F/A-18 Advanced Weapons Lab Software Development Team

"Fleet Products Developed Utilizing Measurement"

Claire Velicer & Sharon Juarez
F/A-18 Software Development Task Team

NAVAIR

What We Do

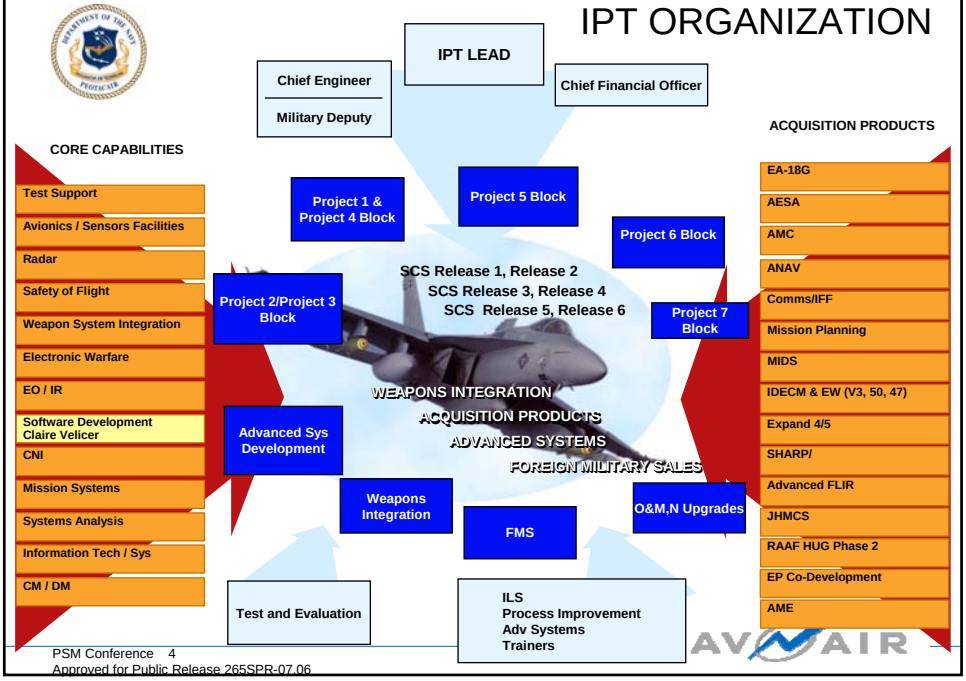
The Advanced Weapons Lab, China Lake -- where Sensor / Smart Plane / Smart Bomb combinations are developed, and wired together to test their real-world, real-time performance - including full-scale, in-lab mock-ups prior to flying..

NAVAIR

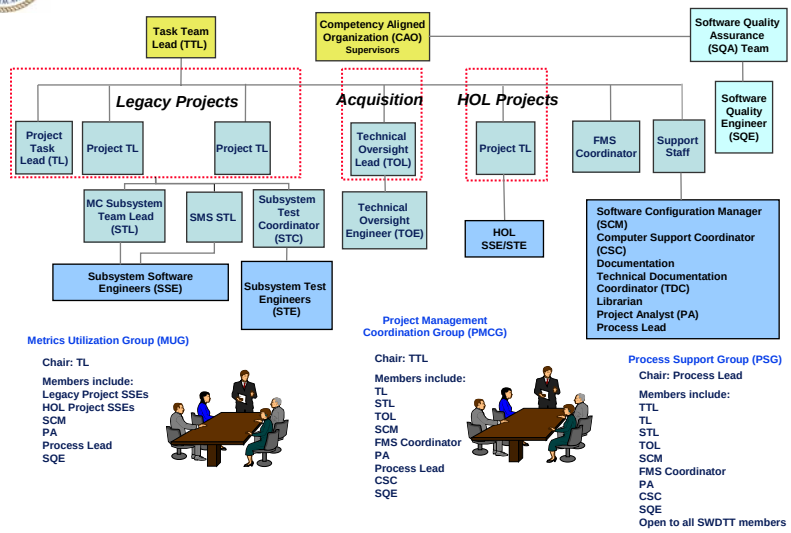
AWL PRODUCTS

- System Configuration Sets
  - Additions to, and modifications of, nearly 12 million lines of software code
- Acquisition Products - stand alone RDT&E projects
- Weapon Integrations
- Fleet Response
  - System problems and new, unpredicted threats
- 7 Foreign Military Sales customer requirements

F/A-18 & EA-18G IPT ORGANIZATION



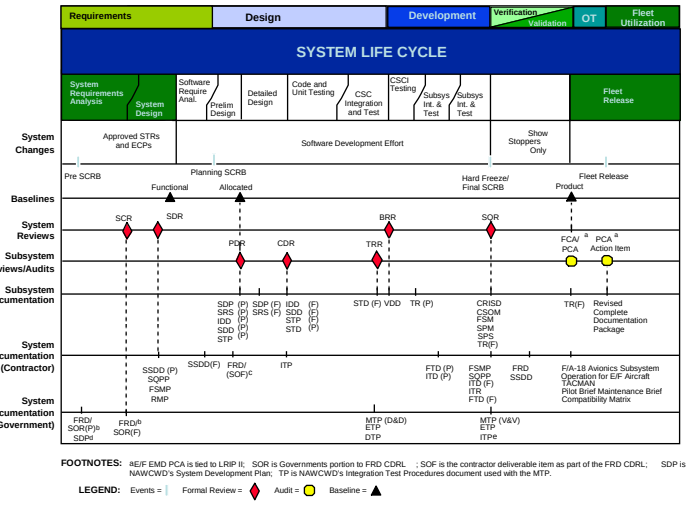
SWDTT Organization



SWDTT Mission

- The F/A-18 Software Development Task Team (SWDTT) will:
  - Provide leadership and expertise in software development and systems engineering to our customers.
  - Produce high quality and defect free products that provide our customers with expanded capabilities to accomplish their missions.
  - Provide challenging and meaningful work, while promoting the personal and professional growth of our workforce.

