel)
	Collaboration support
	Single-domain MMPTS (Models, Methods, Processes, Tools
People Factors	General SE KSAs (Knowledge, Skills, Agility)
	Single Domain KSAs
	Multi-Domain KSAs
	Team Compatibility
Risk Acceptance Factor	

$$D = \prod F_i \sqrt{PM}.$$

D = Estimated Duration
F_i = Scored Factors (see table)
PM = Person Months

Factor	Potential Schedule Increase (longer)	Potential Schedule Decrease (shorter)
Product	1.09	0.87
Process	1.09	0.87
Project	1.08	0.9
People	1.13	0.84
Risk	1.13	0.84

Similarity: Measuring Agility to estimate an impact to schedule, clearly articulating the factors that make a difference
Difference: Their assumption is that effort is constant, although the schedule can be compressed

Objective of the Software Development Questionnaire

<u>Objectives</u>

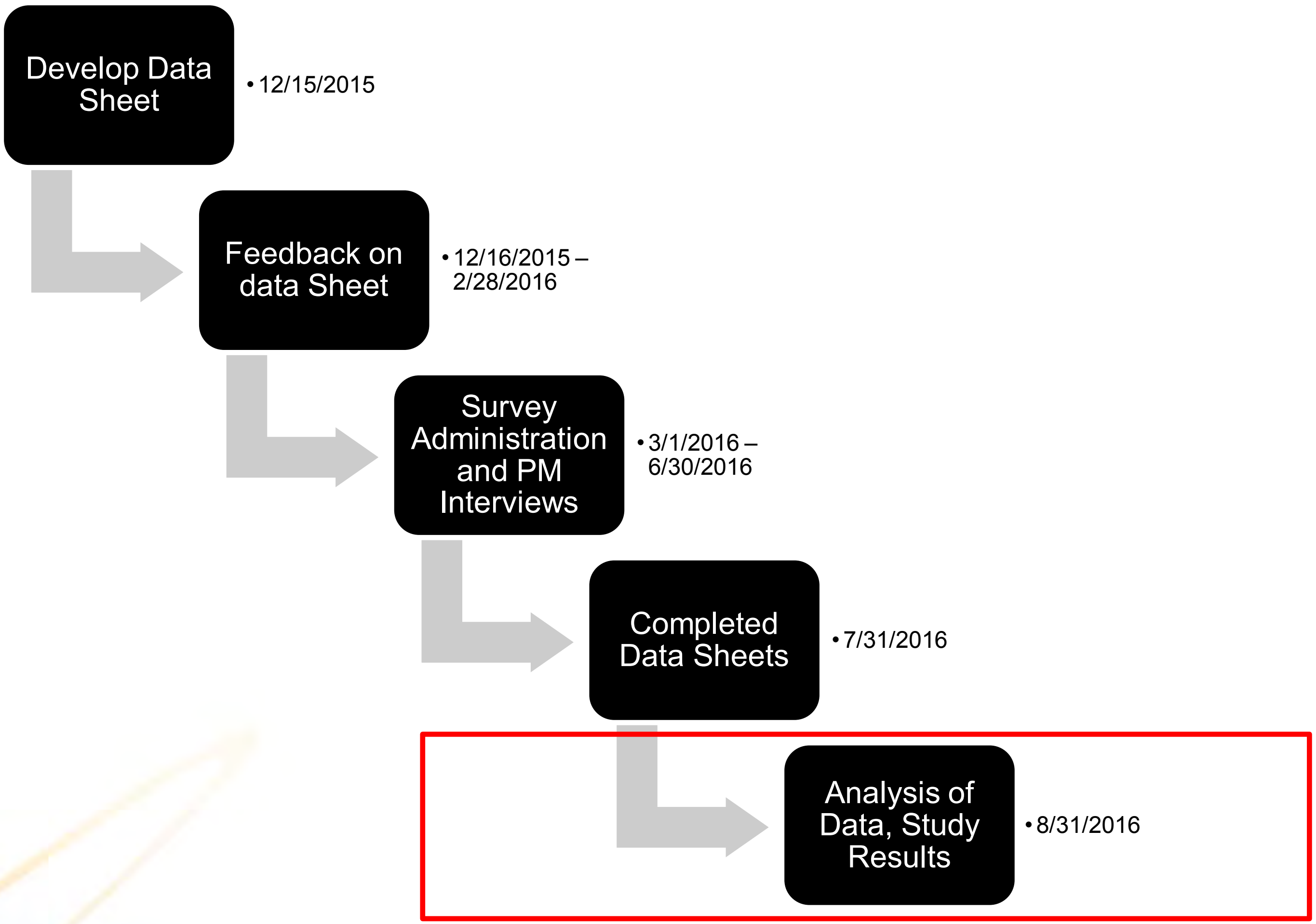
- Develop criteria to place software development programs on a continuum from plan-based to agile
- Identify differing characteristics of software development methodology, testing protocol, and customer interaction associated with agile software methodology
- Quantify potential cost impacts associated with a program's agility
  - Total Cost
  - Cost Profile
  - SEITPM vs Development vs SW Test

<u>Applications</u>

- Define "agile" based on software development business practices rather than assertions
- Begin to investigate/quantity potential savings associated with agile methodologies (if any) to better estimate costs
- Better informed IBR support.

In coordination with other space, defense and intelligence agencies, we developed and distributed a software development questionnaire

Agile Acquisition Model
Project Timeline



Survey Content

Software Development Process Survey (17 Feb 16) – SCORING GUIDE (90 possible points)

INSTRUCTIONS: Please complete this survey for development efforts only (not maintenance efforts) that fall under your purview. This survey is geared towards contractor technical managers with knowledge of the software development team's daily work. If you have questions, please contact Michal Bohn (571-304-8890) or Michelle Jones (571-304-8861) in the NRO CAAG. Thank you for your time and insights!

CONTACT INFORMATION

Preparer Name ____________________ Version ____________
Preparer Position ____________________ Date ____________
SW Development Team Name ____________ Phone ____________
Program Name ____________________

Software Size (of development portion that is addressed in the survey response) (all non-scored)

1. What is the EAC of the software? ______ $K
2. What is the software size of the development effort, inclusive of Prime and Subs? (identify units) ______ Logical SLOC or Function Pts
3. What percentage of the software is reused code? ______ %
4. What percent of the effort is software development (and associated SEITPM)? ______ %
5. How long is the software development phase? ______ months
6. What is the total average yearly staffing? ______ FTEs
7. How many software development teams on the effort? ______
8. What is the primary software development process used?
   a. Agile c. Spiral
   b. Waterfall d. Other: ____________

Acquisition and Development Process

9. Who provides input into HOW requirements are met? Select all that apply. (1.5 pt for each selected points)
   a. Customer c. Contractor
   b. Product Owner d. Program Office
10. What is required to change the scope of work on the contract?
    a. Customer involvement in prioritization of the requirements/Change Requests (CR) (6 pts)
    b. Informal written/verbal agreement between Government and Contractor (3 pts)
    c. An ECP that undergoes an official approval process and is followed by a Contract Mod (0 pt
11. Select the choice that best describes how systems integration takes place:
    a. A single team develops all software, conducts testing, and performs systems integration (6
    b. A single team develops all software and conducts testing and then hands it off to another t performs systems integration (4 pts)
    c. Separate teams develop software, conduct testing, and perform systems integration (2 pts)
    d. Separate teams develop individual software modules, which are handed off to other teams integration and testing (0 pts)
12. Who reprioritizes the requirements throughout the contract (select all that apply)?
    a. User (6 pts) c. Development Team (0 pts)
    b. Customer/Program Office/Bill Payer (3 pts)
13. What is the average duration of a development or release cycle (major or minor releases)?
    a. Less than 4 months (6 pts) c. Greater than 12 months (0 pts)
    b. 4 to 12 months (4 pts)

1

14. Which best describes the frequency of internal development planning? (more often=agile)
    a. Daily (6 pts) c. Monthly (4.5) e. Annually (2)
    b. Weekly (5.5) d. Quarterly (3) f. Intervals exceeding 12 months (0)
15. Who participates in development planning? Select all that apply. (1.2 pt for each checked, max 6 pts)
    a. SW Developers c. Customer e. Integrators
    b. Testers d. Management f. Other: ____________
16. What best describes the frequency of Program Office (to include USG, SETA, FFRDC) participation in development planning? (more often=agile)
    a. Daily (6 pts) c. Monthly (4.5) e. Annually (2)
    b. Weekly (5.5) d. Quarterly (3) f. Intervals exceeding 12 months (0)

Communication (more often=agile)

17. How frequently do/does the team(s) meet to discuss issues?
    a. Daily (6 pts) c. Monthly (4.5) e. Annually (2)
    b. Weekly (5.5) d. Quarterly (3) f. Intervals exceeding 12 months (0)
18. How often does the Program Office (to include USG, SETA, and FFRDC) provide feedback during the development process?
    a. Daily (6 pts) c. Monthly (4.5) e. Annually (2)
    b. Weekly (5.5) d. Quarterly (3) f. Intervals exceeding 12 months (0)
19. How frequently does the development team meet with users?
    a. Daily (6 pts) c. Monthly (4.5) e. Annually (2)
    b. Weekly (5.5) d. Quarterly (3) f. Intervals exceeding 12 months (0)
20. How frequently does the development team demo the software to users or customers?
    a. Daily (6 pts) c. Monthly (4.5) e. Annually (2)
    b. Weekly (5.5) d. Quarterly (3) f. Intervals exceeding 12 months (0)
21. Which best describes the frequency of review sessions?
    a. Daily (6 pts) c. Monthly (4.5) e. Annually (2)
    b. Weekly (5.5) d. Quarterly (3) f. Intervals exceeding 12 months (0)
22. In addition to monthly CDRLs (cost/schedule/performance reporting), which artifacts are rou for each capability delivery or completed work package (whether demonstrated or delivered <Demographic, not scored>
    a. User Guides d. Test Plans/Results g. Known issues
    b. Privileges User Guides e. Program Reviews h. Sizing data
    c. Design documentation f. Version description document i. Other:________

