SCEA Luncheon

Software Cost Estimation Using Function Point Analysis

May 25, 2005

Agenda

- Overview of Function Points
- Using Function Points as a Basis of Estimate
- Other Uses and Benefits of Function Point Analysis
- Examples

Overview of Function Points

Function points are not a new concept but only recently have gained wider acceptance as a software size measure

1979

Allan Albrecht (IBM) developed the original idea of function points

1984

First formal function point guidelines published

1986

First official election of the International Function Point Users Group (IFPUG) Board of Directors

1991

Certification criteria for training materials developed

1994

Release 4.0 of Counting Practices Manual

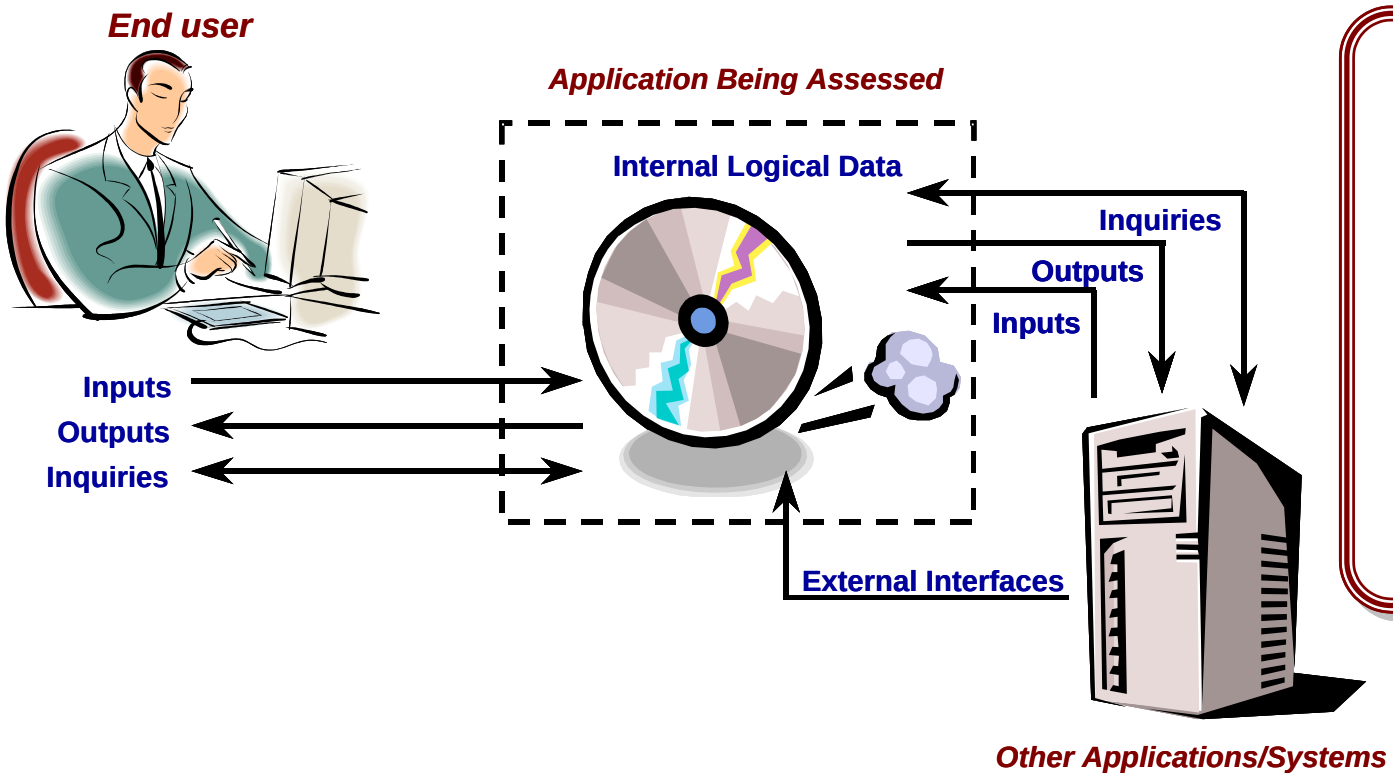
1999

Release 4.1 of Counting Practices Manual

2003

Certified as ISO functional size standard; Release 4.2 of Counting Practices Manual

Function points measure software size based on the functionality requested by and provided to the end user



Function point counting resources

- ✓ *User/analyst interviews*
- ✓ *Requirements documents*
- ✓ *Design documents*
- ✓ *Data dictionaries*
- ✓ *Use cases*
- ✓ *User guides*
- ✓ *Screen captures*
- ✓ *Actual software*
- ✓ *Entity-relationship models*
- ✓ *Semantic object models*

*Function points represent **logical** size, as opposed to **physical** size (like SLOC or objects)*

More complex functions contribute a higher number of function points to the logical size

	Complexity		
Function	**Low**	**Ave**	**High**
Internal Logical File	7	10	15
External Interface File	5	7	10
External Input	3	4	6
External Output	4	5	7
External Inquiry	3	4	6

Function Point Complexity Matrix

- Data functions represent logical groupings of the data end users need to do their jobs
  - Internal data maintained by the application
  - External data referenced by the application
  - Complexity is based on number of data elements and logical sub-groupings
- Transactional functions are the processes and actions end users utilize to manipulate and manage that data in the course of doing their jobs
  - Inputs (add, edit, delete, etc.)
  - Outputs (reports, etc.)
  - Inquiries (search, retrieve, etc.)
  - Complexity is based on number of data elements and files referenced