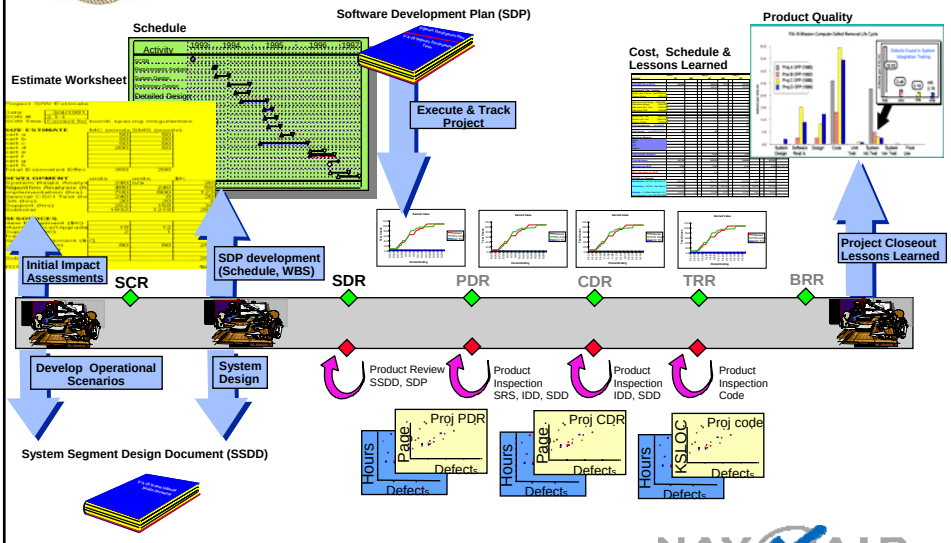
F/A-18 SYSTEM UPGRADE LIFE CYCLE

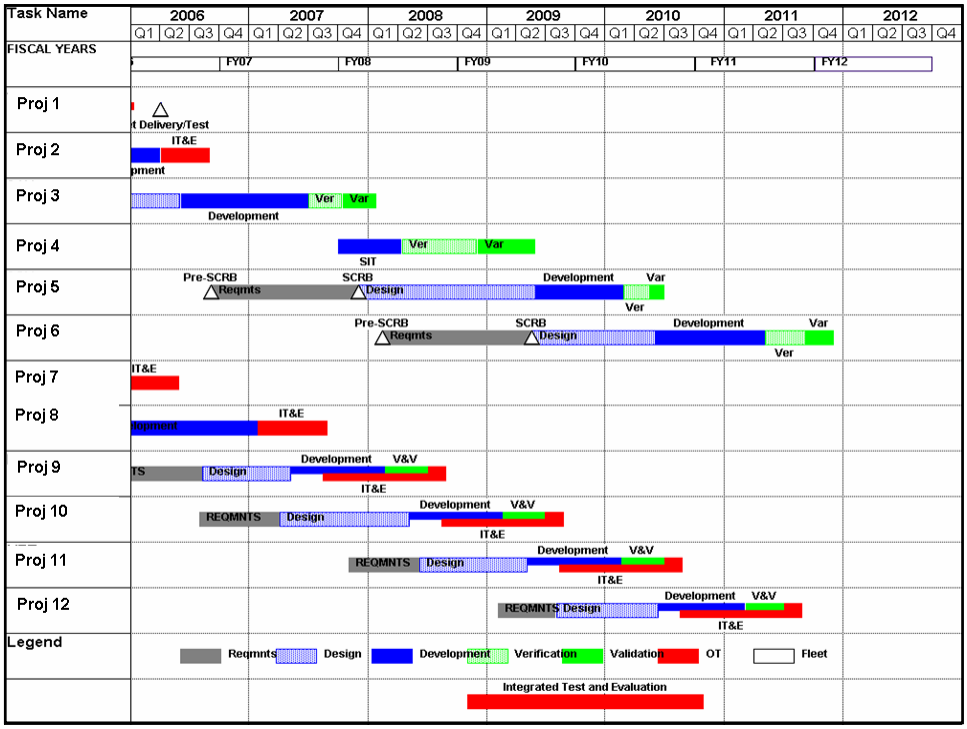


FOOTNOTES: aE/F EMD PCA is tied to LRIP II; SOR is Governments portion to FRD CDRL ; SOF is the contractor deliverable item as part of the FRD CDRL; SDP is NAWCWD's System Development Plan; TP is NAWCWD's Integration Test Procedures document used with the MTP.

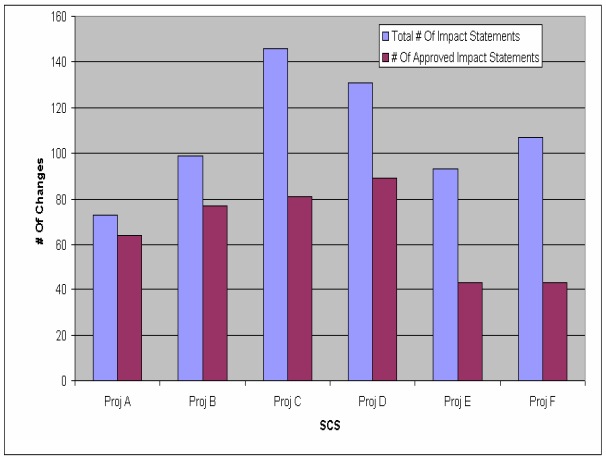
LEGEND: Events = | Formal Review = ◆ Audit = ○ Baseline = ▲