Testing

23. How often is integration testing performed? <Demographic, not scored>
    a. Daily c. Monthly e. Annually
    b. Weekly d. Quarterly f. Intervals exceeding 12 months
24. How often is acceptance testing performed? <Demographic, not scored>
    a. Daily c. Monthly e. Annually
    b. Weekly d. Quarterly f. Intervals exceeding 12 months
25. How regularly are the tested baseline and the developmental baseline synced? (more=agile)
    a. Daily (6 pts) c. Monthly (4) e. Annually (1.5)
    b. Weekly (5.5) d. Quarterly (3.5) f. Intervals exceeding 12 months (0.5)
26. How is "sell-off" defined? (select all that apply)
    a. Demonstration for users (6 pts) c. User acceptance testing (execution of formal test plans) (2 pts)
    b. Demonstration for "bill payer" (4) d. Transition to operations (0 pts)

2

Demographics

Software Development Process

Communication

Testing

Scoring Approach

Primary Method

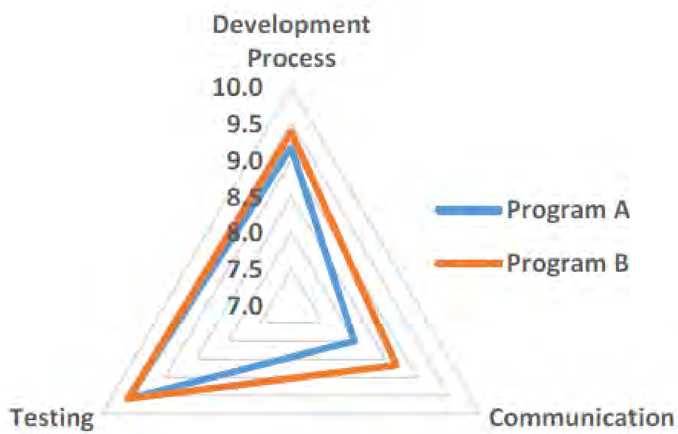
- Survey has 26 questions
  - 15 are scored, 11 are for demographic purposes
- Each question has equal weighting
  - May assign varied weightings in future analysis
- Each question is scored on a scale of 0 to 6 points
  - Distribution of points within answer choices varies across questions

Illustration by Program

- Points were totaled for three areas: Acquisition and Development Process, Communication and Testing
  - Maximum score is 90 points
- Points were normalized to 10 and charted on a spider graph

Preliminary

Scoring Example



Program A	Max Points	Raw Score	Normalized Score*
Development Process	48	44	9.2
Communication	30	24	8.0
Testing	12	11.5	9.6
Total	90	79.5	26.75

Program B	Max Points	Raw Score	Normalized Score*
Development Process	48	45	9.4
Communication	30	26	8.7
Testing	12	11.5	9.6
Total	90	82.5	27.625

*Normalized Score is based on 10 possible points

Preliminary

Discussion

NATIONAL RECONNAISSANCE OFFICE

SUPRA ET ULTRA

Adventures in Collecting, Evaluating, and Analyzing Army System Data to Objectively Estimate Software Maintenance Costs

Deputy Assistant Secretary of the Army for Cost and Economics (DASA-CE)

24 August 2016

U.S. Army Software Maintenance Initiative

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

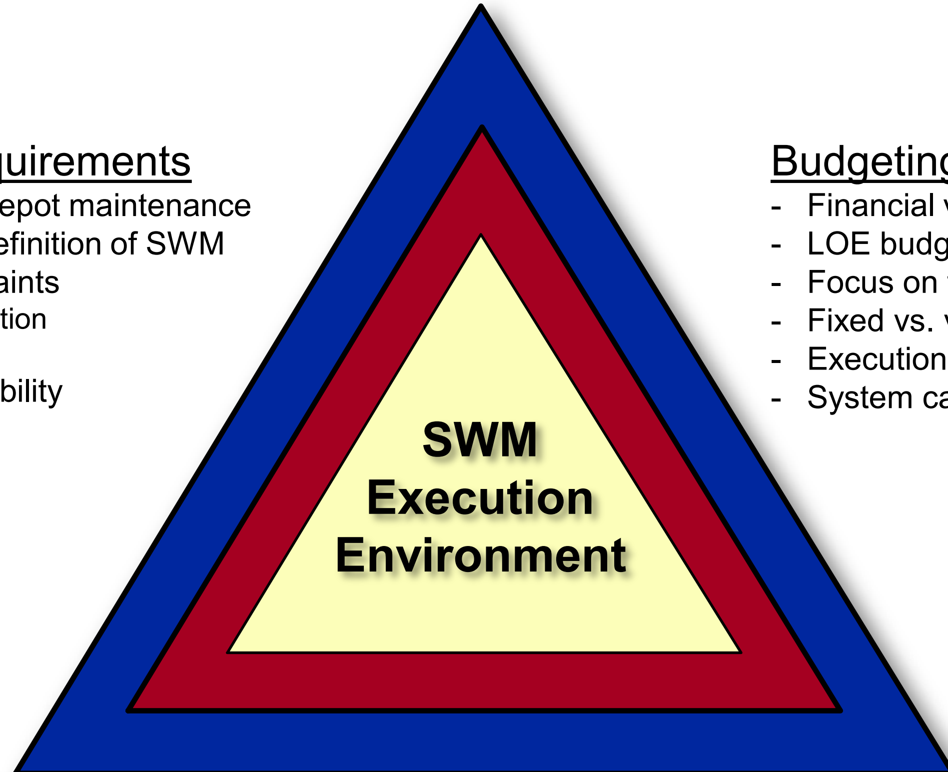
Our Immediate Focus

- Collection and evaluation of correlated system SWM cost and technical execution data
  - All Army operational systems
  - System and release level
  - Enterprise data call(s) - Phase I and Phase II
- Generation and validation of effective CERs
  - System software functional domains
  - Army software acquisition and sustainment organizations
  - Government and contractor costs
  - Variable and fixed (core) SWM costs
- Instantiation of systemic Army SWM data collection and analysis processes
  - Valid data collection requirements
  - Adaptation of existing financial and technical ERP systems
  - Populated cost and technical data repository
- Addressing Army SWM policy, business, and technical inconsistencies
  - Legislation, regulation, and policy constraints
  - Financial business processes
  - Software engineering life cycle change process(es)

Army Software Maintenance Conflicts

Statute - Policy Requirements

- SWM a component of depot maintenance
- Antiquated "waterfall" definition of SWM
- Colors of money constraints
    - accounting separation
    - usage restrictions
- Organizational responsibility
- Governance variations
- Contracting



Budgeting-Accounting Process

- Financial vs. Cost accounting constructs
- LOE budgeting/obligation model
- Focus on funding - not execution
- Fixed vs. variable costs
- Execution volatility
- System capabilities/limitations

Software Engineering - Change Integration Process

- Incremental release process - development and sustainment
- Continuous system software life cycle development and update
- "Pre-Planned" technical (mission) debt - deferred functionality
- Financial and technical "spillover" - development to sustainment
- Dedicated system software teams - all life cycle phases
- Emergent software change volume, technical, and management drivers

Overarching Result

There is no direct accounting that relates the dollars expended with the software maintenance products delivered to the warfighter

Pertinent Discussion Points

- How do we define Army Software Maintenance?
  - What's In - What's Out?
  - What are our SWM output products? Changes? Releases? Capabilities?
  - What types of dollars pay for what?
  - Where/how are the dollars executed?
  - Maintenance vs. Sustainment?

- What is the utility of the data we are currently collecting with respect to cost estimation?
  - Satisfaction of system and enterprise stakeholder information requirements?
  - Data characterization and evaluation?
  - Ability to generate valid CERs?
  - How do we improve?

- Can we really expect to implement the changes that we need to effectively manage the operational software change process ?
  - Cost accounting rather than financial accounting
  - Cost-to-product allocations in an environment defined by LOE constraints
  - Movement towards data informed - rather than arbitrary - resourcing decisions
  - Execution focus rather than planning focus
  - Software change "portfolio" management within the enterprise
  - Linking dollars to mission capability

DASA-CE SWM Definition

Software Maintenance

- Software maintenance includes all software change activities and products associated with modifying a software system after EMD has completed and a software release has been provided to an external party
- The release is the primary SWM change product - a composite of one or more changes - it can be either a formal release or an engineering release
- SWM includes software enhancements and software corrections/adaptations
- SWM includes activities and change products funded by multiple funding sources (RTDE, Production, OMA, FMS, OCO, etc.)
- Fixed and Variable costs accrued at both the system and organizational levels by both organic and contractor resources.
- Software maintenance and software sustainment are considered to be synonymous.