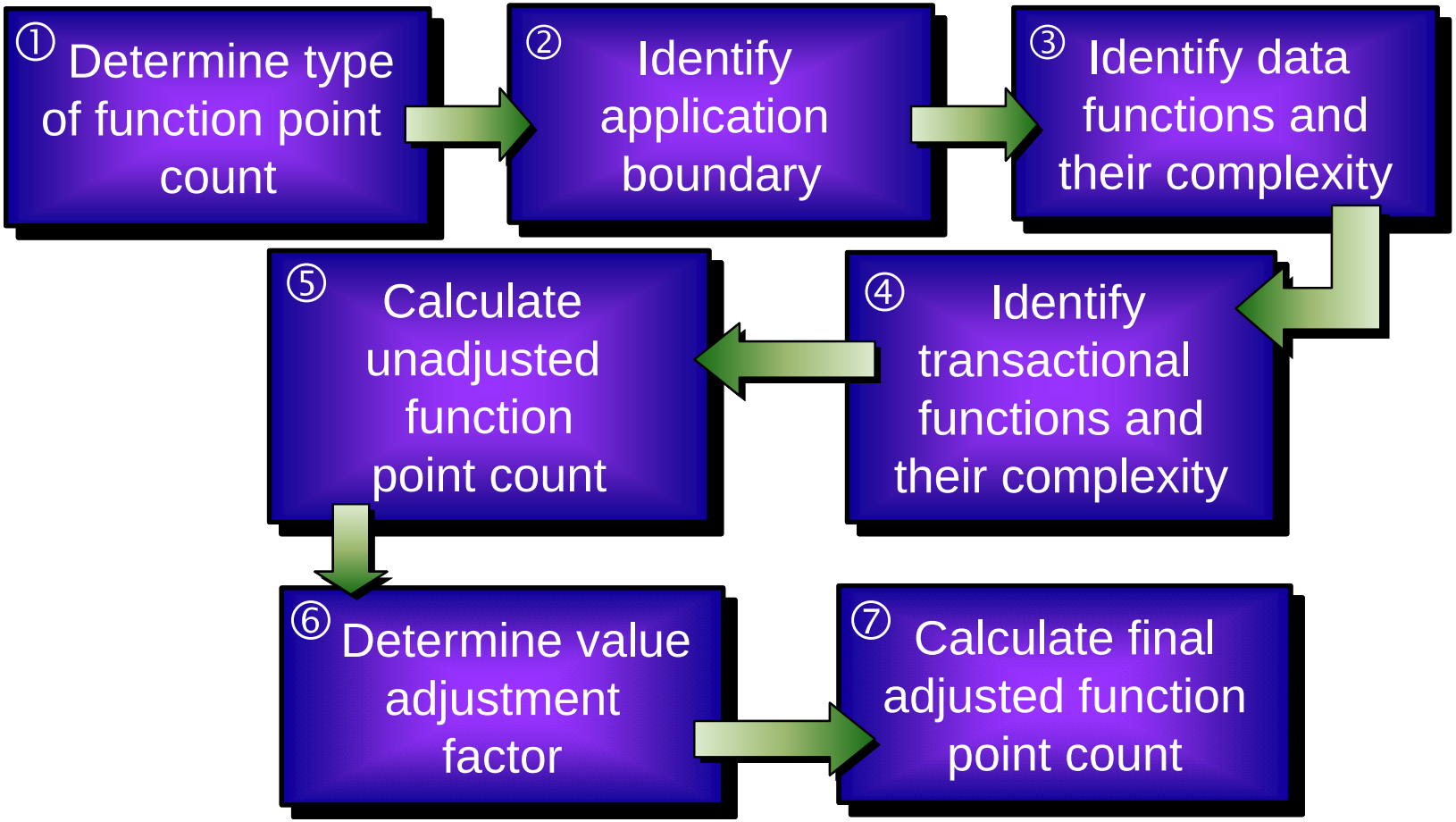
Function point analysis also accounts for more general functionality provided to the end user

Data communications	Online data entry	Installation ease
Distributed data processing	End-user efficiency	Operational ease
Performance	Online update	Multiple sites
Heavily used configuration	Complex processing	Facilitate change
Transaction rate	Reusability	

- The fourteen areas (called general system characteristics, or GSCs) are evaluated against standard criteria and scored from 0 to 5
- The GSC scores are used to calculate the value adjustment factor (VAF), which can increase or decrease the function point estimate by 35 percent

GSCs are more subjective and are not typically used when estimating software costs

The standard function point counting methodology is set forth in the IFPUG *Counting Practices Manual*



Function point analysis (FPA) provides a consistent, documentable, repeatable measurement methodology

- Standards are established and managed by International Function Point Users Group (IFPUG)
- Function points accepted as a standard size measure by ISO (ISO 20926:2003)
- Certified Function Point Specialist (CPFS) professional certification program recognizes trained experts
- Because it is linked directly to system requirements and functionality, FPA puts size analysis into terms that a client or end user can understand
  - Function points can help with communications between the end user community and the developer
    - A client would never say, “I need a system that is 20,000 lines of code”
    - A client says, “Build me a system that does…and supports these processes”

"A World Of Information"
http://www.ifpug.org/

FPA offers many advantages over other software sizing techniques

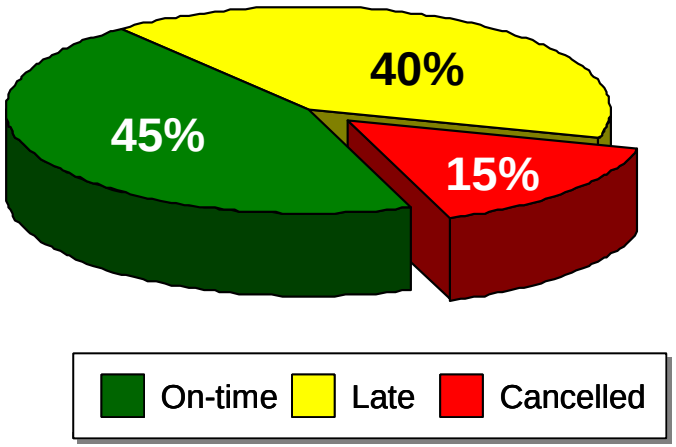
Pros

- Available from early requirements stage and applicable for full life-cycle analysis
- Technology, platform, language independent
- Provides advantages over lines of code (not dependent on engineering estimates, better metrics)
- Fully documentable and repeatable
- Assists with requirements management; functional size traceable throughout entire life cycle
- Provides quantitative basis for earned value management

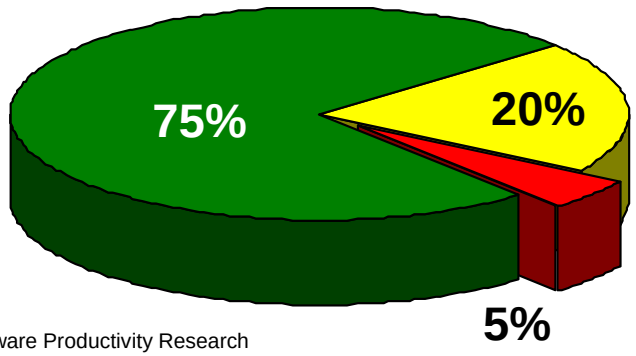
Cons

- Accurate counting requires in-depth knowledge of standards
- Largely manual process
- Some variations exist that aren't standardized (Mark II, 3D, full, feature points, object points, etc.)