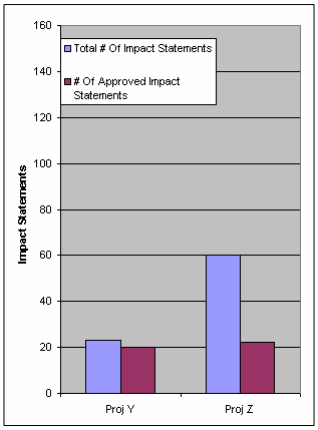
PRODUCTION PROCESS





SCS REQUIREMENT VOLATILITY



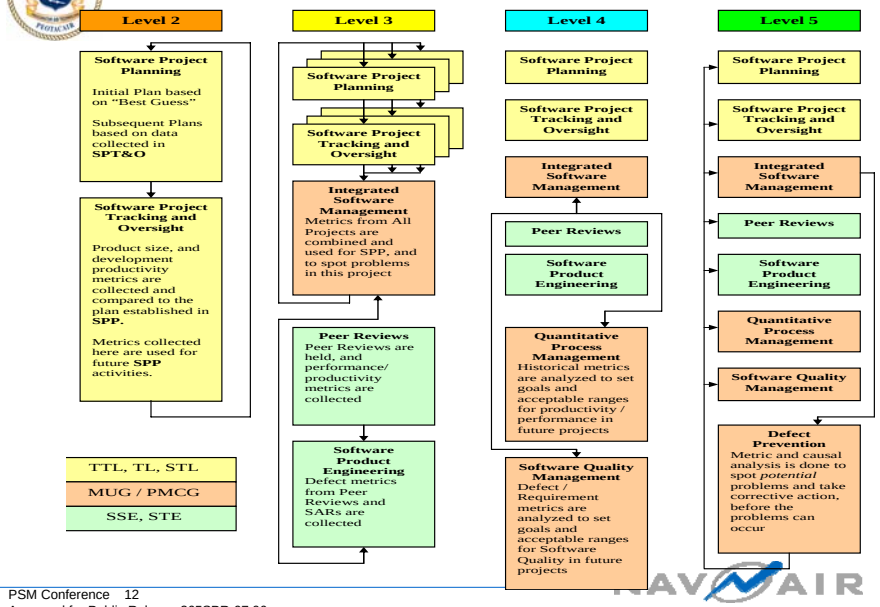


Why Metrics Matter to Us

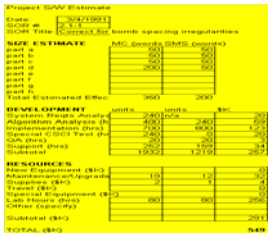
Project Management and Improvement

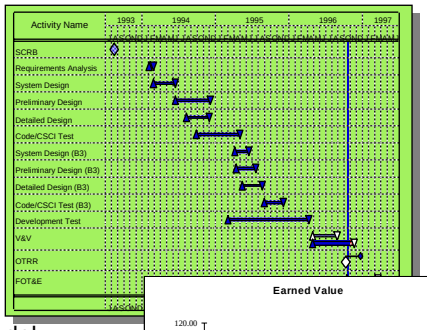
- Project Management
  - Understand where our projects are
  - Are we on schedule?
  - Are we on budget?
  - Is our productivity as planned?
  - What is our estimate to complete?
- Why Improvement
  - Better quality products to the fleet sooner
  - More value for the $ to the warfighter
    - How is our production process performing?
    - Where are our major costs?
    - Where are the escapes in our process?
    - Where can we improve?
      - If you can't measure it – how do you know you've made it better or not?

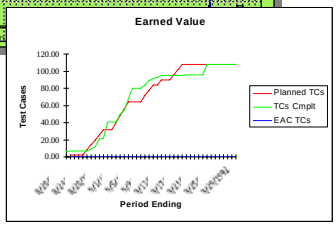
Level 5 Metrics Overview



Repeatable Method of Planning and Tracking



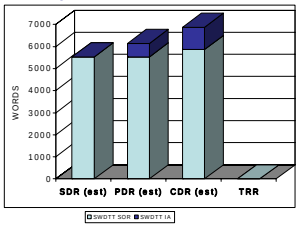


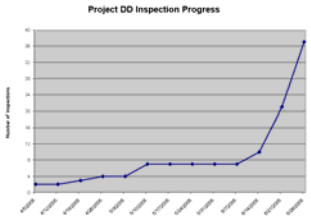


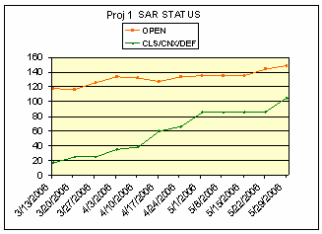
- Planning:
  - Archive and use historic data
  - Compare historic data to cost model
  - Developed procedure for:
  - Impact statement
  - Preliminary based on SLOCs
  - Final estimate based engineer looking at SSDD
- Tracking:
  - WBS tracking earned value
  - Close work package based on quality