Army Software Maintenance WBS

Software Maintenance

System Specific

1.0 Software Change Product

Change requirements
Change development
B/L Integration & Test
IV&V

System Specific

2.0 Project Management

Planning
Execution management
Configuration management
Resource & team management
Contracting management
Measurement - reporting

System/Non-System Specific

3.0 Software Licenses

License - Right to Use
License - Maintenance
- COTS
- NDI
- Other

System Specific

4.0 Certification & Accreditation

Security
Safety
Networthiness
Airworthiness

System/Non-System Specific

5.0 System Facilities

Hardware
- Software development assets/workstations
- System integration & test facilities
- Test equipment - tools

Facility Operations

System Specific

6.0 Sustaining Engineering

Engineering Support
- Test Support
- Software Delivery
- Technical Studies

User Support
- Help desk
- Training

System Specific

7.0 Field Software Eng.

On-Site technical assistance
Problem Troubleshooting
S/W Installation
Operational Assistance
On-site Training

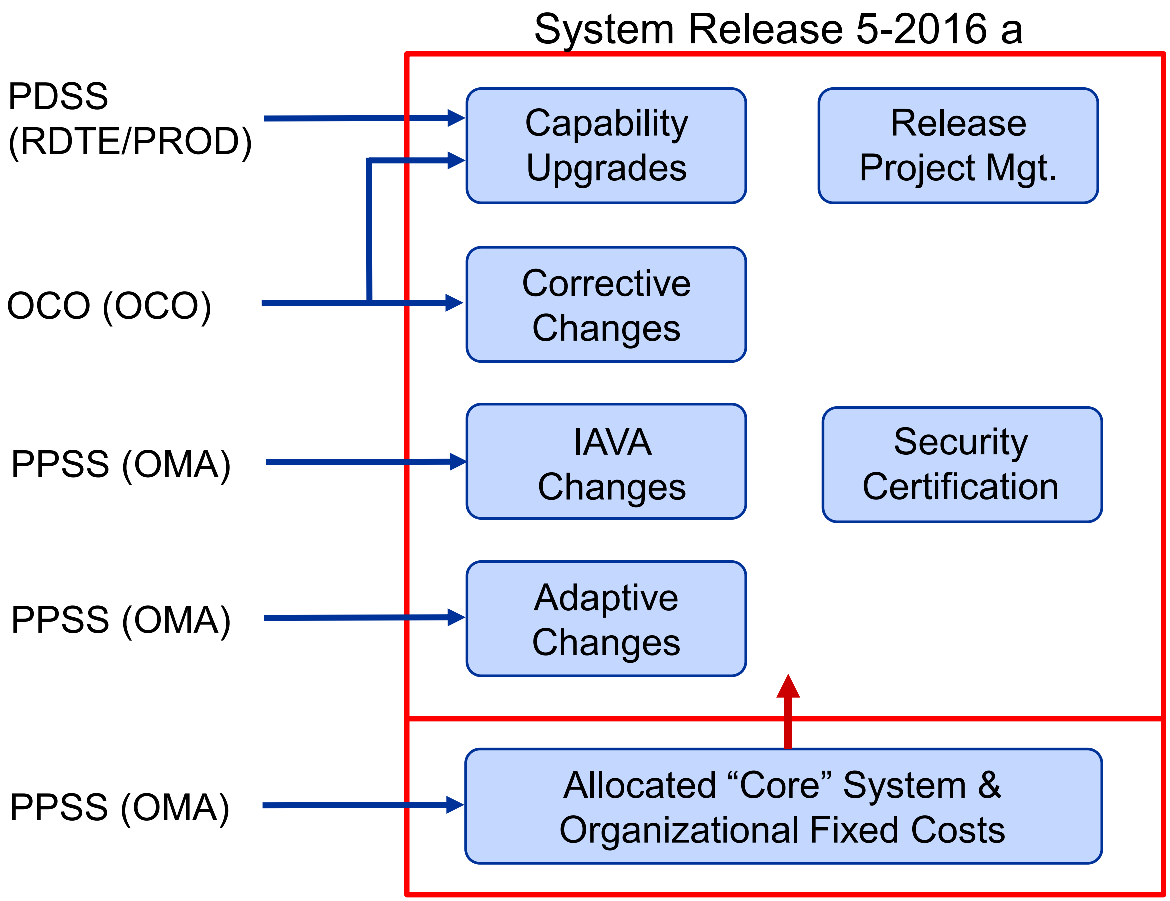
Non-System Specific

8.0 Operational Management

Operations
Organization management
Personnel management
Financial management
Information management
Process management
Change management

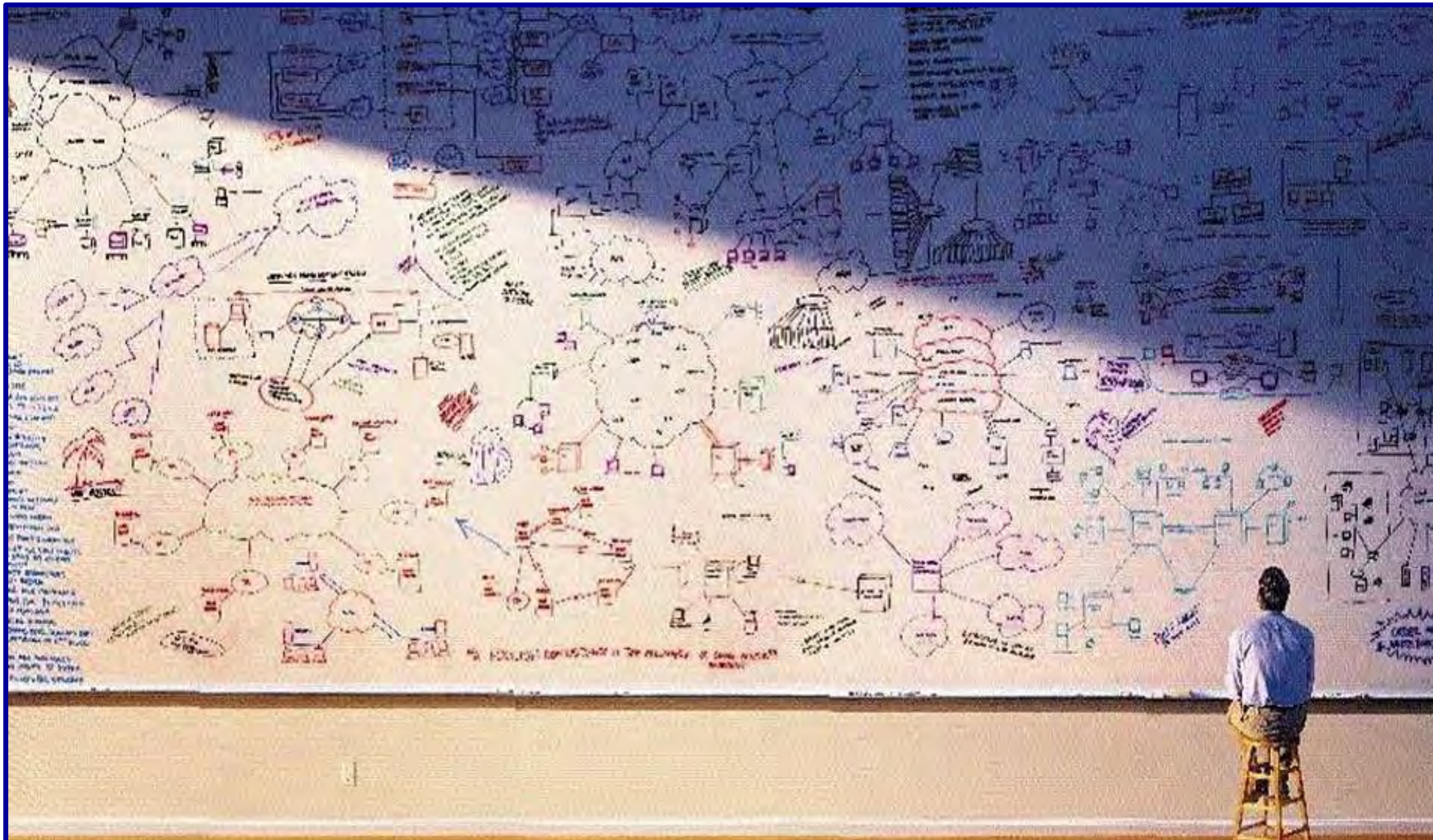
Version 4.4d

Notional Cost - Army SWM Change Product



Multiple appropriations funding different types of changes integrated into the same release
Each appropriation requiring a separate accounting path

Army Software Maintenance Data



Software Maintenance Data Call

- **Purpose** - Collect software maintenance data to inform cost models, data availability, data requirements, and to initiate the systemic collection of software cost and correlated technical measures.
- **Phase I Status**
  - Received and verified data from 56 programs
  - Currently developing cost models and cost estimating relationships (CERs) (Nov 2016)
- **Phase II Status**
  - Identified 205 systems for data collection
  - Developed phased plan for data collection and verification
  - Currently validating submissions received to date (65 programs)
- **Ongoing** – Update cost model and CERs on an incremental basis as additional data is received and verified

Data Requirements

- System Level Data
  - Context information
  - Organizations involved
- Annual effort /cost data (WBS elements #2 through #8, plus total annual)
- Release level data
  - Release context information
  - Operating environment
  - Application domain
  - Size data (those that apply)
    - Software requirements
    - External requirements
    - Source Lines of Code (SLOC)
    - Non-SLOC based size (e.g. RICE-FW, use cases, story points)
    - Software changes counts by priority (e.g. change requests, problem reports, defects)
    - IAVAs
  - Release effort / cost (WBS element #1)
  - Schedule - start and end dates
- Details on Software Licenses
  - Right to use and maintenance

Data Evaluation Factors

Data Evaluation

Availability

- Completeness of required data set - compliance to scope
- Underlying SWM business and technical processes are well enough defined to produce objective data on a periodic and/or event driven basis
- IT systems and tools exist to enable systematic and timely data collection
- Stakeholder cooperation

Integrity

- Data are derivatives of actual SWM technical and management processes
- All data (measures) are explicitly defined - measurement contexts are known
- Cost data is directly correlated with the WBS defined output products & activities
- Data is consistent - methods exist to address system conflicts (normalization)
- Parameter relationships are consistent

Usability

- Data is aligned with stakeholder decision information needs
- Data can be objectively characterized and interpreted
- Mapping and aggregation structures and methods exist to combine data
- Potential emerging information requirements have been considered

