AEW&C Software Metrics Program

PSM Users' Conference
Keystone, Colorado

14 - 18 July 2003

Stuart Garrett
AEW&C RPT Software Engineering Manager

Acknowledgment

- Data sourced from the Prime (Boeing)
- Boeing obtains the data from Software Development Team and Subcontractors
- Acquisition Project Office is a user - not generator - of the data

AEW&C Software Metrics Program - Introduction

- Metrics used on AEW&C Project
- Brief history of selection process
- Source of Data
- How metrics are used by Contractor
  - not “just another CDRL”
- How metrics are used within the CoA

Project Elements

737 AEW&C Aircraft

Mission Crew Simulator

Flight Crew Simulator

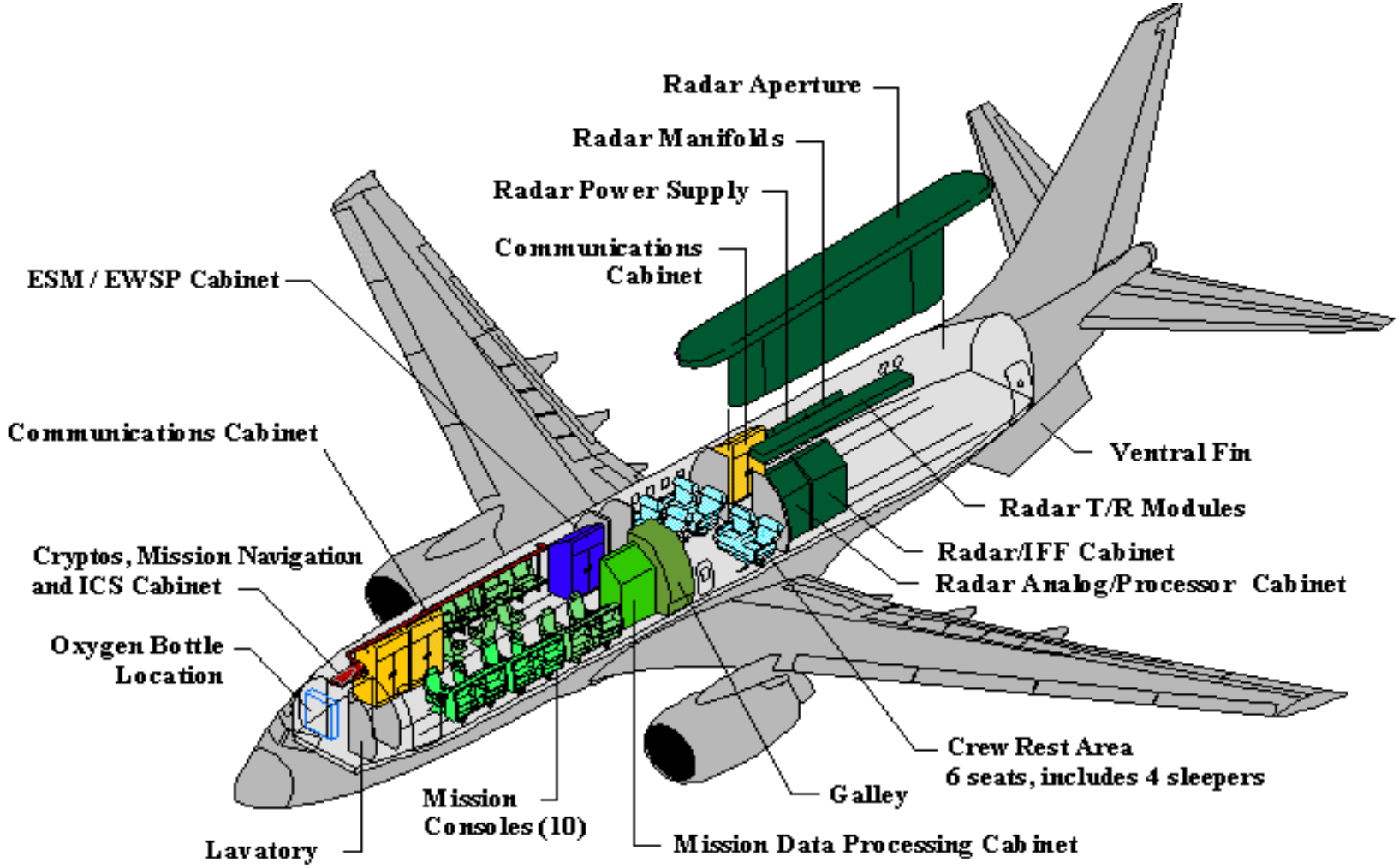
Mission Support