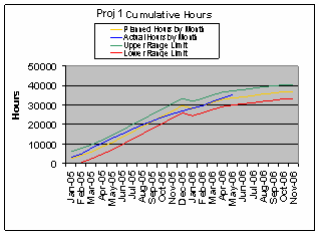
Project Management Metrics

Proj 1 MC OFP Estimated Size

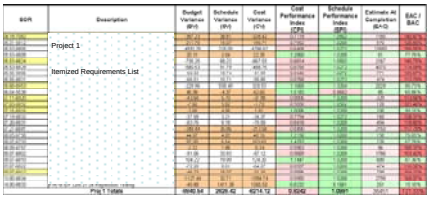


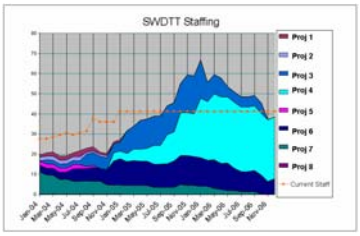


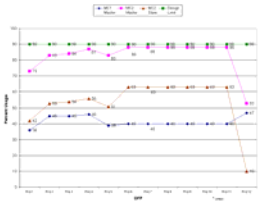


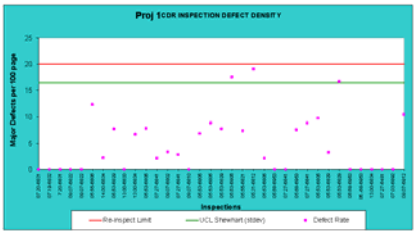


Project Management Metrics

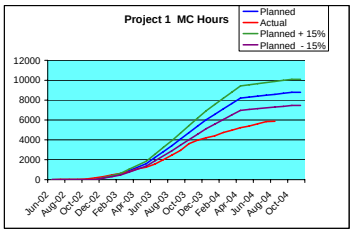


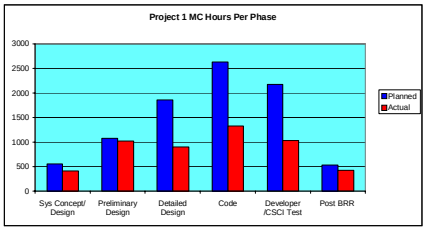


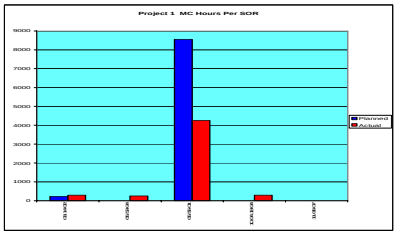




Project Management Metrics







Quantitative Project Management

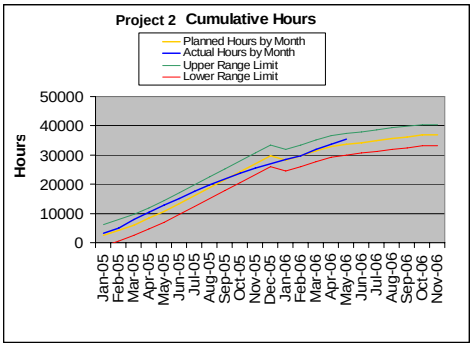
Quantitative Process Management and Software Quality Management Goals

Area	Goal	Metric	Collection
Performance By SOR	Track SPI, CPI and explain and/or take corrective action for indices below .5 or above 2	SPI, CPI	MS Project / Excel / Time Tracker 3
Performance By Project	Track SPI, CPI and explain and/or take corrective action for indices below .75 or above 1.5	SPI, CPI	MS Project / Excel / Time Tracker 3
Defect Removal	Track actual defect density against expected per life cycle phase, explaining or taking corrective action on any inspections where the defect density exceeds the Upper Control Limit (UCL) calculated on historical data.	Defect Density (Defects found per KSLOC or per 100 pages)	Inspection Summary / Excel
	Re-inspect packages when Major Defect Density exceeds .02 defects/SLOC (20 defects/KSLOC) or .2 defects/Page (20 defects/100 Pages)	Defect Density	Inspection logs
	Inspection Preparation Rate should be no greater than 200 SLOCs/hour or 20 pages/hour. Terminate inspection if this has not been met	Preparation Rate (pages or SLOCs per hour)	Inspection logs

- Set for each project
- Recorded in the Software Development Plan
- Monitored on a regular basis
  - Statused formally once a month in project status report
- Evaluated by Metrics Utilization Group to determine appropriate thresholds

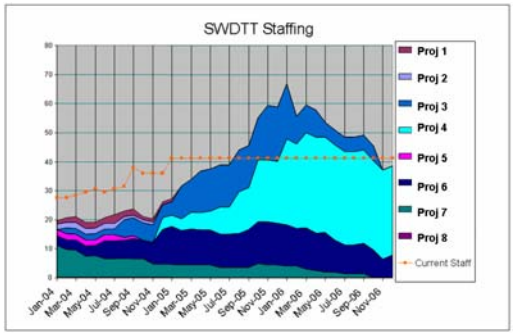
Examples

- Cumulative hours burned on a project was below the threshold set in the goals
  - Action taken evaluate each individual SOR and see which ones were on/off plan
    - Change in requirements?
    - System not maturing?
    - Impacts approved to program?
  - Re-plan where necessary
    - Update estimates for affected SORs
    - Adjust cost and schedule



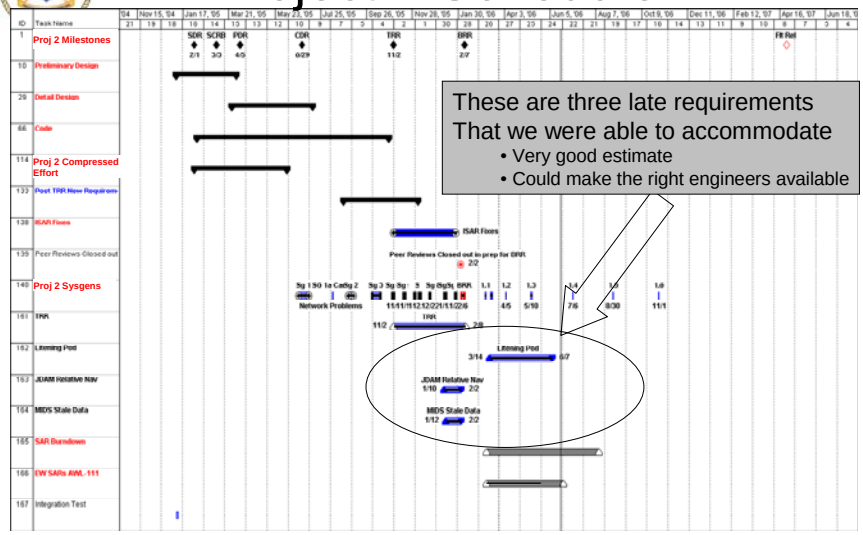
Examples

Staffing Issues



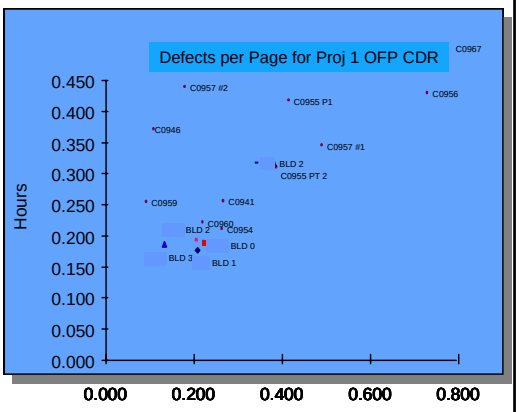
- Can work be re-scoped?
- Is productivity what was expected?
- Productivity –
  - Plan for average engineer
  - Do you have some with outstanding productivity numbers? Can you utilize this to make up schedule?
  - Maximize the use of your work force