Phase 1 - Program Summary Data

			Initial System Overall			Detailed System Assessment						
PEO	**SEC**	**System**	**Total Program Effort/Cost**	**WBS 2-8**	**Project Mgmt (WBS-2)**	**License Management (WBS-3)**	**C&A Support (WBS-4)**	**System Facilities Management (WBS-5)**	**Sustaining Engineering (WBS-6)**	**Field S/W Engineering (WBS-7)**	**Operational Management (WBS-8)**	**License Costs**
PEO 1	SEC 1	System 1	G	Y	R	G	G	Y	Y	N/A	Y	R
PEO 2	SEC 1	System 2	G	G	G	G	G	G	G	N/A	G	G
PEO 2	SEC 1	System 3	G	G	G	G	G	G	G	N/A	G	G
PEO 2	SEC 1	System 4	R	R	R	R	R	R	R	R	R	G
PEO 3	SEC 4	System 5	G	Y	Y	Y	Y	Y	Y	N/A	R	G
PEO 3	SEC 4	System 6	G	Y	Y	R	Y	Y	Y	N/A	R	G
PEO 3	SEC 4	System 7	G	G	G	N/A	G	G	G	N/A	R	N/A
PEO 6	SEC 4	System 8	G	G	G	G	G	G	G	N/A	G	G
PEO 5	SEC 4	System 9	G	Y	R	R	N/A	G	O	N/A	R	G
PEO 4	SEC 2	System 10	Y	Y	R	R	O	O	R	N/A	R	G
PEO 1	SEC 1	System 11	G	Y	R	R	G	Y	G	G	G	R
PEO 1	SEC 1	System 12	G	G	G	G	G	G	G	G	G	G
PEO 8	SEC 2	System 12	G	O	O	R	Y	Y	R	G	R	R
PEO 8	SEC 2	System 13	G	Y	G	G	N/A	R	G	N/A	G	G
PEO 8	SEC 2	System 14	G	N/A	R	N/A	Y	N/A	N/A	N/A	N/A	N/A
PEO 8	SEC 2	System 15	G	G	G	N/A	G	G	G	N/A	G	O
PEO 7	SEC 3	System 16	Y	Y	Y	R	R	Y	R	N/A	Y	G
PEO 7	SEC 3	System 17	G	G	G	G	G	G	G	N/A	G	G
PEO 7	SEC 3	System 18	G	G	G	G	G	G	G	N/A	G	G
PEO 7	SEC 3	System 19	Y	G	Y	N/A	Y	N/A	Y	Y	N/A	G
PEO 6	SEC 1	System 20	G	Y	G	G	G	G	Y	N/A	Y	G

Evaluation of Phase 1 Data System Summary Data

- Data was provided for 56 systems
- System data was provided as follows:
  - Total system SWM effort/cost: 42 systems
  - Certification and accreditation cost: 35 systems
  - License costs: 41 systems

			Initial System Overall				Detailed System Assessment						
			Total Program Effort/Cost	WBS 2-8		Project Mgmt (WBS-2)	License Management (WBS-3)	C&A Support (WBS-4)	System Facilities Management (WBS-5)	Sustaining Engineering (WBS-6)	Field S/W Engineering (WBS-7)	Operational Management (WBS-8)	License Costs
Counts for Data Provided		R	13	11		28	30	10	20	26	17	32	12
		O	1	15		2	2	8	6	3	2	1	3
		Y	12	17		8	4	15	13	10	8	10	1
		G	30	12		18	12	20	15	16	6	11	35
		N/A	0	1		0	8	3	2	1	23	2	5
		Total	56	56		56	56	56	56	56	56	56	56

Phase 1 - Release Data

				Initial Release Overall		Detailed Release Assessment							
PEO	SEC	System	Release	CER Usability	SER Usability	Size: Requirements	Size: External Interfaces	Size: SLOC	Size: Non-SLOC	Size: SW Changes	IAVAs	Effort (WBS-1)	Schedule (WBS-1&2)
PEO 1	SEC 1	System 1	V1.0	G	G	G	N/A	G	N/A	G	N/A	G	G
PEO 1	SEC 1	System 2	V1.0	G	G	G	N/A	G	N/A	G	N/A	G	G
PEO 2	SEC 5	System 3	V1.0	Y	Y	G	G	G	N/A	G	G	Y	G
PEO 2	SEC 5	System 3	V2.0	Y	Y	G	G	G	N/A	G	G	Y	G
PEO 2	SEC 5	System 4	V1.0	Y	Y	G	N/A	G	N/A	G	N/A	Y	G
PEO 2	SEC 5	System 5	V1.0	G	G	R	R	Y	N/A	G	G	G	G
PEO 4	SEC 5	System 6	V1.0	Y	Y	G	N/A	G	N/A	G	N/A	Y	G
PEO 4	SEC 5	System 6	V2.0	Y	Y	G	N/A	G	N/A	G	N/A	Y	G
PEO 4	SEC 5	System 6	V3.0	Y	Y	G	N/A	G	N/A	G	N/A	Y	G
PEO 4	SEC 5	System 6	V4.0	Y	Y	G	N/A	G	N/A	G	N/A	Y	G
PEO 3	SEC 5	System 7	V1.0	G	G	G	G	G	N/A	G	N/A	G	G
PEO 10	SEC 2	System 8	V1.0	Y	Y	G	N/A	G	N/A	G	G	Y	G
PEO 10	SEC 2	System 8	V2.0	Y	Y	G	N/A	G	N/A	G	G	Y	G
PEO 10	SEC 2	System 8	V3.0	Y	Y	G	N/A	G	N/A	G	G	Y	G
PEO 9	SEC 2	System 9	v1	O	O	G	R	R	R	G	G	Y	G
PEO 9	SEC 2	System 9	v2	O	O	N/A	N/A	N/A	N/A	N/A	G	O	Y
PEO 9	SEC 2	System 9	v3	O	O	N/A	N/A	N/A	N/A	N/A	G	O	Y
PEO 9	SEC 2	System 9	v4	O	O	N/A	N/A	N/A	N/A	N/A	G	O	Y
PEO 9	SEC 2	System 9	v5	O	O	N/A	N/A	N/A	N/A	N/A	G	O	Y
PEO 9	SEC 2	System 9	v6	O	O	N/A	N/A	N/A	N/A	N/A	G	O	Y
PEO 9	SEC 2	System 9	v7	O	O	N/A	N/A	N/A	N/A	N/A	G	O	Y
PEO 9	SEC 2	System 9	v8	O	O	N/A	N/A	N/A	N/A	N/A	G	O	Y
PEO 9	SEC 2	System 9	v9	O	O	N/A	N/A	N/A	N/A	N/A	G	O	Y
PEO 9	SEC 2	System 9	v10	O	O	N/A	N/A	N/A	N/A	N/A	G	O	Y
PEO 9	SEC 2	System 9	v11	O	O	N/A	N/A	N/A	N/A	N/A	G	O	Y
PEO 9	SEC 2	System 9	v12	O	O	N/A	N/A	N/A	N/A	N/A	G	O	Y
PEO 9	SEC 2	System 9	v13	O	O	N/A	N/A	N/A	N/A	N/A	G	O	Y
PEO 9	SEC 2	System 9	v14	O	O	N/A	N/A	N/A	N/A	N/A	G	O	Y
PEO 8	SEC 3	System 10	v1	G	G	G	Y	G	N/A	G	G	G	G
PEO 8	SEC 3	System 10	v2	G	G	G	Y	G	N/A	G	G	G	G
PEO 6	SEC 1	System 11	v1	G	G	G	Y	G	N/A	G	R	G	G
PEO 6	SEC 1	System 12	v1	G	G	G	G	G	N/A	G	G	G	G
PEO 7	SEC 2	System 13	v1	G	G	G	R	R	N/A	G	G	G	G
PEO 7	SEC 2	System 13	v2	G	G	G	R	R	N/A	G	G	G	G
PEO 7	SEC 2	System 13	v3	G	G	G	R	R	N/A	G	G	G	G
PEO 5	SEC 4	System 13	v4	G	G	N/A	N/A			G	G	G	G
PEO 5	SEC 4	System 13	v5	G	G	G	G			G	G	G	G
PEO 5	SEC 4	System 13	v6	G	G	G	G			G	G	G	G
PEO 5	SEC 4	System 13	v7	G	G	G	G	N/A	N/A	G	G	G	G
PEO 3	SEC 1	System 14	v1	G	G	G	G	G	N/A	G	N/A	G	G
PEO 3	SEC 3	System 15	V1	G	G	G	G	G	N/A	R	N/A	G	G
PEO 3	SEC 3	System 15	V2	G	G	G	G	G	N/A	R	N/A	G	G

Release Data

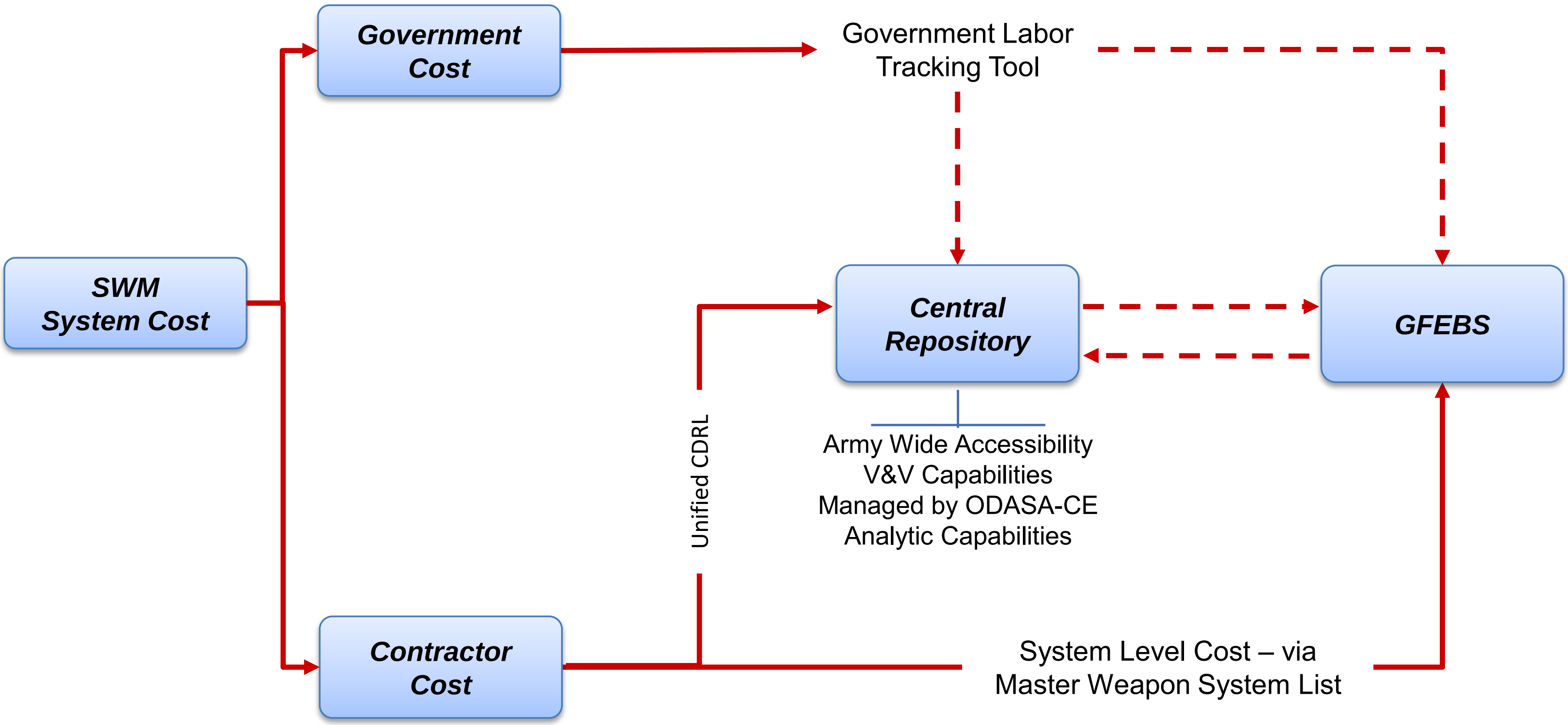
- Releases are the deliverable software maintenance change products - they incorporate one or more software changes
- Data was provided for 188 releases
- 154 releases could provide at least one size measure
  - 64 releases provided 3 or more size measures
- Data that was usable for creating Estimating Relationships were provided for:
  - Cost (CERs): 65 releases (actual/FTE effort), 46 releases (planning effort)
  - Schedule (SERs): 76 releases, 45 releases