AEW&C Support Facility

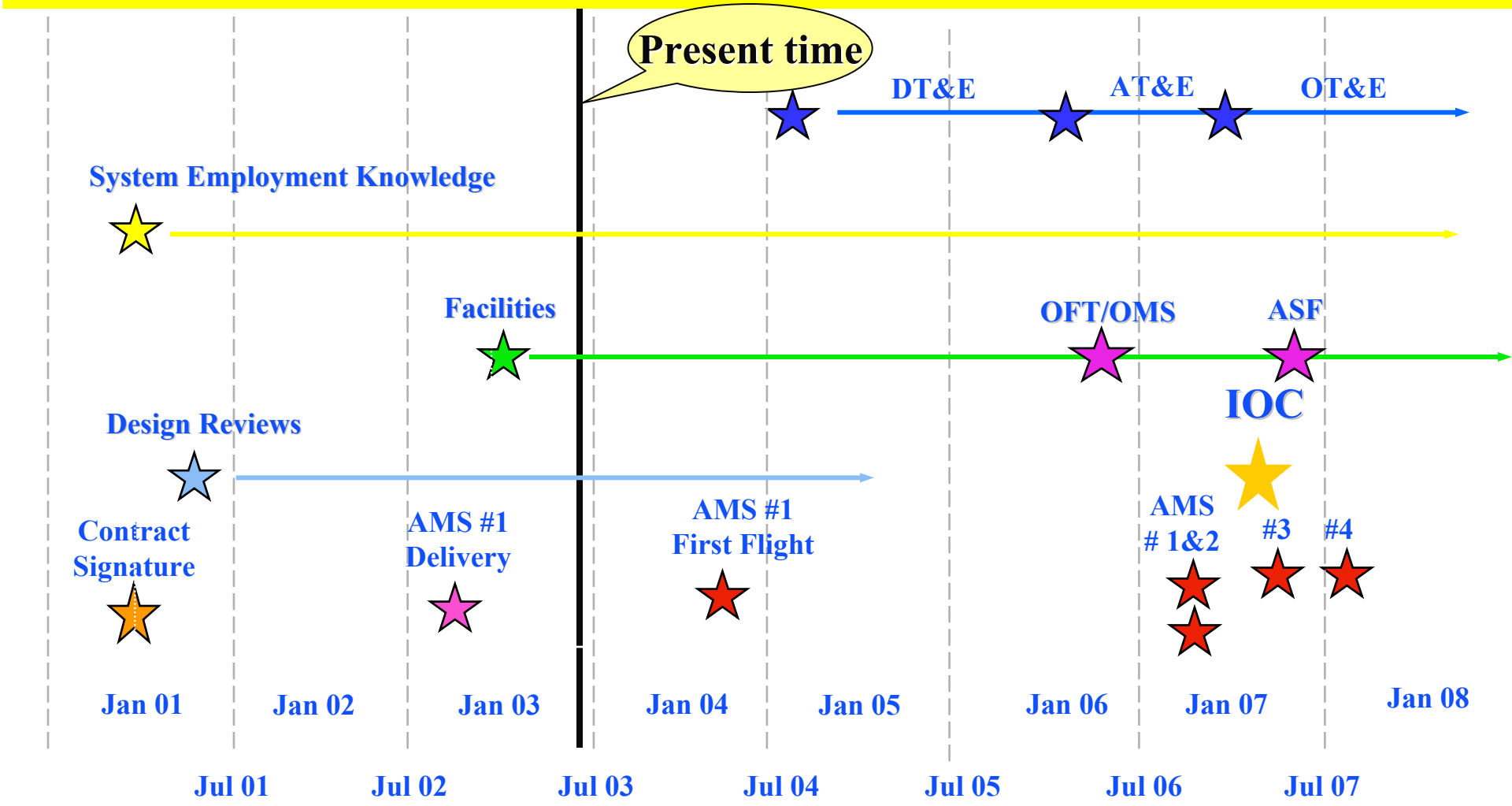
Integrated Logistics Support

Personnel and Training

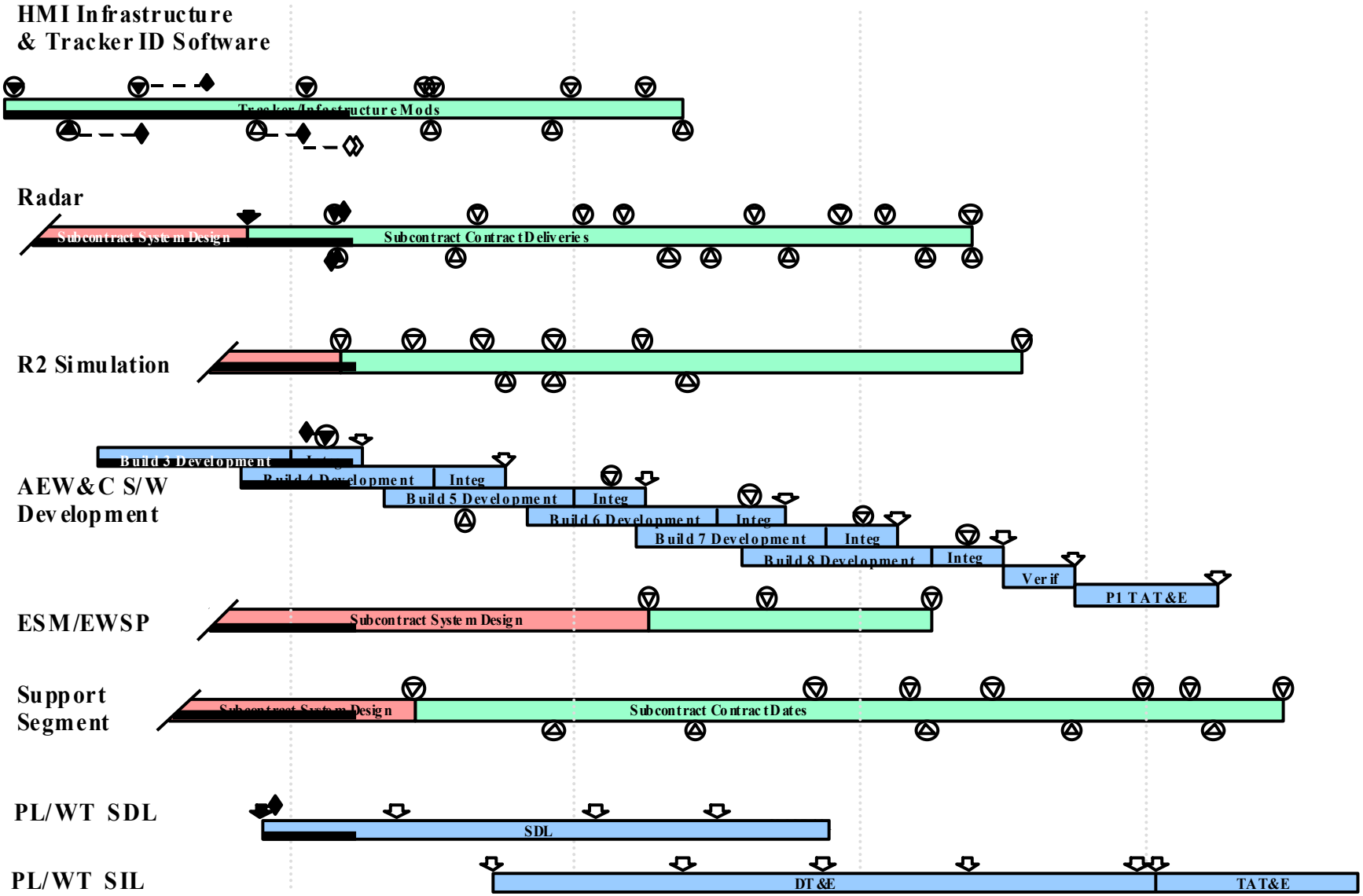
Airborne Mission Segment



Schedule - The Big Picture



Software Build Integration



Testing

- Desktop
- Software Development Laboratory
- System Integration Laboratory
- Aircraft on ground
- Aircraft in air