Project 2 Schedule



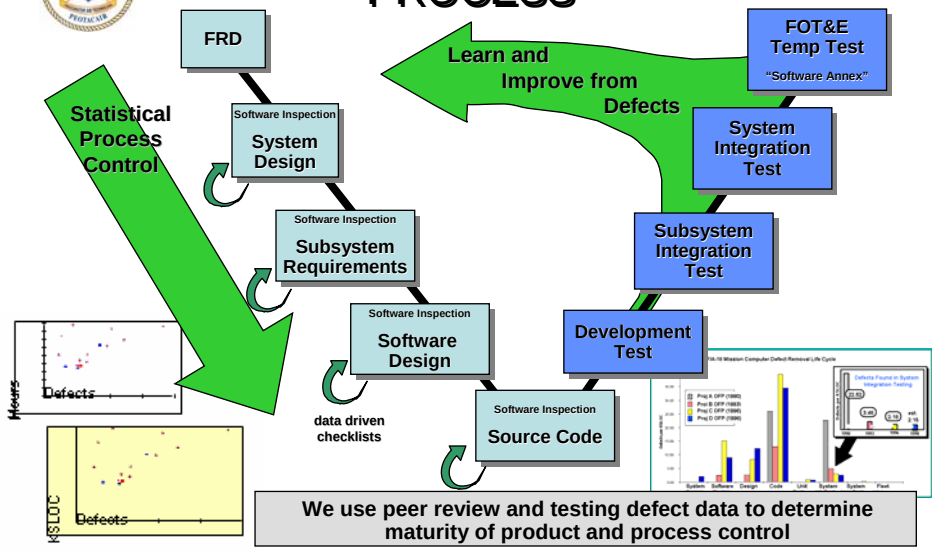
Formal Product Inspections

- **What are Inspections?**
  - Peer reviews of software, test, and documentation products during each of the software development phases
- **Why do we perform Inspections?**
  - Find errors early in the process when errors are less costly to fix
  - Correct non-defects that may lead to maintainability problems
  - Used for process quality control
  - Ensure that a quality product is being developed
- **Who is involved with Inspections?**
  - Customer, TL, STL, SSEs, STEs, Human Factors, SQA, SCM (potentially any SWDTT member)
- **What type of defect data is collected?**
  - Time spent on review (prep rate)
  - Severity of Defects: Major or Minor
  - Class: Wrong, Missing, Extra, Unsure
  - Types: Clarity, Complexity, Data, and many more



Inspections remove defects earlier and cheaper

FORMAL DEFECT REMOVAL PROCESS



We use peer review and testing defect data to determine maturity of product and process control

Defect Types

Type	Key Questions
1. Clarity	1.1. Does each code block have a unique label that matches the flow chart? 1.2. Do code block comments match flow chart block comments? 1.3. Are the scaling comments correct? 1.4. Do the routine headers match the routine header template? 1.5. Are descriptions in comment fields accurate and complete? 1.6. Are the comments consistent and compliant with the FMS Releasability Guidelines (Phrase Table)?
2. Completeness	2.1. Are any blocks from the flow chart missing from the code? 2.2. Does the code implement the design? 2.3. Have tags been inserted in accordance with the Releasability Guidelines? (SOR Releasability Matrix)
3. Complexity	
4. Consistency	4.1. Is the existing module scheme for saving return addresses continued? 4.2. Do Jump instruction match flow chart labels? 4.3. Is the name of a new SYSPROC in the header and footer (CMS)? 4.4. Is the header block's OPTION field set-up correctly (CMS)? 4.5. Are parameters modified before being saved?
5. Constraints	5.1. Does the code adhere to the constraints identified in requirements and design?
6. Data	6.1. Are all data parameters in the parameter database? 6.2. Do Store/Load for multiple instructions access correct number of words? 6.3. Do parameters of less than 16 bits use the correct mask? 6.4. Are the proper temporary (VVxxxx) variables being used? 6.5. Did you ensure that the sign bit is not inadvertently overwritten? 6.6. Can the data be out of range?
7. Error Handling	7.1. Is division by zero avoided? 7.2. Is there any overflow/underflow/unwanted truncation during calculation or shifting.
8. Functionality/Logic	8.1. Are literal instructions used when loading values less than 16? 8.2. Are overflow/underflow conditions addressed during calculations and bit shifting (scaling)? 8.3. Are the scale values used during operations correct 8.4. Are the register/parameter usage for AYK-14 arithmetic operations correct? 8.5. Are there any register conflicts between routines and their callers?
9. Initialization	9.1. Are Initial entry value of registers saved when necessary? 9.2. Is the Stack pointer properly initialed before the routine call.
10. Interfaces	10.1. Are the Registers correctly loaded prior to invocation of a math subroutine.
11. TBD	11.1. Is there any portion of the code that has not yet been determined?
12. New/Changed	12.1. Have any new requirements surfaced that have not been addressed? 12.2. Has a requirement changed?
13. Resource Management	13.1. Has the code been written to conserve scarce resources?
14. Standards	14.1. Have all standards been followed? 14.2. Does the style conform to policy, process, and procedure? 14.3. Have FMS tags followed the Releasability Program Package User's Guide?
15. Testability	15.1. Is the code testable?
16. Traceability	16.1. Is the code traceable to the detailed design?
17. Non Coding	17.1 Is the package content complete?-

- List of Defect types unique to every phase and subsystem
- Refined by metrics feedback from past inspections
- Used by engineers to categorized the defects found in during Product Inspections