	Initial Release Overall			Detailed Release Assessment							
	CER Usability	SER Usability		Size: Requirements	Size: External Interfaces	Size: SLOC	Size: Non-SLOC	Size: SW Changes	IAVAs	Effort (WBS-1)	Schedule (WBS-1&2)
R	76	66		66	46	41	28	35	46	59	27
O	46	45		7	2	7	0	3	3	49	14
Y	27	23		3	5	4	2	5	0	24	27
G	38	53		72	56	38	12	90	110	55	119
N/A	1	1		40	79	95	143	55	29	1	1
Total	188	188		188	188	185	185	188	188	188	188

SWM Data Collection Challenges

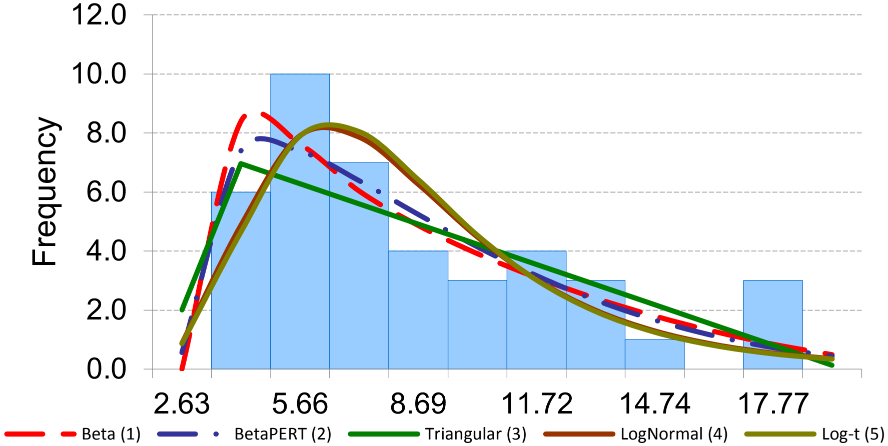
- Master system list
- Collection of both contractor and government data
  - Limited collection mechanisms today for contractor SWM data
  - Government labor
- Government doesn't always have detailed contractor data
- Availability and quality of data
  - Data in proper units of measurement
  - Data for all WBS elements (e.g. COTS, facilities, C&A)
  - Significant amount of rework/normalization required for analysis
- Wide scale implementation of data
  - Contractor AND government
  - Questionnaire? SRDR-M (revised)? CDRL? Level of automation
  - Ability to collect cost data that ties to SWM technical data
    - Cost CDRL - CEM WBS - SRDR-M - financial systems (Army GFEBS) - invoices

Notional Data Collection Process



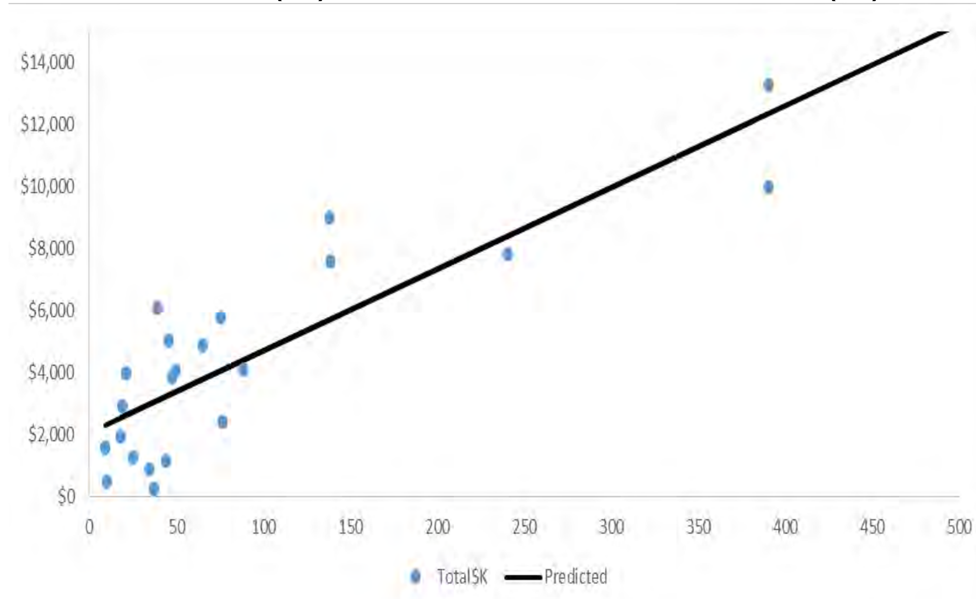
SWM Phase I CER

Total Cost per IAVA Distribution



LogNormal	Descriptive Statistics
Mean	7.8115
Std Dev	3.9809

ESLOC (K) vs Total Release Cost (K)



- Graph above (left) depicts the distribution of total cost per IAVA
- Graph above (right) shows the relationship between equivalent source lines of code and the total release cost
- This data facilitates informed decision making based on historical program data and can provide a realism check of future estimates
- Cost model will include use cases, directions for implementation, and will address all OMA categories

*CERs displayed are draft and should not be used for decision making

The Way Ahead

- Can we really expect to implement the changes that we need to effectively manage the operational software change process ?
  - Cost accounting rather than financial accounting
  - Cost-to-product allocations in an environment defined by LOE constraints
  - Movement towards data informed - rather than arbitrary - resourcing decisions
  - Execution focus rather than planning focus
  - Software change “portfolio” management within the enterprise
  - Linking dollars to mission capability

- Enablers
  - Acquisition focus on life cycle software sustainment cost projections - OSRs
  - Arbitrary and significant OMA funding cuts
  - Significant reduction in OCO funding
  - Emerging Army policy

- Strategy
  - Initiate the implementation of an Army software maintenance “information infrastructure”
  - Systemic system software maintenance data collection using existing ERP assets
  - Enterprise reporting of executed costs
  - Software maintenance data repository
  - Expanded set of SWM CERs - Updated estimation methodology
  - Service and DOD collaboration

Army Directive 2016-16
Changing Management Behavior: Every Dollar Counts

“The goal is to achieve the highest level of readiness given the resources provided”

“Be singularly focused on achieving the highest level of readiness with the greatest efficiency”

“ Avoid using budget execution data and obligation rates as the primary measure of fiscal success”

“Tie resource expenditures to outcomes”

Points of Contact

James Judy
Chief, Networks, Information, Software & Electronics Costing (NISEC) Division
U.S. Army Office of the Deputy Assistant Secretary of the Army for Cost & Economics (ODASA-CE)
(703) 697-1612
james.m.judy.civ@mail.mil

James Doswell
U.S. Army Office of the Deputy Assistant Secretary of the Army for Cost & Economics (ODASA-CE)
(703) 697-1572
james.p.doswell.civ@mail.mil

Cheryl Jones
U.S. Army RDECOM - ARDEC
(973) 724-2644
cheryl.l.jones128.civ@mail.mil

John McGarry
U.S. Army RDECOM - ARDEC
(973) 724-7007
john.j.mcgarry4.civ@mail.mil

Jenna Meyers
U.S. Army Office of the Deputy Assistant Secretary of the Army for Cost & Economics (ODASA-CE)
(703) 697-1645
jenna.e.meyers.civ@mail.mil

COSYSMO 3.0: The "Expert" Model

Jim Alstad and Dr. Barry Boehm

USC Center for Systems and Software Engineering

Software and IT CAST

Lockheed Martin Global Vision Center

August 24, 2016

Jim Alstad* 310/344-0894 jalstad@usc.edu

Acknowledgements

- **This presentation was adapted from one with these authors:**

Jim Alstad	USC Center for Systems and Software Engineering
Dr. Barry W Boehm	
Dr. Jo Ann Lane	
Mr. Garry Roedler	Lockheed Martin
Dr. Gan Wang	BAE Systems
Ms Marilee Wheaton	The Aerospace Corporation