Software Metrics Program - What Type of Measurements Were Sought

- **Required measurements that could assist with:**
  - Early identification of problems
  - Evaluating the maturity of the software
  - Measuring progress of the development and test efforts
  - Evaluating software risk
  - Determining mitigation actions when required

Metrics - Selection Process

- Workshop with Facilitator
- Attended by Project Office, Prime (Boeing) and Major Subcontractors
  - NG, BAE Systems
- Objectives
  - Identify appropriate set of metrics
  - scope and define the metrics
  - facilitate buy-in and ownership with all parties

Software Metrics Program

Measures to monitor Maturity

Percent Activity Complete	Measures the progress of software development activities
Software Requirements Volatility	Measures requirements growth and maturity by tracking the number of requirements changes over time
Build Content	Measures the planned and actual progress of the incorporation of functionality for each software build release
Work Products	Measures the progress of discrete software products during development
Test Coverage	Measures the planned and actual progress of the results of testing for each software build release
Problem Reports	Measures the status, priority, effectivity, and resolution time of software change requests

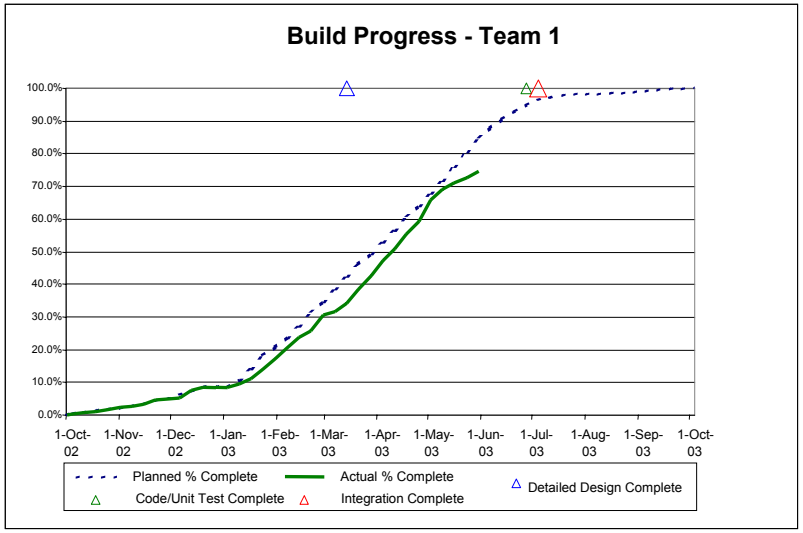
Software Metrics Program

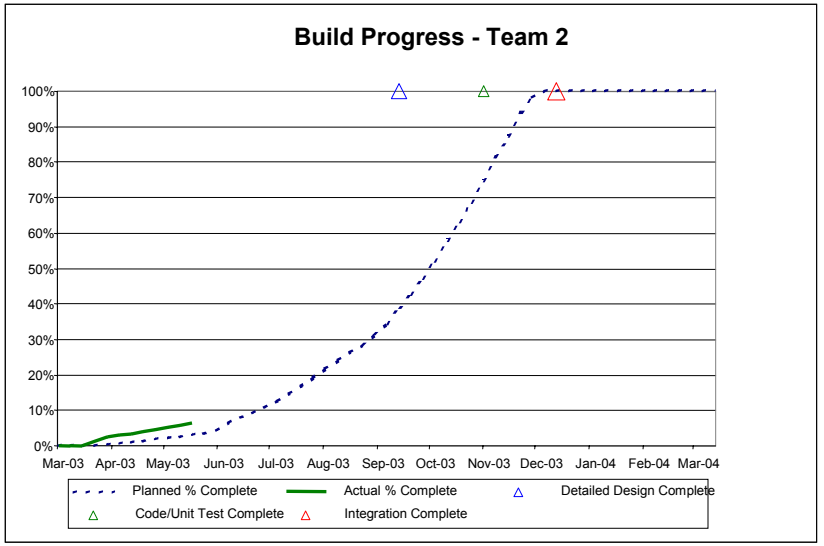
Measures for Schedule Performance and Risk

Software Product Size	Measures the planned and actual size of the software products
Data Deliverable Items	Measures the planned and actual, delivery and approval of contractually-required software data items
Staffing Profile	Measures the headcount, experience, and training for the software organizations
Computer Resource Utilization	Measures the planned and actual utilization of computing resources (CPU, storage, input/output)