Sample Inspection Log

Inspection Summary Report

Inspection Date: 8/1/2005 | Memo #: 5286 | Project: Proj 2 | Subsystem: RC | Re-Inspection? Yes ☐ No ☑ | Build: 1 | Development Phase: Code

Title: Part 1 Sample Code Inspection Package

TC: N/A | Part 1 of 4 | CPCR: XXXX | CPDR: N/A | SDP(s): 05.XX-XXXX | SAR(s): N/A | STR(s): N/A | Coordinator: S. Glenn | Moderator: Larson | Designer(s): Quint, Garcia, DeYeokes | Recorder: Larson | Technical Reviewers: Wilson; Miller | Observers: | Setup Reviewers

Metrics Data:			Re-Inspection Criteria:
Package Size	Total	443	SLOCs >500 SLOCs
	Added	438	
	Modified	5	
	Deleted	0	
Total Number of Technical Reviewers:		2	<2 persons
Total Technical Reviewers Preparation Time (Hrs):		9.50	
Total Preparation Time (Technical Reviewers and Observers):		9.50	
Preparation Rate:		93.26	SLOCs/hr >200 SLOCs/hr
Total Major Defects (excluding Legacy Defects and Package Defects):		1	
	Wrong (W)	0	
	Missing (M)	0	
	Extra (E)	0	
	Unsure (U)	1	
Total Minor Defects (excluding Legacy Defects and Package Defects):		6	
	Wrong (W)	2	
	Missing (M)	4	
	Extra (E)	0	
	Unsure (U)	0	
Total Defects (excluding Legacy and Package Defects):		7	
Major Defect Density:		2.26	per kSLOCs 20 per kSLOCs
Defect Density:		15.80	per kSLOCs
Legacy Defects:	Major 0	Minor 0	
Package Defects:	Major 0	Minor 0	
Number of Action Items:		0	
Meeting Time (hours):		0.50	Time
Post-Meeting Developer Time (hours):		0.17	Time
Post-Meeting Moderator Time (hours):		0.26	Time

Product Disposition: Accept ☐ Conditional Accept ☑ Re-Inspection Required ☐

I certify that this package is complete and satisfies the completion (exit) criteria.

E-mail sent ☑ | Initials (typed): kl | Moderator signature: | Date: 08/24/05

(Office use only) Metrics collected ☑

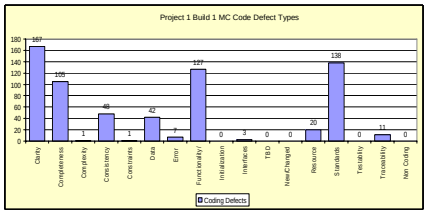
Inspection Discrepancy Log

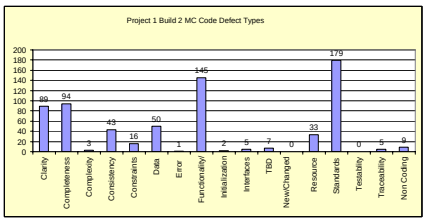
Reviewer: Wilson | Technical Reviewer ☑ | Reviewer Time: 2.00 Time | Developer Correction Time: 0.00 Time | Development Phase: Code | Project: Proj 2 | CPCR/CPDR: N/A | Part 1 of 4 | CSCI: RC | SSDD: XXXX | Title: Part 1 Sample Code Inspection Package

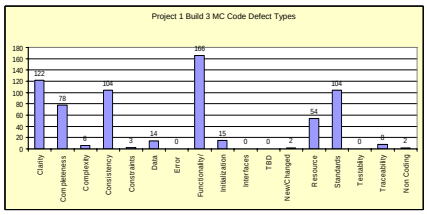
This form is used to record defects and issues to be discussed in the review meeting. All items recorded on this form will be tracked through the defect removal process. (This does not mean that all defects are fixed, merely that all defects are recorded.) Note that you may optionally "red-line" the documents and attach to this log. This log should be given to the recorder at the end of the inspection meeting, even when no errors are recorded, so that pre-meeting time is tracked.

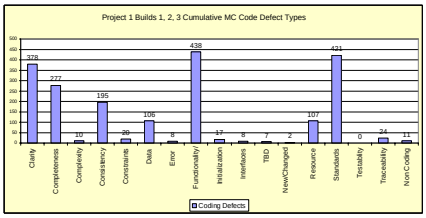
Product & Version	Page	Line of Code	Section/ Subsection	Severity	Class	Type	Description	Defect Completed
Severity: MA = Major m = minor For Trivials, please "red-line" document							Class: W = Wrong, M = Missing, E = Extra, U = Unsure, L = Legacy Type: I–16 See Checklist	
CODE	10	9171	YSFAK450	Deleted	Extra	3	~~Could use a common Load and Store for YSFLAG in blocks 400 and 450.~~	
CODE	15	6001	HEADER	minor	Missing	2	R9 is also used for XYZWSP	8/17/05 GG
CODE	15		HEADER	minor	Missing	3	R13 and R14 are also used.	8/17/05 GG
CODE	17	6974	YSFAK450	minor	Missing	14	Missing comment.	8/17/05 GG
CODE	17	6976	YSFAK450	minor	Missing	14	Missing comment.	8/17/05 GG
CODE	22	13837	YSFAK200	Deleted	Extra	3	~~Could move LLRS instruction to block 600 to make it common for blocks 700 and 800 to save a word.~~	
CODE	30	15015	YMFAK330	minor	Wrong	14	Block description does not match flow.	8/17/05 GG
CODE	30	15049	YMFAK476	Dup	Wrong	4	YSHNH is cleared in the code, but the flow states that YSHNH should be cleared.	