Without Function Point Analysis



With Function Point Analysis



Source: Software Productivity Research

Using SLOC-based metrics can lead to the "productivity paradox"

Same software project, 3 different languages

	A	B	C
SLOC Productivity	556	350	333
$ per SLOC	$30	$47	$50
Language	Assembly	Ada 83	C++
Total SLOC	10,000	3,500	2,500
Total Effort (staff-months)	18.0	10.0	7.5
Total Cost	$300K	$166K	$125K

"Best" metrics

Highest effort and cost

Function points metrics are comparable and logical across projects, platforms, and languages

Same software project, 3 different languages

	A	B	C
FP Productivity	2.78	5.00	6.67
$ per FP	$6,000	$3,320	$2,500
Language	Assembly	Ada 83	C++
Total Function Points	50	50	50
Total Effort (staff-months)	18.0	10.0	7.5
Total Cost	$300K	$166K	$125K

Best metrics

Lowest effort and cost

The Certified Function Point Specialist (CFPS) program recognizes trained experts in the field

- Usually start by taking Function Point 101 training class
- Experience counting (with oversight) is also beneficial
- Exam administered at IFPUG Spring Workshops (April) and annual International Software Measurement and Analysis (ISMA) Conference (September)
  - Regional and international exams can also be scheduled
- Exam is 3 hours long (timed) and "open book" (Counting Practices Manual)
  - Section 1: Definitions
  - Section 2: Implementation
  - Section 3: Case Studies

http://www.ifpug.org/certification/CFPSbrochure.pdf

- Why become a CFPS? As a cost analyst, it gives you the ability to completely develop, understand, and justify your software cost estimate.
  - Don't have to rely on other sources for sizing data that may or may not be accurate

Estimating Effort, Cost and Schedule with Function Points

Most of the typical software estimation methodologies can leverage function points as the key size input

- **Manual** – Straightforward calculations that can be done on a spreadsheet or calculator
  - Cost estimating relationships (CER)
  - Analogy
  - Level of effort (LOE)/engineering build-up

- **Automatic** – Models or tools driven by more complex internal algorithms based on historical project data and relationships between software development cost drivers
  - Commercial off-the-shelf (COTS) parametric tool
  - Custom parametric model

The CER approach applies ratios or metrics to a some known or estimated factor to generate effort, cost, and schedule estimates

Example 1

- Estimated 1,200 function points
- Developers estimate 150 FP per effort month for *coding,* which equals 8 effort months
- We have a benchmark CERs that says requirements analysis usually adds 10% to the coding effort, design adds 15% and testing adds another 75% to the coding effort.
- Total effort estimate is 16 effort months

Example 2

- We know the average effort to run a test case is 1.5 hours
- Estimated 200 function points
- We have a benchmark CER that says the number of test cases is equal to the number of function points raised to the 1.2 power
- Each test case will be run 4 times
- Test effort estimate is 3,462 hours

Analogous estimation extrapolates effort, cost, and schedule by comparing known information from similar projects and utilizing adjustment factors to account for differences

Project A (completed)

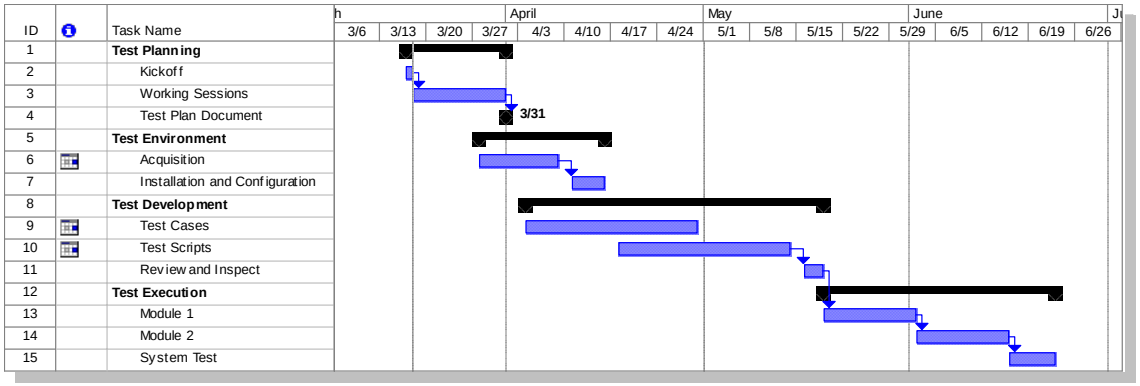
- Actual size: 1,000 function points
- Web platform, JAVA
- Mid-level experience for development staff
- Requirements & Design: 7,781 effort hours and 10 months duration
- Coding: 9,203 effort hours and 5 months duration
- Testing: 9,475 effort hours and 4 months duration

Project B (proposed)

- Estimated size: 900 function points (10% lower than known project)
- Web platform, JAVA
- Highly experienced development staff (increases productivity by 10%)
- Requirements & Design: 6,200 effort hours and 8 months duration
- Coding: 7,400 effort hours and 4 months duration
- Testing: 7,600 effort hours and 3.5 months duration

LOE estimates take a bottoms-up approach by building a detailed activity list and estimating the effort and duration of those activities

- Starts with a work breakdown structure (WBS), which is a detailed decomposition of development activities
- Several LOE estimation methodologies



- *Point estimate* – Individuals predict estimates for specific activities
- *Delphi* – Each member of estimating team provides estimates for each activity in WBS; team discusses high and low estimates collaboratively
- *Three Point* – Each member of estimating team generates low, likely, and high estimates; expected values are averaged and high and low ranges are noted
- *Wideband Delphi* – combination of *Delphi* and *Three Point*

Tough to really incorporate function points into this methodology

Automated COTS estimation tools leverage large historical data sets and flexible input parameters

Examples

SEER

PRICE S

KnowledgePLAN

SLIM-Estimate®

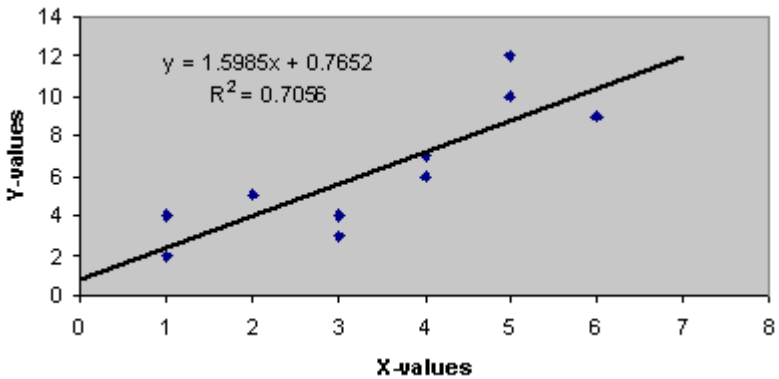
COST XPERT

COCOMO USC

- Tools are parametric in nature, meaning calculations are based on complex statistical algorithms
- Outputs from model are based on input assumptions
  - Size (function points)
  - Personnel skills and experience
  - Development environment
  - Productivity factors
  - Labor rates
- Estimates can be generated with as much or as little information as is available
- Tools typically estimate all development life cycle activities,
- Need to ensure estimated activities match up with expected activities on the project

Custom estimation models require rigorous data collection and documentation of past organizational projects

- Organizations with robust measurement programs can confidently collect and analyze project data to develop internal predictive models
- Models contain effort, cost, and schedule drivers most relevant to the specific organization



- Requires a substantial number of data points
- Data should be scrubbed for accuracy, and it should be maintained for relevancy
- Typically involves multivariate regression
- Need to be very clear on what activities are included in the estimate
- Should apply consistent counting of function points (adjusted vs. unadjusted, etc.)