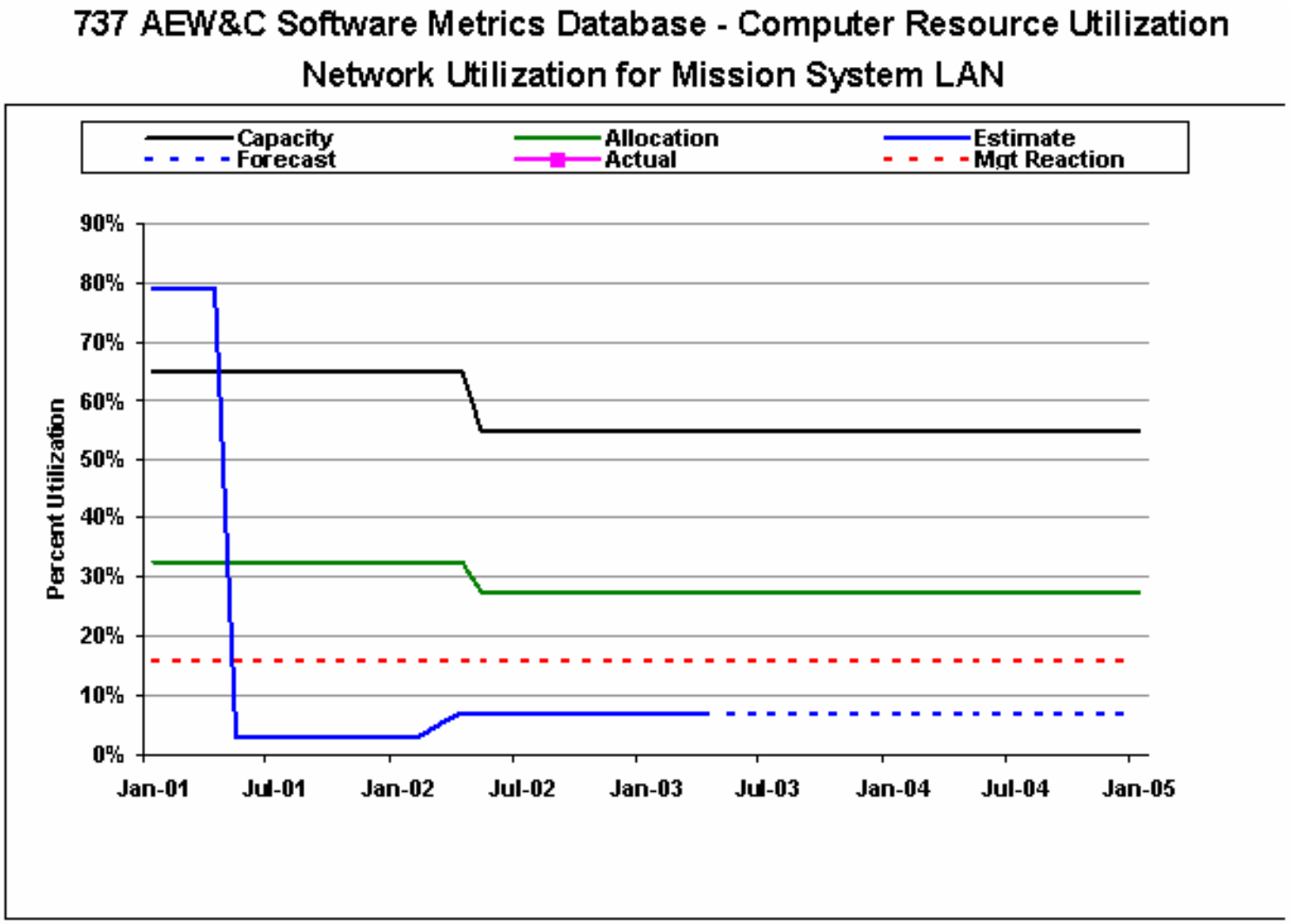
Management of Resources

- **Percentage Activity Complete**



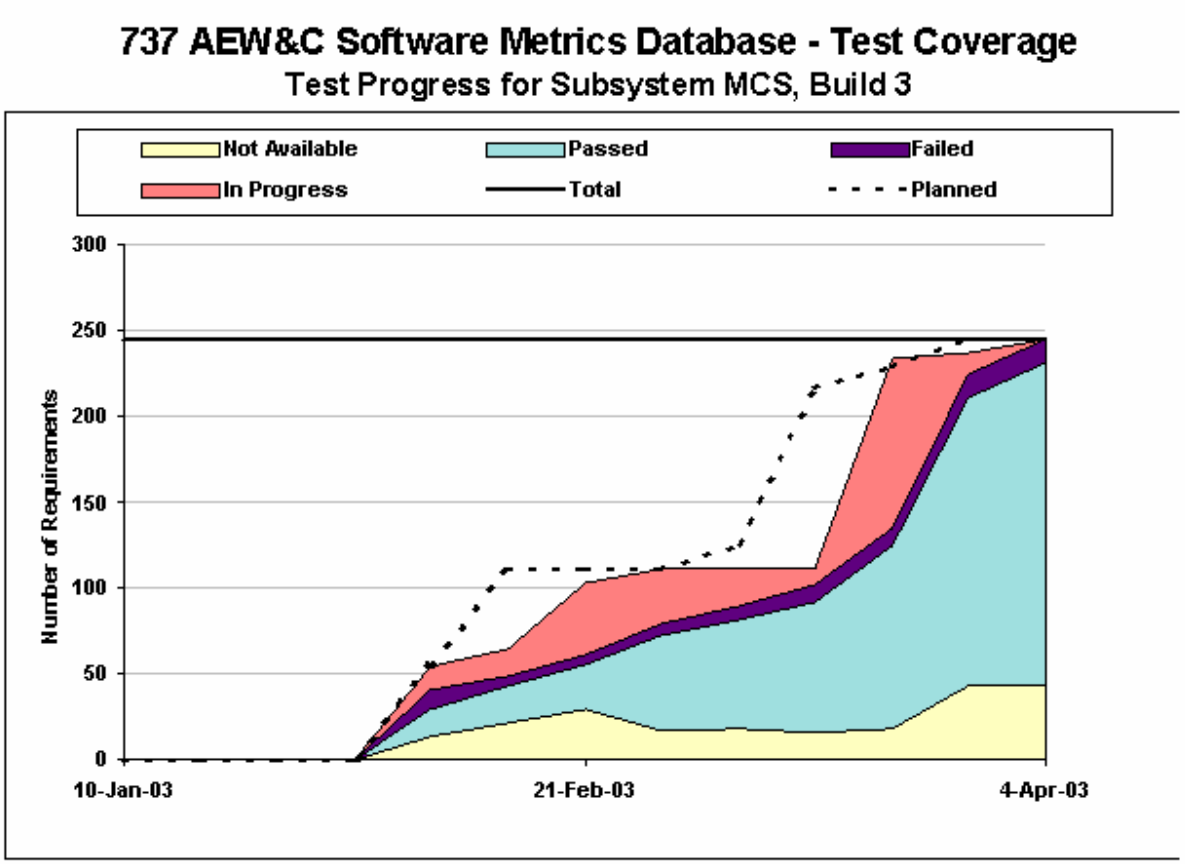


Risk Identification



Status Monitoring

- **Test Coverage**

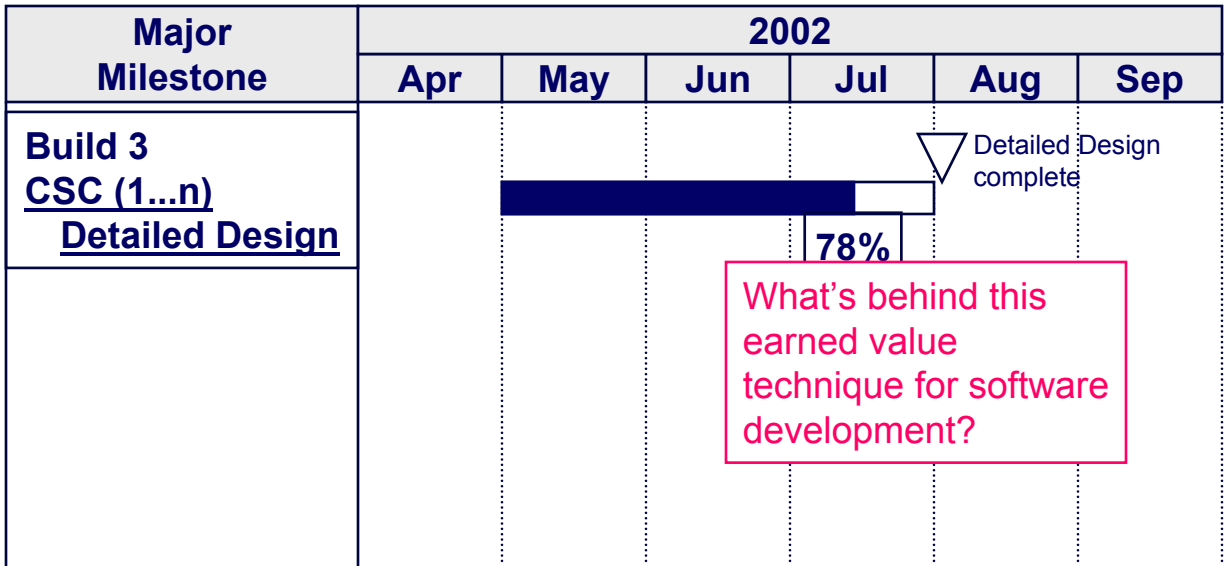


Status Reporting

- Software Metrics database provided each month to CoA Project Office
- On-line access to Boeing staff and Management
- Metrics summaries generated for weekly management reports
- Used as integral element of Risk Management System
- Feeds into technical and design reviews

Percentage Activity Complete -
Software Work Packages and Earned Value

- Software work packages are established at WBS level 4
  - Schedule, plan for completion, and status (work complete) tracked at lower (WBS 5) levels



Detailed Software Development Status Reporting

- **WBS level 4 status reported represents composite work package summary of software development activities**