Defect Type Distribution









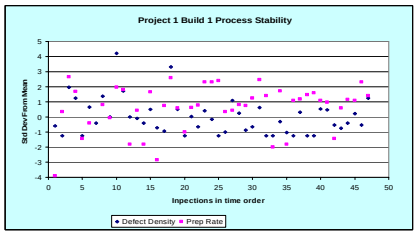
Package Size Table

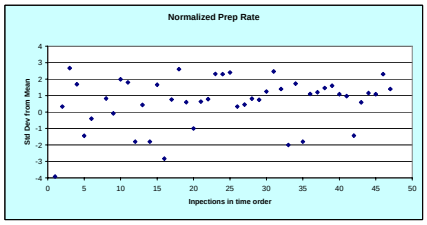
Subsystem	Phase	Package Size Unit	Size limit	Prep. Rate (per hour)	Major Defect Density
MC	System Requirements	pages	50	20	0.2
	System Design	pages	50	20	0.2
	Preliminary Design	pages	50	20	0.2
	Detailed Design	pages	50	20	0.2
	Code	SLOCs	500	200	0.02
SMS	System Requirements	pages	50	20	0.2
	System Design	pages	50	20	0.2
	Preliminary Design	pages	50	20	0.2
	Detailed Design	pages	50	20	0.2
	Code	SLOCs	500	200	0.02
SRMS	System Requirements	pages	50	20	0.2
	System Design	pages	50	20	0.2
	Preliminary Design	pages	50	20	0.2
	Detailed Design	pages	50	20	0.2
	Code	SLOCs	1000	400	0.02
EOPT	System Requirements	pages	50	20	0.2
	System Design	pages	50	20	0.2
	Preliminary Design	pages	50	20	0.2
	Detailed Design	pages	50	20	0.2
	Code	SLOCs	1000	400	0.02
ADVSYS	System Requirements	pages	50	20	0.2
	System Design	pages	50	20	0.2
	Preliminary Design	pages	50	20	0.2
	Detailed Design	pages	50	20	0.2
	Code	SLOCs	1000	400	0.02
OTHER	System Requirements	pages	50	20	0.2
	System Design	pages	50	20	0.2
	Preliminary Design	pages	50	20	0.2
	Detailed Design	pages	50	20	0.2
	Code (Assembly)	SLOCs	500	200	0.02
	Code (HOL)	SLOCs	1000	400	0.02

- Recommended Package size by processor by phase
- Recommendation is results of metrics analysis
  - Defects/page
  - Prep rate
  - Major Defect Density
- Moderator is responsible for making inspection pass/fail determination
  - Thresholds are allowed to be crossed with justification
  - Failure requires rework and re-inspection

Metrics Analysis

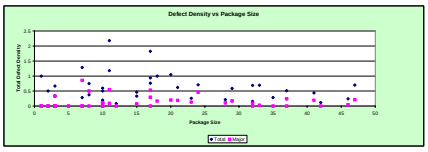
- This is a sample from Project 1 Build 1 CDR
- Plots Defect Density
- Plots Prep Rate
- Looking for outliers
  - If found need an explanation why or should re-inspect

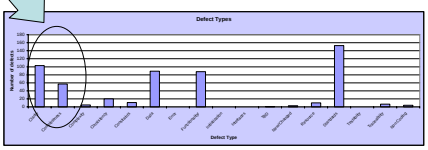




Metrics Analysis

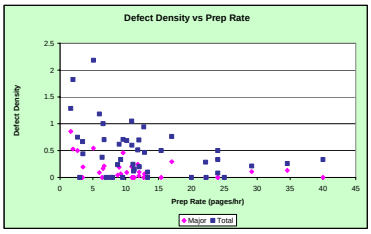
- Defect Density vs Package size
  - Very controversial with team
  - This dictated a procedural change to smaller package sizes
  - Issues with institutionalization
    - Engineers were meeting the package size
    - Packages were too small from the perspective that the subdividing of design material was causing continuity errors to be missed
    - Resulted in education that package size was a guideline and engineering judgment should be used to not only make manageable sized packaged, but logical engineering segments in each package

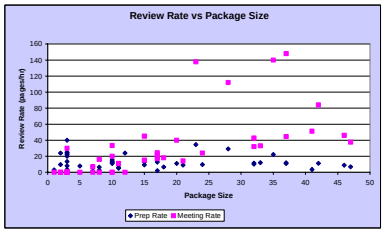




Metrics Analysis

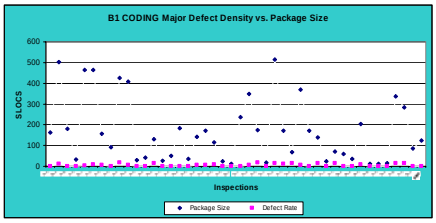
- Adequate Prep Rate
  - Helps indicate quality of review
  - Cancel review if prep rate not met
  - Not cost effective to over-prep
- Guidelines for prep-rate were established based on performance analysis of previous inspections