Other Uses and Benefits of Function Point Analysis

Earned Value Management requires three critical pieces in order to succeed

① Good baseline cost and schedule plans at the project start

- Accurate estimates of budget and schedule are key factors in effective EVM implementation
- Unachievable cost and schedule plans set the project up for failure from the start
- Basis of estimate must be well-defined and clearly understood

② Provide a structured change control process that effectively assesses impact to cost and schedule

- Changes to project scope must be controlled in order for EVM to be effective
- Change control process must be able to demonstrate impacts to cost and schedule of new/changed requirements
- If changes are accepted, cost and schedule baseline plans must be updated

③ Accurately and quantitatively assess percent complete

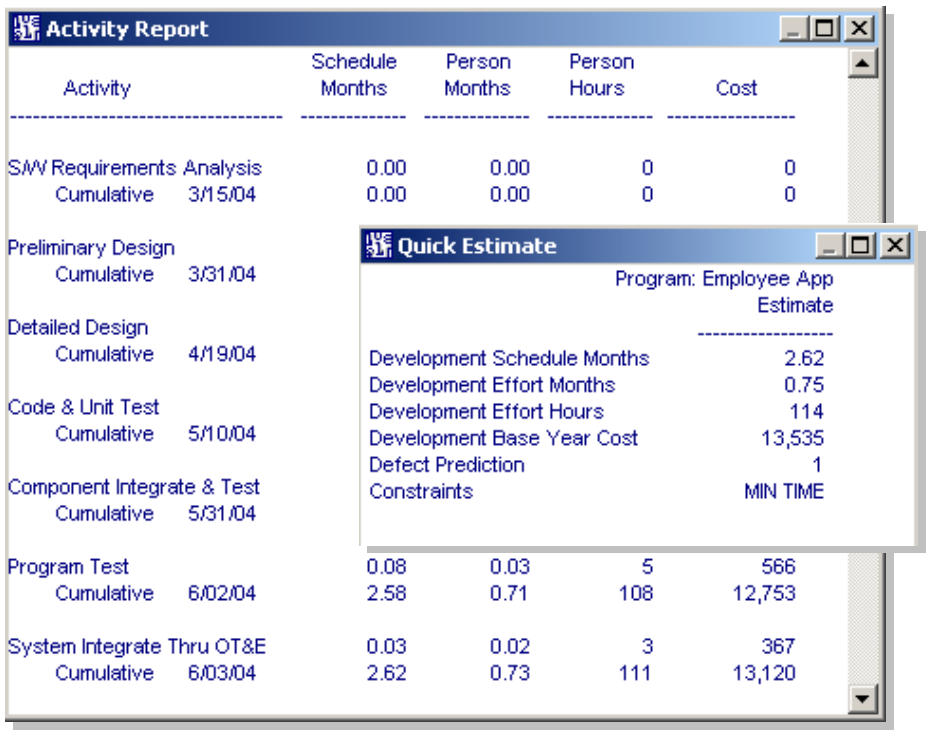
- Earning value is dependent upon progress of the project against the cost and schedule plans
- Work completed must be able to be quantitatively measured and expressed

Function points provide the ability to perform each of these on software development projects

Function point sizing can provide a solid basis of estimate for cost and schedule plans

- Using a software estimation model, function points can then be translated into cost and schedule estimates for the initial project plan
  - Model should be tailored as much as possible to the specific development team and environment, and even calibrated with historical data if possible
- Initial FPA provides a functional baseline in terms of what will be developed
- The process of FPA can also help identify any requirements gaps early in the development life cycle
  - Function points put the size assumptions into terms the client can understand and can review for completeness and accuracy
  - Linking transactions with logical data can help identify where some requirements may have been overlooked or not defined correctly

① Establish a good baseline cost and schedule plan at the project start



Activity Report

Activity		Schedule Months	Person Months	Person Hours	Cost
S/W Requirements Analysis		0.00	0.00	0	0
Cumulative	3/15/04	0.00	0.00	0	0
Preliminary Design					
Cumulative	3/31/04				
Detailed Design					
Cumulative	4/19/04				
Code & Unit Test					
Cumulative	5/10/04				
Component Integrate & Test					
Cumulative	5/31/04				
Program Test		0.08	0.03	5	566
Cumulative	6/02/04	2.58	0.71	108	12,753
System Integrate Thru OT&E		0.03	0.02	3	367
Cumulative	6/03/04	2.62	0.73	111	13,120

Quick Estimate

	Program: Employee App Estimate
Development Schedule Months	2.62
Development Effort Months	0.75
Development Effort Hours	114
Development Base Year Cost	13,535
Defect Prediction	1
Constraints	MIN TIME

Size estimates can be directly linked to baseline requirements

- The resulting project plan can reflect this linkage, which sets up a solid EVM framework

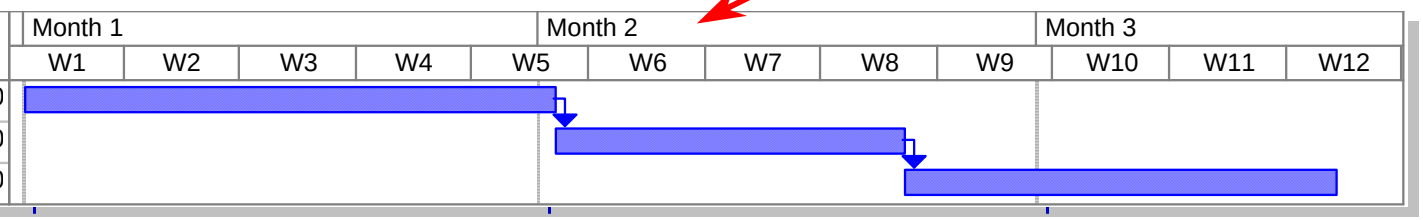
① Establish a good baseline cost and schedule plan at the project start