COSYSMO 3.0 Objectives

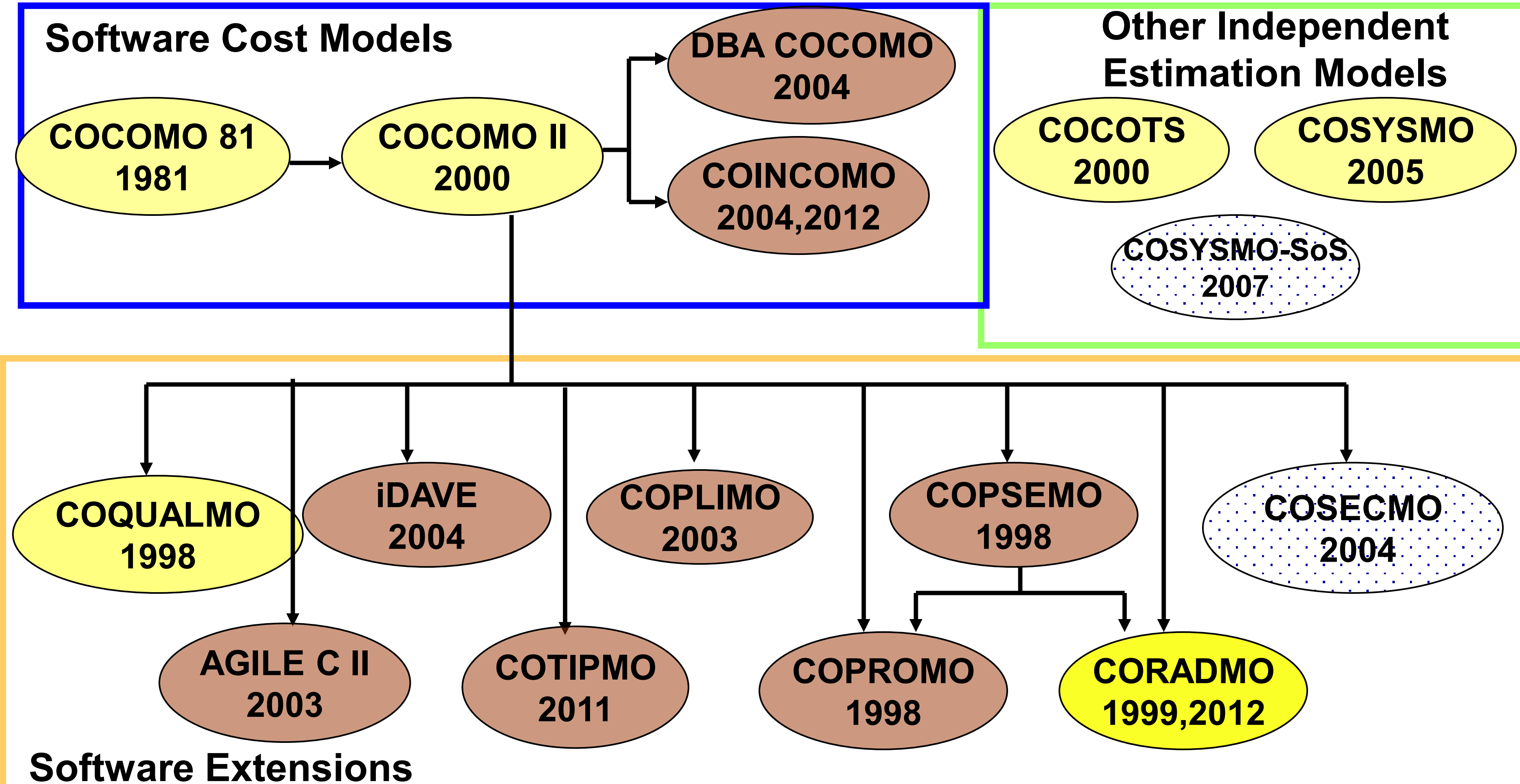
- **Context:**
  - Current and future trends create challenges for full-system cost estimation
    - Emergent requirements, rapid change, net-centric systems of systems, COTS, clouds, apps, widgets, high assurance with agility, multi-mission systems
  - Current development practices can minimize cost of one phase, such as development, while raising full-system cost
- **COSYSMO 3.0 is being developed to mitigate this situation by supporting accurate estimates of systems engineering costs, with benefits including:**
  - Allowing thoughtful system-level systems engineering during development, which can result in, for example, choosing new technologies that reduce total system cost
  - Allowing thoughtful engineering of systems to support life-cycle flexibility

Agenda

Agenda:

- **The motivation for COSYSMO 3.0**
- **History of COSYSMO 3.0**
- **Overview of the content of the COSYSMO 3.0 estimating model**
- **System-of-systems estimating: interoperability in COSYSMO 3.0**
- **Model status & plans**
- **Numerical values of COSYSMO 3.0 parameters**
- **Summary**

COCOMO Family of Cost Models



Legend:
Model has been calibrated with historical project data and expert (Delphi) data
Model is derived from COCOMO II
Model has been calibrated with expert (Delphi) data

Dates indicate the time that the first paper was published for the model

History of COSYSMO Models

COSYSMO 1.0
Valerdi, 2005

- Identifies form of model
- Identifies basic cost drivers
- Identifies Size measure

With Reuse
Wang et al, 2008

- Adds weights to Size elements, reducing net Size in the presence of reuse

Req'ts Volatile
Pena, 2012

- Adds scale factor based on requirements volatility

For Reuse
Wang et al, 2014

- Adds weights to Size elements, reducing net Size when artifacts are only partially completed

Sys of Sys
Lane et al, 2011

- Adds effort multiplier when in the presence of system-of-systems

COSYSMO 3.0
Alstad, 2016?

- Integrates features of previous models

COSYSMO 3.0 Directions

Incorporate and harmonize existing COSYSMO model research and experience for estimating systems engineering effort:

- Several factors affecting the COSYSMO cost model have been shown to be valuable in increasing estimation accuracy (terminology from [24]):
  - Reuse (partial model—Development With Reuse) [3, 24]
  - Reuse (with Development For Reuse) [24]
  - Requirements volatility (RV) [4]

  The rating scales for these could be integrated into a comprehensive COSYSMO model.

Enhancement included:

- System-of-system considerations are hypothesized to affect system engineering costs:
  - Interoperability considerations [6]

COSYSMO 3.0 Directions Part 2

Enhancements under discussion:

- **Explore a model for total development cost based primarily on the COSYSMO parameters (following work led by Reggie Cole of Lockheed Martin [17, 7])**

Agenda

Agenda:

- The motivation for COSYSMO 3.0
- History of COSYSMO 3.0
- Overview of the content of the COSYSMO 3.0 estimating model
- System-of-systems estimating: interoperability in COSYSMO 3.0
- Model status & plans
- Numerical values of COSYSMO 3.0 parameters
- Summary

COSYSMO 3.0 Top-Level Model

$$PH = A \times (AdjSize)^{E} \times \prod_{j=1}^{15} EM_j$$

Elements of the COSYSMO 3.0 model:

- **Calibration parameter A**
- **Adjusted Size model**
  - **eReq submodel, where 4 products contribute to size**
  - **Reuse submodel**
- **Exponent (E) model**
  - **Accounts for diseconomy of scale**
  - **Constant and 3 scale factors**
- **Effort multipliers EM**
  - **15 cost drivers**

Harmonized COSYSMO 3.0 Size Model

$$AdjSize_{C3} = \sum_{SizeDrivers} eReq(Type(SD), Difficulty(SD)) \times PartialDevFactor(AL_{Start}(SD), AL_{End}(SD), RType(SD))$$

- *SizeDriver* **is one of the system engineering products that determines size in the COSYSMO family (per [2]). Any product of these types is included:**
  - **System requirement**
  - **System interface**
  - **System algorithm**
  - **Operational scenario**
- **There are two submodels:**
  - **Equivalent nominal requirements ("eReq")**
    - **Raw size**
  - **Partial development**
    - **Adjusts size for reuse**

Size Model – eReq Submodel

- **The eReq submodel is unchanged from [2].**
- **The submodel computes the size of a *SizeDriver*, in units of eReq ("equivalent nominal requirements")**
- **Each *SizeDriver* is evaluated as being easy, nominal, or difficult.**
- **The following table contains conversion factors for the conversion of a *SizeDriver* to a number of eReq:**

Size Driver Type	Easy	Nominal	Difficult
System Requirement	0.5	1.0	4.5
System Interface	1.9	4.0	9.0
System Algorithm	1.9	3.8	9.8
Operational Scenario	6.4	13.6	26.3

How Reuse Is Addressed

Reuse has two aspects [1]:

- **Development with reuse (DWR): previously developed artifacts are reused on the current project**
  - **Addressed completely by the DWR partial development model**
- **Development for reuse (DFR): the current project is creating artifacts to be reused on other projects**
  - **One aspect of DFR development is that DFR costs more than ordinary development**
    - **Addressed by the DFR cost driver (below)**
  - **Another aspect of DFR is that the artifacts may be only partially completed, as during an IR&D project**
    - **Addressed by the DFR partial development model**

Size Model – Partial Development Submodel

- **(Concepts here are simplified a little)**
- **The basic DWR concept:**
  - **If a reused *SizeDriver* is being brought in, that saves effort, and so we adjust the size by multiplying the raw size by a *PartialDevFactor* less than 1.**
  - **The value of *PartialDevFactor* is based on the maturity of the reused *SizeDriver*, and is looked up in a table [24].**
    - **How fully developed was the *SizeDriver*?**
  - **If there is no reuse for this *SizeDriver*, then *PartialDevFactor* = 1 (no adjustment).**

DWR Activity Level:	New	Design Modified	Design Im-plemented	Adapted for Integration	Adopted for Integration	Managed
DWR % for this AL through end	100.00%	83.00%	70.13%	56.88%	37.82%	17.50%

- **The basic development-for-reuse (DFR) concept is analogous:**
  - **A product to be reused may be not be taken through the full development cycle (e.g., an IR&D project)**

DFR Activity Level:	Conceptualized for Reuse	N/A	Designed for Reuse	Constructed for Reuse	N/A	Validated for Reuse
DFR % from start through this AL	31.96%		54.60%	78.06%		90.69%

COSYSMO 3.0 Exponent Model

- Exponent model is expanded from Peña [4, 9]

$$E = E_{Base} + SF_{ROR} + SF_{PC} + SF_{RV}$$

Where:

- E_{Base} = A minimum exponent for diseconomy of scale
- SF = scale factor
- *ROR* = Risk/Opportunity Resolution
- *PC* = Process Capability
- *RV* = Requirements Volatility

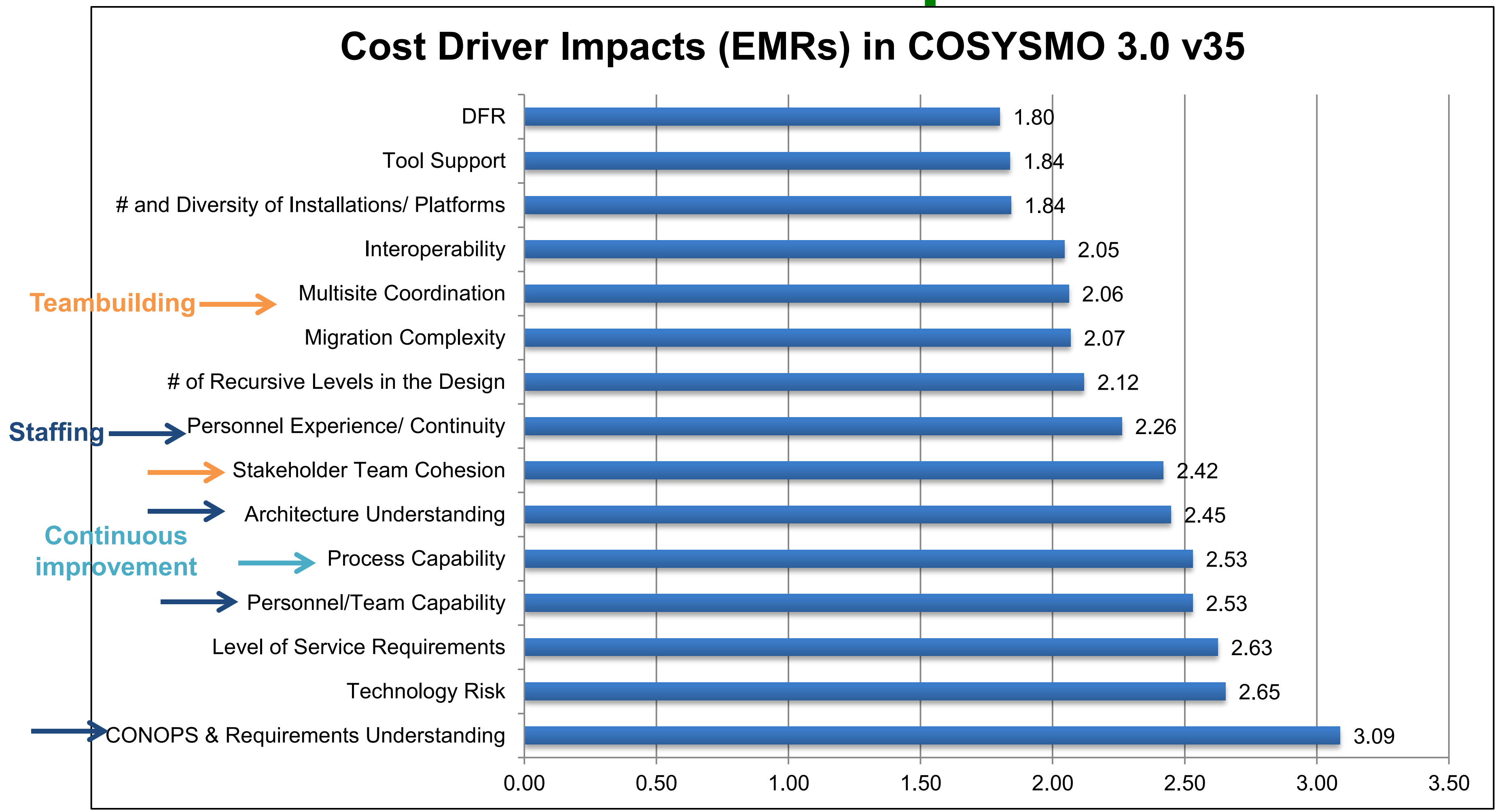
The effect of a large exponent is more pronounced on bigger projects

Harmonized COSYSMO 3.0 Cost Driver Model

- **Here are the 15 cost drivers:**

	Driver Name	**Data Item**
UNDR	CONOPS & requirements understanding	Subjective assessment of the CONOPS & the system requirements
	Architecture understanding	Subjective assessment of the system architecture
	Stakeholder team cohesion	Subjective assessment of all stakeholders
CMPX	Level of service requirements	Subjective difficulty of satisfying the key performance parameters
	Technology risk	Maturity, readiness, and obsolescence of technology
	# of Recursive levels in the design	Number of applicable levels of the Work Breakdown Structure
	Development for reuse	Is this project developing artifacts for later reuse?
OPRN	# and Diversity of installations/platforms	Sites, installations, operating environment, and diverse platforms
	Migration complexity	Influence of legacy system (if applicable)
	Interoperability	Degree to which this system has to interoperate with others
PERS	Personnel/team capability	Subjective assessment of the team's intellectual capability
	Process capability	CMMI level or equivalent rating
	Personnel experience/continuity	Subjective assessment of staff consistency
ENVR	Multisite coordination	Location of stakeholders and coordination barriers
	Tool support	Subjective assessment of SE tools

Harmonized COSYSMO 3.0 Cost Driver Impacts



Agenda

Agenda:

- The motivation for COSYSMO 3.0
- History of COSYSMO 3.0
- Overview of the content of the COSYSMO 3.0 estimating model
- System-of-systems estimating: interoperability in COSYSMO 3.0
- Model status & plans
- Numerical values of COSYSMO 3.0 parameters
- Summary

System-of-Systems and Interoperability

- Suppose that SE work is being done on a system that is a constituent system in a system-of-systems. How is that context manifested in the SE project?
  - Answer: As interoperability requirements
  - Interoperability: The ability of a system to work with another system or group of systems.
- COSYSMO 3.0 includes interoperability as an influence on cost

COSYSMO 3.0 Interoperability Model

- Lane & Valerdi [6] propose that interoperability be considered a cost influence in the COSYSMO family
- Propose this influence could be manifested in two ways:
  - Method 1: Add a new cost driver (covered there)
  - Method 2: Adjust the easy/medium/difficult rating scale for system interfaces (part of the Size model)
- Expert COSYSMO 3.0 includes both methods; only one will be retained in final COSYSMO 3.0.

Size Model – Adjustment for Interoperability

Adjustment for interoperability (Method 2):

- **[6] proposes (in its Table 3) that the table that defines the easy/medium/hard rating scale for a system interface (from [2]) be adjusted by adding a new row (the last row in this table):**

Easy	Medium	Difficult
Simple messages and protocols	Moderate communication complexity	Complex protocol(s)
Uncoupled	Loosely coupled	Tightly coupled
Strong consensus among stakeholders	Moderate consensus among stakeholders	Low consensus among stakeholders
Well behaved	Predictable behavior	Emergent behavior
Domain or enterprise standards employed	**Functional standards employed**	**Isolated or connected systems with few or no standards**

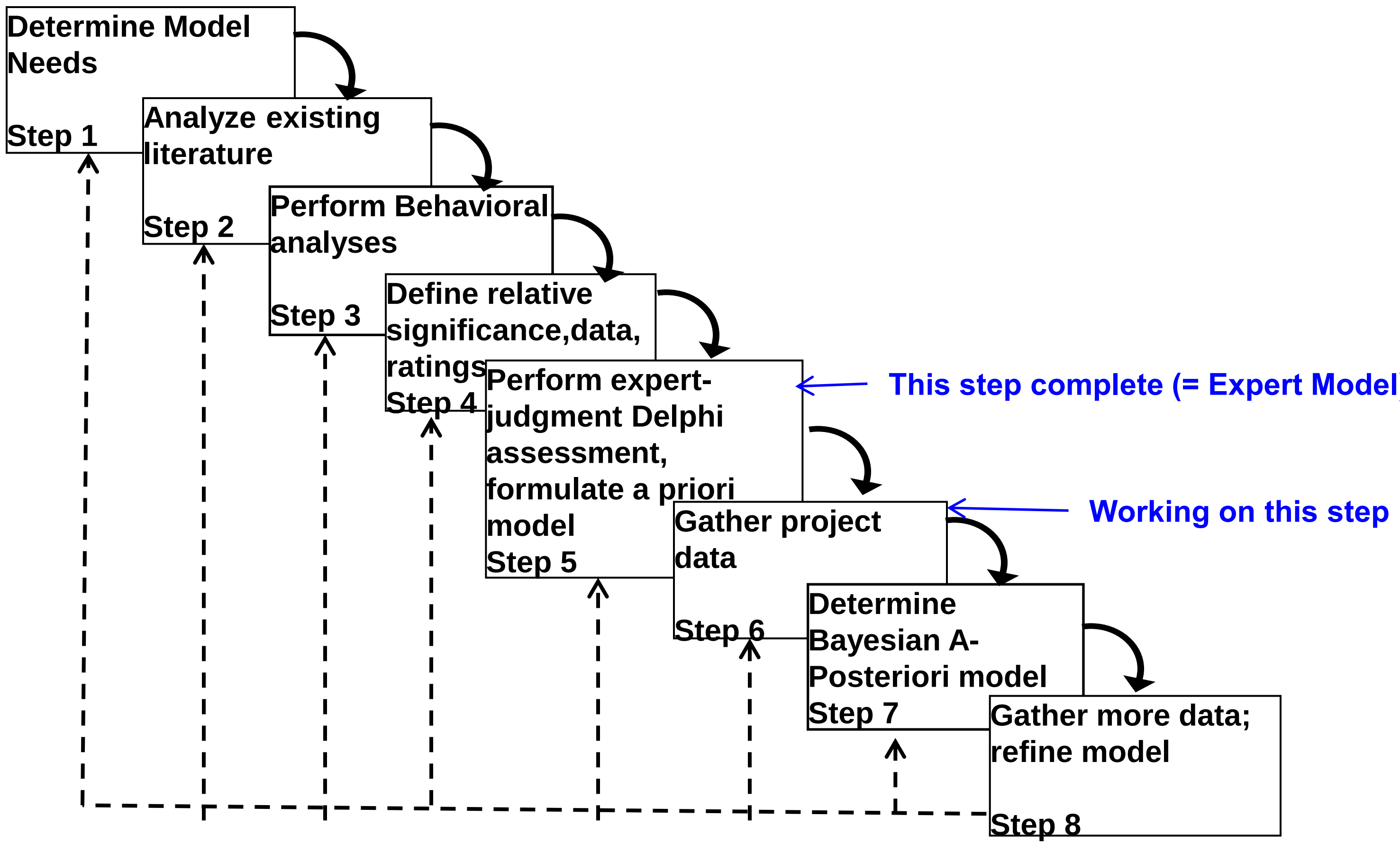
Agenda

Agenda:

- **The motivation for COSYSMO 3.0**
- **History of COSYSMO 3.0**
- **Overview of the content of the COSYSMO 3.0 estimating model**
- **System-of-systems estimating: interoperability in COSYSMO 3.0**
- **Model status & plans**
- **Numerical values of COSYSMO 3.0 parameters**
- **Summary**

USC-CSSE Modeling Methodology

Figure 4.1 from [22]



Model Status & Plans

- The expert-based version of the COSYSMO 3.0 model has been under development for over a year, with critical input from:
  - The COSYSMO 3.0 Working Group
  - Attendees at conferences like this one
- The Expert Model was completed earlier this year
  - Along with a "Rosetta Stone", for rerating old projects under COSYSMO 3.0
- Next work items:
  - Data Collection form
  - Gather new calibration data: completed projects
  - See how model works on existing calibration data

Agenda

Agenda:

- The motivation for COSYSMO 3.0
- History of COSYSMO 3.0
- Overview of the content of the COSYSMO 3.0 estimating model
- System-of-systems estimating: interoperability in COSYSMO 3.0
- Model status & plans
- Numerical values of COSYSMO 3.0 parameters
  - Cost Drivers and Scale Factors
  - (Reuse and Size parameters shown above)
- Summary

Cost Driver Detailed Parameters (1/2)

EMR	Cost Driver	Ratings				
	CONOPS and Requirements Understanding	VL	L	N	H	VH
3.093		1.76	1.33	1.00	0.75	0.57
	Architecture Understanding	VL	L	N	H	VH
2.423		1.56	1.25	1.00	0.80	0.64
	Stakeholder Team Cohesion	VL	L	N	H	VH
2.467		1.57	1.25	1.00	0.80	0.64
	Level of Service Requirements	VL	L	N	H	VH
2.682		1.64	1.28	1.00	0.78	0.61
	Technology Risk	VL	L	N	H	VH
2.581		0.62	0.79	1.00	1.27	1.61
	# of Recursive Levels in the Design	VL	L	N	H	VH
1.932		0.72	0.85	1.00	1.18	1.39
	# and Diversity of Installations/Platforms	N	H	VH	EH	
1.932		1.00	1.25	1.55	1.93	
	Migration Complexity	N	H	VH	EH	
1.996		1.00	1.26	1.59	2.00	
	Interoperability	VL	L	N	H	VH
2.118		1.46	1.21	1.00	0.83	0.69

Cost Driver Detailed Parameters (2/2)

EMR	Cost Driver	Ratings					
	Personnel/Team Capability	VL	L	N	H	VH	
2.690		1.64	1.28	1.00	0.78	0.61	
	Process Capability	VL	L	N	H	VH	EH
2.158		1.36	1.17	1.00	0.86	0.74	0.63
	Personnel Experience/Continuity	VL	L	N	H	VH	
2.315		1.52	1.23	1.00	0.81	0.66	
	Multisite Coordination	VL	L	N	H	VH	EH
1.787		1.26	1.12	1.00	0.89	0.79	0.71
	Tool Support	VL	L	N	H	VH	
1.843		1.36	1.17	1.00	0.86	0.74	
	DFR (Development for Reuse)	L	N	H	VH	EH	
1.638		0.88	1.00	1.13	1.28	1.45	

Scale Factor Detailed Parameters

Maximum	Scale Factor	Ratings					
	Risk/Opportunity Resolution	VL	L	N	H	VH	EH
0.06129		0.0613	0.0490	0.0368	0.0245	0.0123	0.0000
	Process Capability	VL	L	N	H	VH	EH
0.05422		0.0542	0.0434	0.0325	0.0217	0.0108	0.0000
	Requirements Volatility	VL	L	M	H	VH	
0.03788		0.0000	0.0095	0.0189	0.0284	0.0379	

Agenda

Agenda:

- **The motivation for COSYSMO 3.0**
- **History of COSYSMO 3.0**
- **Overview of the content of the COSYSMO 3.0 estimating model**
- **System-of-systems estimating: interoperability in COSYSMO 3.0**
- **Model status & plans**
- **Numerical values of COSYSMO 3.0 parameters**
- **Summary**

Summary

- COSYSMO 3.0 will provide independent estimates of the cost of thorough systems engineering required based on project parameters

Bibliography (1/5)

1. "A Generalized Systems Engineering Reuse Framework and its Cost Estimating Relationship", Gan Wang, Garry J Roedler, Mauricio Pena, and Ricardo Valerdi, INCOSE 2014.
2. "The Constructive Systems Engineering Cost Model (COSYSMO)", Ricardo Valerdi (PhD Dissertation), 2005.
3. "Estimating Systems Engineering Reuse with the Constructive Systems Engineering Cost Model (COSYSMO 2.0)", Jared Fortune (PhD Dissertation), 2009.
4. "Quantifying the Impact of Requirements Volatility on Systems Engineering Effort", Mauricio Pena (PhD Dissertation), 2012.
5. "Life Cycle Cost Modeling and Risk Assessment for 21st Century Enterprises", Barry Boehm, Jo Ann Lane, Supannika Koolmanojwong, Richard Turner (presentation), April 29, 2014.
6. "System Interoperability Influence on System of Systems Engineering Effort", Jo Ann Lane, Ricardo Valerdi, CSER 2011.
7. "COSYSMO Extension as a Proxy Systems Cost Estimation" (presentation), Reggie Cole, Garry Roedler, October 23, 2013.

Bibliography (2/5)

8. "COSATMO: Developing Next-Generation Full-Coverage Cost Models" (presentation), Jim Alstad, USC CSSE Annual Research Review, April 29, 2014.
9. "Quantifying the Impact of Requirements Volatility on Systems Engineering Effort" (presentation), Mauricio Peña, Ricardo Valerdi, October 18, 2012 (COCOMO Forum).
10. "Cost Model Extensions to Support Systems Engineering Cost Estimation for Complex Systems and Systems of Systems", Jo Ann Lane, CSER 2009.
11. "Proposed Modification to COSYSMO Estimating Relationship", Gan Wang, Ricardo Valerdi, Barry Boehm, Alex Shernoff, INCOSE 2008.
12. "Towards COSYSMO 3.0", revised PowerPoint presentation by Gan Wang (filename is "COSYSMO 3.0 Definition Outline.Rev.pptx"), May 2015.
13. "COSATMO: Presenting the Harmonized COSYSMO 3.0 Model" (presentation), Jim Alstad, October 22, 2014 (COCOMO Forum).

Bibliography (3/5)

14. "Towards COSYSMO 3.0", 3rd revision of PowerPoint presentation by Gan Wang (revises [12]) (filename is "COSYSMO 3.0 Definition Outline.Rev3.pptx"), June 2015.
15. "C4ISR Architecture Working Group Final Report - Levels of Information System Interoperability (LISI)", Department of Defense, Washington DC: OSD(ASD(C3I)) C4ISR AWG, 1998.
16. "The Levels of Conceptual Interoperability Model", Tolk, A., and Muguira, J., Proceedings of the 2003 Fall Simulation Interoperability Workshop. Orlando FL, September 2003.
17. "Towards a Holistic, Total Engineering Cost Model", Wang, G., Shernoff, A., Turnidge, J., and Valerdi, R., INCOSE Singapore, July 2009.
18. "COSYSMO Reuse Extension", Wang, G., Valerdi, R., Ankrum, A., Millar, C., and Roedler, G., INCOSE Utrecht, June 2008.
19. "COSATMO: Presenting the Harmonized COSYSMO 3.0 Model" (presentation), Jim Alstad, October 22, 2014 (COCOMO Forum)

Bibliography (4/5)

20. "A Survey on Interoperability Measurement." Ford, T., Colombi, J., Graham, S., and Jacques, D., Twelfth International Command and Control Research and Technology Symposium (12th ICCRTS), Newport, RI, June 19-21 2007.
21. "A Framework For Intelligent Assessment and Resolution of Commercial-Off-The-Shelf Product Incompatibilities", Bhuta, Jesal (PhD Dissertation), 2007.
22. "Software Engineering Economics" Boehm, Barry W., Prentice-Hall, Upper Saddle River NJ, 1981.
23. "The Incremental Commitment Spiral Model", Boehm, B., Lane, J., Koolmanojwong, S., Turner, R., Addison-Wesley, Upper Saddle River, NJ, 2014.
24. "The Generalized Reuse Framework—Strategies and the Decision Process for Planned Reuse", Wang, G., INCOSE SEDC16.
25. "Systems and Software Engineering—Life Cycle Management—Part 1: Guide for Life Cycle Management", ISO/IEC TR 24748-1:2010(E).

Bibliography (5/5)

26. "Guide to the Systems Engineering Body of Knowledge (SEBoK)" version 1.5.1, SEBoK Authors, The Trustees of the Stevens Institute of Technology, Hoboken NJ, 18 December 2015, http://sebokwiki.org/wiki/.
27. "System Verification" page in [26] <http://sebokwiki.org/w/index.php?title=System_Verification&oldid=50858>, 16 June 2015 13:52.