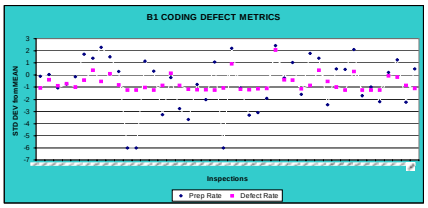


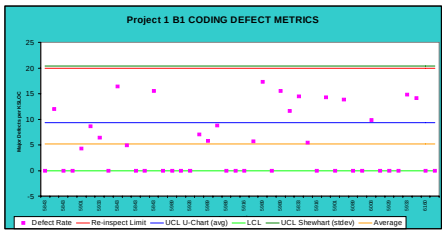


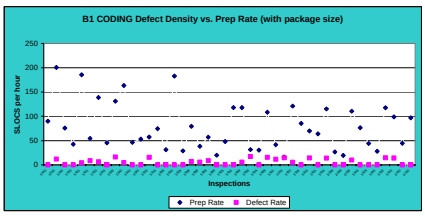
Metrics Analysis

- Project 1 Build 1 Coding metrics sample



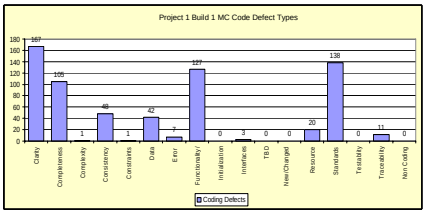


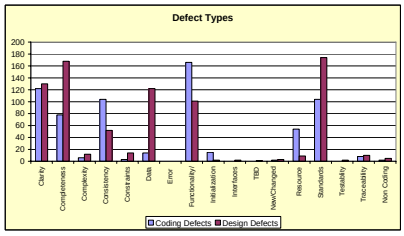




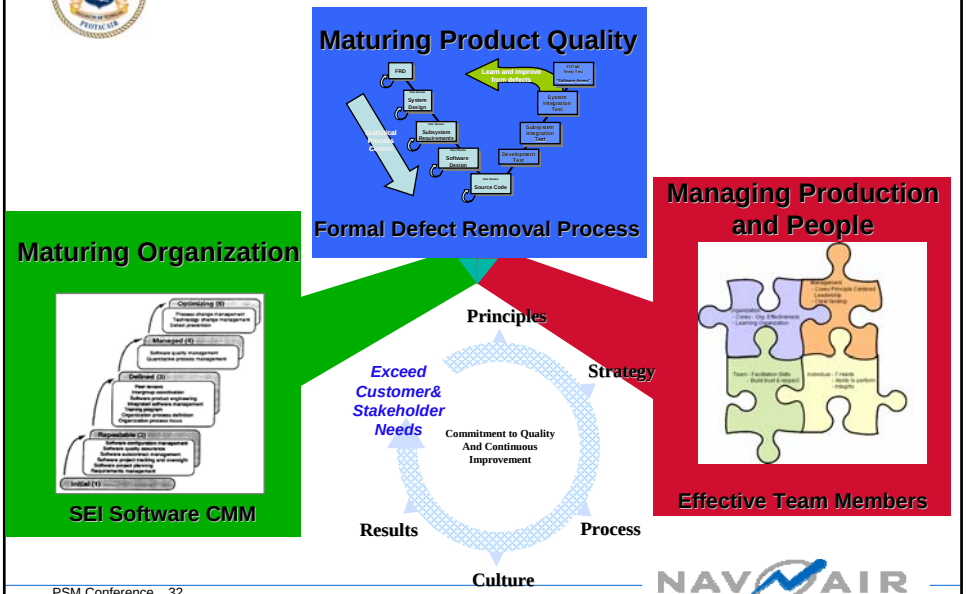
Metrics Analysis

- Root cause analysis
  - Cause of most errors
  - What changes can you make to “prevent” that type of error
- Escapes
  - Phase error found in versus phase error introduced
  - Can measure errors that have escaped from the previous phase
  - Determine impact of escapes on quality



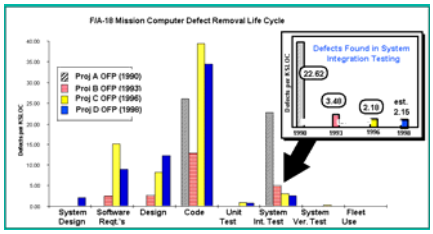


Key Elements of Software Production



Challenges

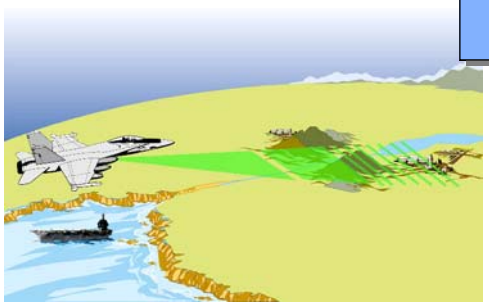
- Process Institutionalization
  - The people aspect is the most challenging
  - Have to earn their buy in and trust
    - They perceive metrics as more work –
    - The quality of the data is only as good as they give you
  - Have to demonstrate how their investment in providing quality data makes a difference
  - Many times charts/progress are only briefed up the chain to management – absolutely have to brief charts on a periodic basis to team to demonstrate
    - What's been measured
    - How changes have achieved goal
    - How changes have not achieved goal and new plan
  - Plan for resistance – but have a vision of where your measurement plan will lead your organization

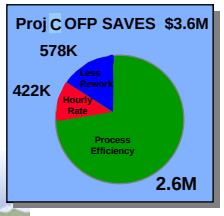


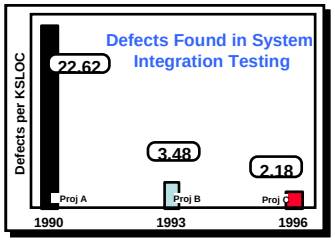
For example
Peer Reviews were added to the engineering life cycle. This was perceived by the software engineers as more work
Peer Reviews added to the lifecycle improved the quality of the product delivered to the AWL
The payoff came in less integration testing during D&D and V&V

This graphic illustrates how Peer Reviews impacted product quality at the end of the product lifecycle and convinced the engineers to embrace the Peer Reviews

Demonstrated Results





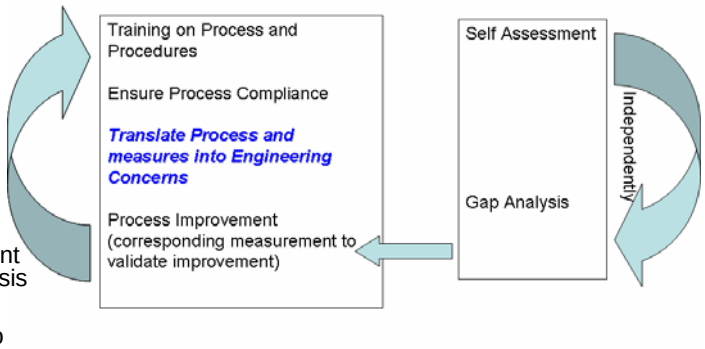


$3.6 Million Cost Savings

Moving Forward

Make process improvement cultural

- Five Steps
  - Education and Training
  - Process Compliance
  - Self-Assessment and Gap Analysis
  - Translation into Engineering Concerns
  - Improvement Initiatives



"Steps" must be concurrent
"Steps" must be sustained