Plan	Function Points	Design $ Plan	Coding $ Plan	Test $ Plan	Totals
The system shall store employee information	10	$ 1,250	$ 1,518	$ 2,232	$ 5,000
The system shall permit the user to add, change, and delete employee data	9	$ 1,125	$ 1,366	$ 2,009	$ 4,500
The system shall permit the user to search for an employee	3	$ 375	$ 455	$ 670	$ 1,500
The system shall produce monthly employee summary reports	5	$ 625	$ 759	$ 1,116	$ 2,500
Totals	27	$ 3,375	$ 4,098	$ 6,027	$13,500
Schedule (workdays)		24	15	18	57

Planned work allocated to each requirement

Planned schedule with phase milestones

ID	Task Name	Duration	Budget
1	Design	24 days	$3,375.00
2	Code	15 days	$4,098.00
3	Test	18 days	$6,027.00



Cost and schedule re-estimation are essential aspects of an effective, structured change control process

- Requirements for a software development project inevitably change or are redefined as design and development activities progress
  - Before the changes are incorporated into the project scope, impact to cost and schedule must be evaluated
  - The change control process should provide a go/no go decision as to whether the changes should be made to the requirements or delayed for future work

② Provide a structured change control process that effectively assesses impact to cost and schedule

- Because they are linked directly to requirements, function points provide a quantifiable, traceable, and documentable methodology for resizing the software and re-estimating cost and schedule

New

Changed

Plan Revision 1	Function Points	Design $ Plan	Coding $ Plan	Test $ Plan	Totals
The system shall provide login security	25	$ 3,344	$ 4,061	$ 5,972	$ 13,377
The system shall store employee information	10	$ 1,338	$ 1,624	$ 2,389	$ 5,351
The system shall permit the user to add, change, and delete employee data	9	$ 1,204	$ 1,462	$ 2,150	$ 4,816
The system shall permit the user to search for an employee	3	$ 401	$ 487	$ 717	$ 1,605
The system shall produce a monthly employee summary report **and an annual employee summary report**	10	$ 1,338	$ 1,624	$ 2,389	$ 5,351
Totals	57	$ 7,625	$ 9,259	$ 13,616	$ 30,500
Schedule (workdays)		32	20	23	75

This requirements churn would add 30 function points to the project, increase cost by $17K and extend schedule by 18 days

EVM monitoring compares the actual performance against the time-phased plan

- Function points can help quantify progress against the plan
  - When an activity is complete for a requirement, the project has "earned" the value of the function points associated with that requirement
- The logical nature of function points easily apply to any life cycle phase (design, code, test), whereas physical size measures do not

③ **Accurately and quantitatively assess percent complete**

So how is this project doing?

Completed Requirements	Function Points	Design	Coding	Test
The system shall store employee information	10	10	10	0
The system shall permit the user to add, change, and delete employee data	9	9	9	0
The system shall permit the user to search for an employee	3	3	0	0
The system shall produce monthly employee summary reports	5	5	5	0
Totals	27	27	24	0
Percent Complete		100%	89%	0%

Design is complete. Coding is complete on three requirements, which represent 89% of the total functional size

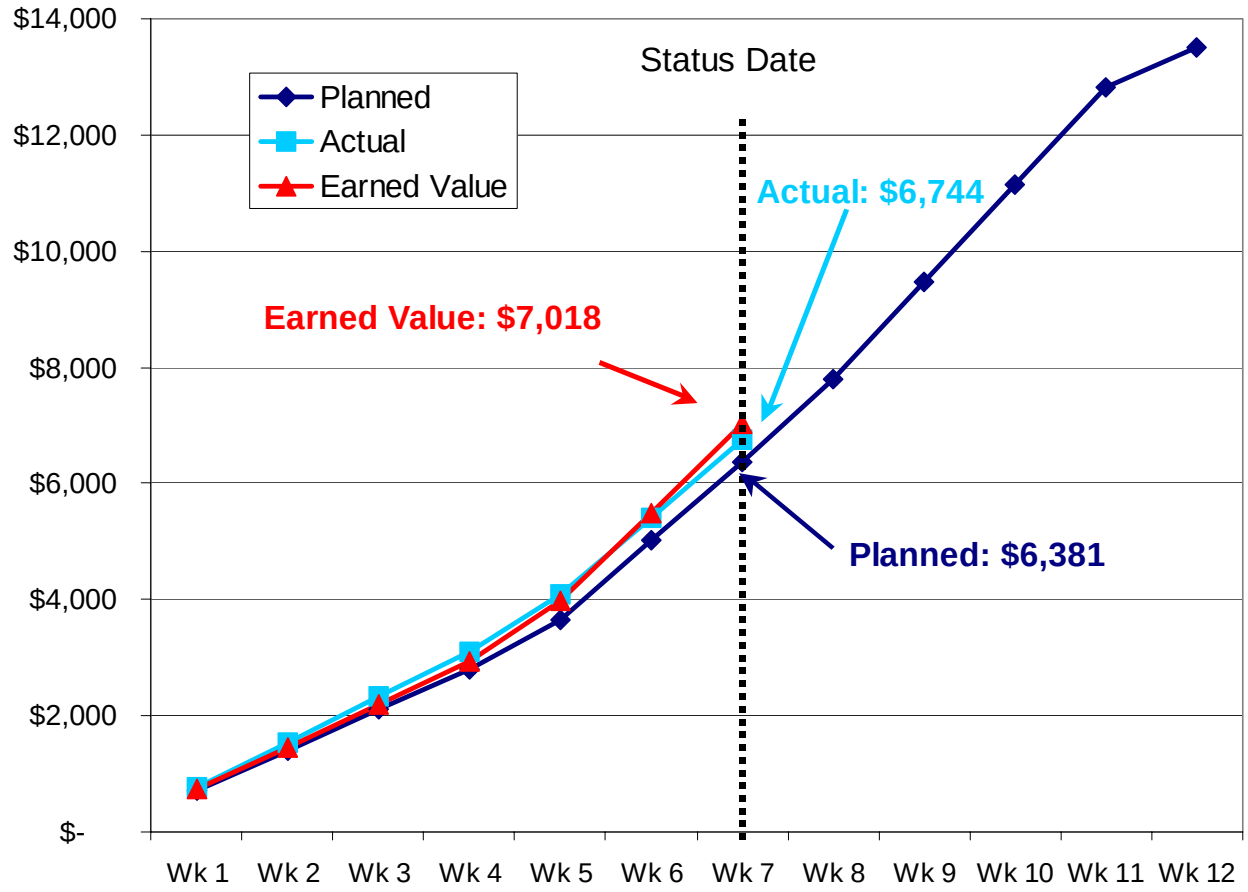
Actual cost of completed work

Actual	Function Points	Design $ Actual	Coding $ Actual	Test $ Actual	Totals
The system shall store employee information	10	$ 1,342	$ 1,273	$ -	$ 2,615
The system shall permit the user to add, change, and delete employee data	9	$ 1,289	$ 1,108	$ -	$ 2,397
The system shall permit the user to search for an employee	3	$ 302	$ -	$ -	$ 302
The system shall produce monthly employee summary reports	5	$ 630	$ 800	$ -	$ 1,430
Totals	27	$ 3,563	$ 3,181	$ -	$ 6,744
Schedule (workdays)		23	12	0	35

EVM tells a very different story than traditional project management

- Comparing planned costs to actual costs, the project might seem over budget
- However, EVM demonstrates that the at this point the project has delivered what was planned in 90 percent of the estimated schedule and at 96 percent of the expected cost

	Planned	Actual	Earned Value
Wk 1	$ 703.13	$ 774.57	$ 733.70
Wk 2	$ 1,406.25	$ 1,549.13	$1,467.39
Wk 3	$ 2,109.38	$ 2,323.70	$2,201.09
Wk 4	$ 2,812.50	$ 3,098.26	$2,934.78
Wk 5	$ 3,648.21	$ 4,093.17	$3,982.14
Wk 6	$ 5,014.29	$ 5,418.58	$5,500.00
Wk 7	$ 6,380.36	$ 6,744.00	$7,017.86
Wk 8	$ 7,808.04		
Wk 9	$ 9,482.14		
Wk 10	$11,156.25		
Wk 11	$12,830.36		
Wk 12	$13,500.00		



Function points can be included in RFPs to provide a common understanding for all bidders

- Function points provide a common assumption of size for all interested bidders
  - Translates requirements into size estimates
  - Function point results can be included as part of the RFP along with requirements document
  - Provides opportunity for discussion/debate over assumptions uncertainties

- FPA can help identify gaps in requirements analysis and avoid early introduction of defects
  - Because it methodically assesses functional requirements, it serves as a cross-check for completeness of requirements and an understanding of how “developable” they are
  - Example: The application shall be able to generate reports
  - Or: Data is identified that is stored within the application but no transactions are included that update that data

Function points can be the foundation of any robust software measurement and analysis program

- Quality measures (defects per function point)
- Size, cost, schedule variation
- Size growth/scope creep
- Productivity measures (effort per function point, cost per function point)
- Delivery rate (function points per calendar month)
- Benchmarking comparisons

Function points are also very effective in sizing and estimating software enhancement projects

Screen: Search MTF

Function Detail/Description
- Search/Sort/Return List
  - Edit MTF
  - Edit MTF Beds
  - View MTF
  - View MTF Beds
  - Print MTF Information
  - Export

Search Fields
- See requirements below

IRRA Original Requirements

CR Ref#	Requirement
34	The system shall provide visibility of non-US military MTFs within a user's area of responsibility (default) by allowing the creation and maintenance of a user defined MTF database

Detailed Requirements

Capability	CR Ref#	SSS Ref#
[DR_TIUIS#104] TIUIS shall provide the capability, for the authorized user, to view a list of facilities (PMRCs, MRAs, MASFs/ASFs, Airfields, Domains) linked with an MTF. Listing shall include: - Facility Type (PMRCs, MRAs, MASFs/ASFs, Airfields, Domains) - Facility Code - Facility Name - Facility Location (City, State, Country) - Facility Comm Phone Number - Facility DSN Phone Number		
[DR_TIUIS#XXX] TIUIS shall provide the capability, for the authorized user, to search for and sort an individual or listing of MTFs for bed capability processing (viewing, updating). Search Fields include: - MTF Code - MTF Name - MTF Type		

Requirements

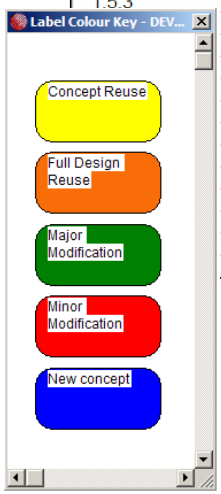
Function Point Workbench
TRANSACTION AND FILE LIST
TRAC2ES — IRRA Functional Enhancements
Baseline Count 01-2005 — Enhancement Project

Identifier	Name	Type	Enhance	Cmplx	UFP
1.5 MTF					
1.5.1	Create MTF	EI	CHANGE	AVE	4
1.5.3	Search MTF	EO	CHANGE	AVE	5
1.5.5	View MTF (facility info)	EQ	CHANGE	LOW	3
1.5.6	Edit MTF facility info	EI	CHANGE	AVE	4
1.5.15	View MTF facility linking info	EI	CHANGE	LOW	3
1.5.16	Edit MTF facility linking info	EI	CHANGE	LOW	3
1.5.17	View MTF assigned med specs	EQ	ADD	AVE	4
1.5.18	Edit MTF assigned med specs	EI	ADD	LOW	3
1.5.19	View MTF med spec info	EQ	ADD	AVE	4
1.5.20	Edit MTF med spec info	EI	ADD	AVE	4
1.5.21	View non-conting bed reporting	EQ	ADD	AVE	4
1.5.22	Edit non-conting bed report	EI	ADD	AVE	4
1.5.23	View contingency bed reporting	EQ	ADD	AVE	4
1.5.24	Edit contingency bed report	EI	ADD	AVE	4

Function Point Size

Functionality can be allocated to more granular reuse/rework categories for more targeted estimation

- This step involves working sessions between cost analysts, developers, and system analysts who are knowledgeable of existing functionality as well as the requested enhancements
  - Identify what Acquisition Method categories are applicable to the given project
  - Use the detailed definitions provided in SEER-SEM help as guidance for the discussion
  - Link requirements with the Acquisition Method category that best describes the type of work necessary to make the enhancement work



Function Point Workbench
TRANSACTION AND FILE LIST
TRAC2ES — IRRA Functional Enhancements
Baseline Count 01-2005 — Enhancement Project

Identifier	Name	Type	Enhance	Cmplx	UFP	Develop/ Reuse
1.5 MTF						
1.5.1	Create MTF	EI	CHANGE	AVE	4	Minor Mod
1.5.3	Search MTF	EO	CHANGE	AVE	5	Minor Mod
	View MTF (facility info)	EQ	CHANGE	LOW	3	Minor Mod
	Edit MTF facility info	EI	CHANGE	AVE	4	Minor Mod
	View MTF facility linking info	EI	CHANGE	LOW	3	Major Mod
	Edit MTF facility linking info	EI	CHANGE	LOW	3	Major Mod
	View MTF assigned med specs	EQ	ADD	AVE	4	New
	Edit MTF assigned med specs	EI	ADD	LOW	3	New
	View MTF med spec info	EQ	ADD	AVE	4	New
	Edit MTF med spec info	EI	ADD	AVE	4	New
	View non-conting bed reporting	EQ	ADD	AVE	4	New
	Edit non-conting bed report	EI	ADD	AVE	4	New
	View contingency bed reporting	EQ	ADD	AVE	4	New
	Edit contingency bed report	EI	ADD	AVE	4	New

	App 1	App 2	App 3	App 4
New	172	-	-	36
Concept Reuse	52	350	-	-
Full Design Reuse	16	-	87	-
Major Mod	110	86	4	-
Minor Mod	42	159	6	44

Engaging in this dialogue helps the technical staff think about the problem and approach in a way that is meaningful to the cost analyst

Examples

RSupply function point example

- Site Subsystem ⇨Activity Control Information ⇨Activity Information

- Data functions
  - Activity Control Information: Average complexity internal logical file ⇨10 function points

- Transactional functions
  - Retrieve activity info (implied): Average complexity inquiry ⇨4 function points
  - Enter activity info: Average complexity input ⇨4 function points
  - Modify activity info: Average complexity input ⇨4 function points

22 Total FP

SITE>ACTIVITY CONTROLS>OWN ACTIVITY>ACTIVITY CONTROL INFO

On the Activity Info tab

1. Enter or modify the mandatory fields:

   Srvc Code, Activity Name, TYCOM, FC Dsgntr, Ship Type, Hull Number, Activity Type, Acty RI, Reporting Officer, Signing Authority, High Money Value.

2. Check **Ship in ILO**, if ship is undergoing an Integrated Logistics Overhaul. (This option is not yet available.)

   Additional options display for selection.

On the Address tab

1. Enter or modify the mandatory fields:

   Address, City, State, Zip.

On the Controls tab

1. Enter or modify mandatory fields:

   Fiscal Year, Transfer Tape, Supply Status.

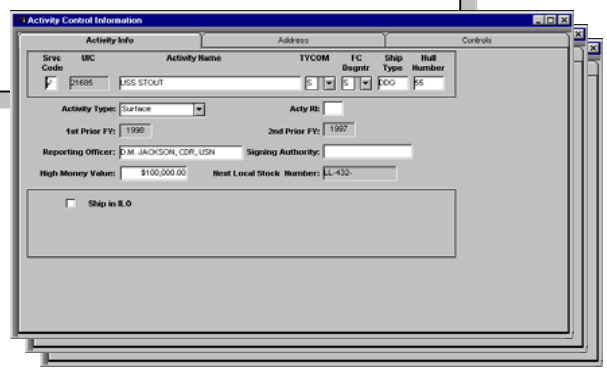
   The next Transmittal Number for the current fiscal year is filled by default.

   The ILO Sequence Numbers is used to maintain ship/ILO tape sequence numbers.

2. Select required Interfaces.
3. The Force Inventory Drawdown allows you to enter a Sequence Number and the Last Date Uplined information.

To complete the process

1. Click **Apply**.
2. Click **Close Window**.



OMMS-NG function point example

- Ship Change Alteration
- Data functions
  - Ship Change Alteration Information: Average complexity internal logical file ⇨10 function points
- Transactional functions
  - Retrieve ship change alteration info (implied): Average complexity inquiry ⇨4 function points
  - Add ship change alteration: Average complexity input ⇨4 function points
  - Modify ship change alteration: Average complexity input ⇨4 function points
  - Delete ship change alteration: Low complexity input ⇨3 function points

25 Total FP

Ship Change Alteration Procedures

Adding a Ship Change Alteration

Although the majority of ship change alterations will be created off ship by the planning organization and uploaded to OMMS NG, you may create a ship change alteration as outlined below.

Menu Path:

FILE>NEW>SHIP CHANGE ALTERATION BUTTON

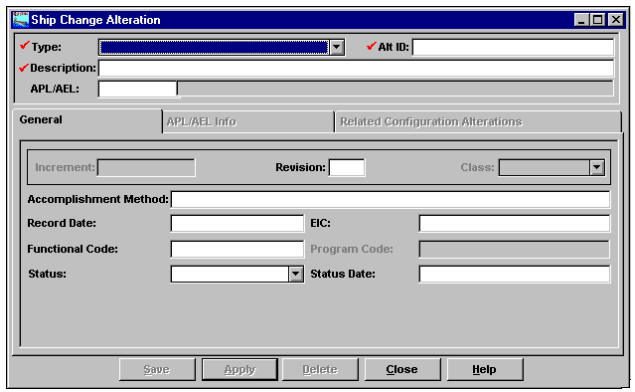
1. At the top of the window:
   - Use the Type drop-down list to select an alteration type and enter an Alt ID and a Description.
   - Enter an APL/AEL and press the Tab key. The APL/AEL information will be filled in.

NOTE: If the APL/AEL does not exist and you have permission to add an APL/AEL, you will get a message asking whether you want to enter the APL/AEL as a local. Click Yes to go to the APL/AEL windows and enter the APL/AEL information.

Double-click in any date field or click on the toolbar to select dates from a pop-up calendar.

2. On the General tab, fill in or select all applicable data.
3. Click Apply to save your changes and continue working or Save to save your changes and close the Ship Change Alteration windows. The Record Date will automatically be filled in with today's date when you click Apply or Save.

NOTE: You must complete all fields with a ✓ before the Apply or Save buttons will become available.

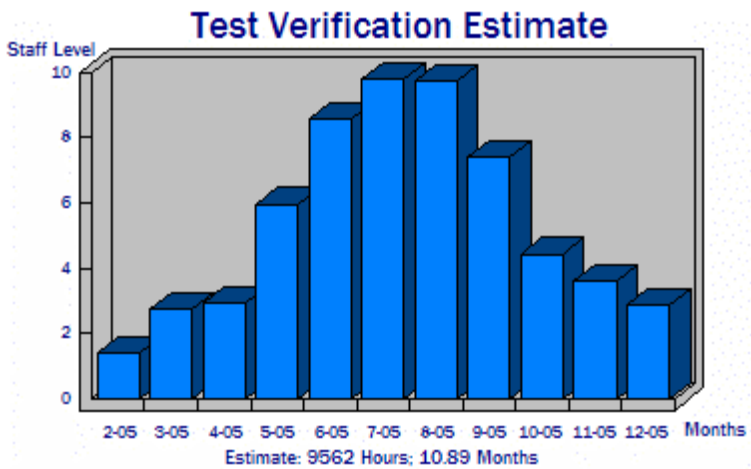


Client wanted to revalidate and retest entire installed baseline application

Installed Application
Size: 1,248
function points

Function Implementation Mechanism		Java	
Programs Included In Size	1	1	1
- FUNCTIONS (Classic)			
- NEW			
New Functions	0	0	0
Software phase at estimate		Done	
- Pre-exists, not designed for reuse	437	437	437
Pre-existing functions	1,248	1,248	1,248
Funcs to be deleted in pre-exstg	0	0	0
Software phase at estimate		Done	
Redesign required	0.00%	0.00%	0.00%
Reimplementation required	0.00%	0.00%	0.00%
Retest required	100.00%	100.00%	100.00%

Probability	Sched Months	Effort Hours		Cost
1%	8.57	4,613	$	423,784
10%	9.14	5,603	$	514,761
20%	9.39	6,072	$	557,826
30%	9.57	6,430	$	590,801
40%	9.73	6,752	$	620,326
50%	9.87	7,065	$	649,104
60%	10.08	7,508	$	689,794
70%	10.30	8,008	$	735,743
80%	10.57	8,630	$	792,871
90%	10.94	9,562	$	878,502
99%	11.86	12,144	$	1,115,725



Project: Air Combat Command Flying Hours Online

Issues

- Developer completed Version 1.0 a software tool for ACC to track flying time performed by pilots
- Team was putting together a white paper on a major upgrade (Version 2.0) and wanted to include potential cost and schedule estimates in the document

Methodology

- Reviewed Version 1.0 user guides, conducted function point analysis of actual application
- Used SEER-SEM to model development effort for Version 1.0
- Conducted FPA against new requirements, developed SEER model to estimate potential cost and schedule ranges for Version 2.0

Results

	Effort	Cost	Schedule
FHOL Actuals	1,589	$99,228	10
Seer 50% Estimate	1,578	$104,196	10.10
Delta	-1%	5%	1%

- Project team liked the results of the initial model, which made them willing to believe the outputs of the second model

Next Steps

- If next version of FHOL gets the green light, work with project team on price proposal and initial project plan
- Baseline Version 2.0 work (size, cost, schedule)
- Update Version 2.0 as scope and requirements change

Project: SPAWAR PMW-151 eNTCSS Baseline Analysis

Issues

- SSCN estimated it would cost $12M to migrate RADM, RSupply, and OMMS-NG to web-based platform
- It eventually cost around $32M
- PMW-151 wanted an audit of the software development effort to identify points of failure and potential areas of improvement

Methodology

- Conducted function point analysis of 3 web-based NTCSS applications
- Modeled size in SEER-SEM, tailored model to system requirements and development environment
- Compared estimates to actuals, conducted cause and sensitivity analyses

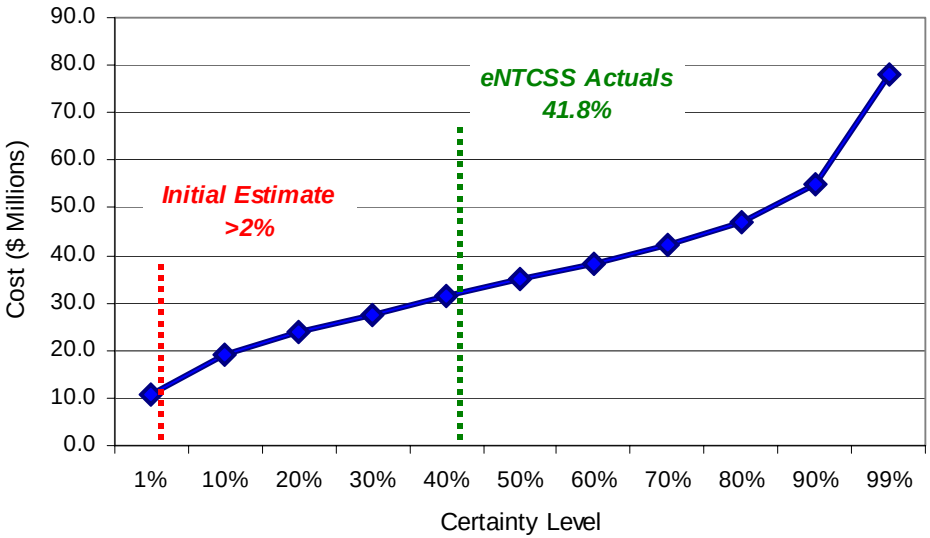
Conclusions

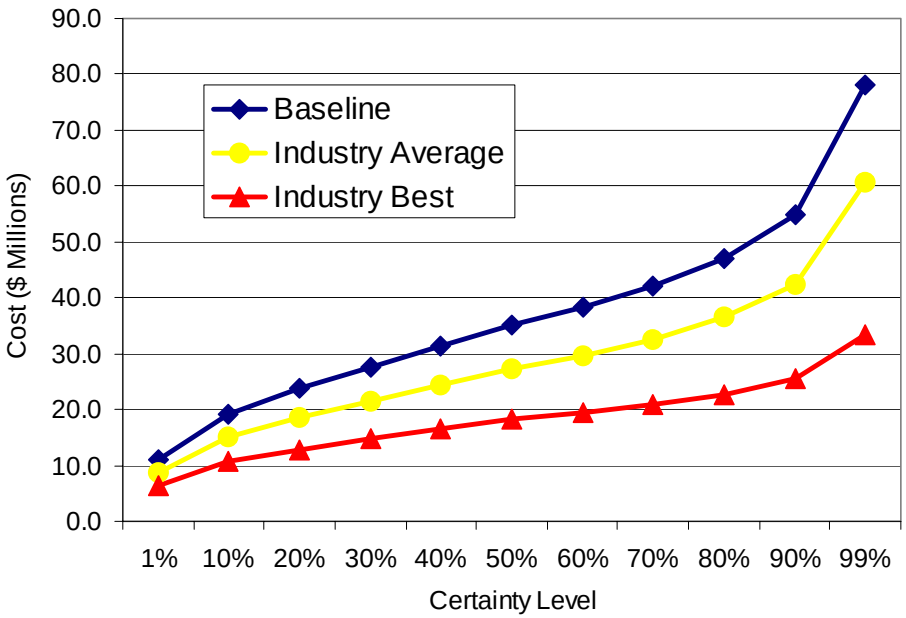
- eNTCSS applications are very large
- SSCN delivered eNTCSS within reasonable cost and schedule
- The initial $12 million eNTCSS cost estimate was extremely optimistic
- With improved process stability and maturity SSCN could have developed eNTCSS for $17 to $25 million

Recommendations

- SSCN should implement more structured cost and schedule estimation processes
- A more consistent means of controlling project scope should be established (needs to include all stakeholders)
- SSCN should continue to pursue process improvement initiatives

Project: SPAWAR PMW-151 eNTCSS Baseline Analysis





Contact Information

Ian Brown,
Certified Function Point Specialist
Senior Associate
McLean, VA
(703) 902-4971
brown_ian@bah.com