- **In this example, WBS level 5 status reported is based on detailed design development for Software CSCs (Computer Software Components)**
  - **Plan for individual components established at start of build cycle.**
  - **Status reflects actual completion dates and/or % completion.**

Architecture Team CSC	Detailed Design Complete		
	Plan	Actual	% Complete
CSC 1	12-Jul-02	30-Jul-02	100%
CSC 2	12-Jul-02	30-Jul-02	100%
CSC 3	24-Jul-02		78%
CSC 4	22-Jul-02		0%
CSC 5	19-Jul-02	31-Jul-02	100%
CSC 6	19-Jul-02	23-Jul-02	100%
CSC 7	28-Jun-02	11-Jul-02	100%
CSC 8	19-Jul-02		97%
CSC 9	19-Jul-02	31-Jul-02	100%
CSC 10	18-Jul-02		42%
CSC 11	19-Jul-02		97%
CSC 12	25-Jul-02		40%
CSC 13	22-Jul-02	22-Jul-02	100%
CSC 14	25-Jul-02		50%
CSC 15	12-Jul-02	22-Jul-02	100%
CSC 16	1-Aug-02		81%
CSC 17	2-Aug-02		84%
CSC 18	9-Aug-02		9%
CSC 19	20-Jul-02	2-Aug-01	100%
Work Package Summary			78%

Reported as summary Tier IV Work Package status

Individual Developer Progress Status and Reporting

CSC:	CSC n Detailed Design	
Detailed Design Status	Completion date	% Activity Complete
Interfaces (25%)		25%
objects defined *	11-Jul-02	
types defined *	11-Jul-02	
exceptions defined *	11-Jul-02	
methods defined *	11-Jul-02	
impl classes (25%)		25%
implemented interface defined *	11-Jul-02	
other impl classes inherited *	11-Jul-02	
relationships between impls *	11-Jul-02	
Component design (40%)		40%
lead classes identified *	25-Jun-02	
periodic jobs named *	25-Jun-02	
sequence diagrams *	15-Jul-02	
Internal design defined	11-Jul-02	
MSDL data defined	31-Jul-02	
Checkpoint data defined	31-Jul-02	
Recorded data defined	31-Jul-02	
HMI defined	n/a	
algorithms defined	n/a	
Peer Review (10%)		10%
review package available	10-Jul-02	
Conduct peer review	11-Jul-02	
action items closed	11-Jul-02	
Detailed Design Complete (Plan)	**28-Jun-02**	
Detailed Design % Complete	**11-Jul-02**	100%

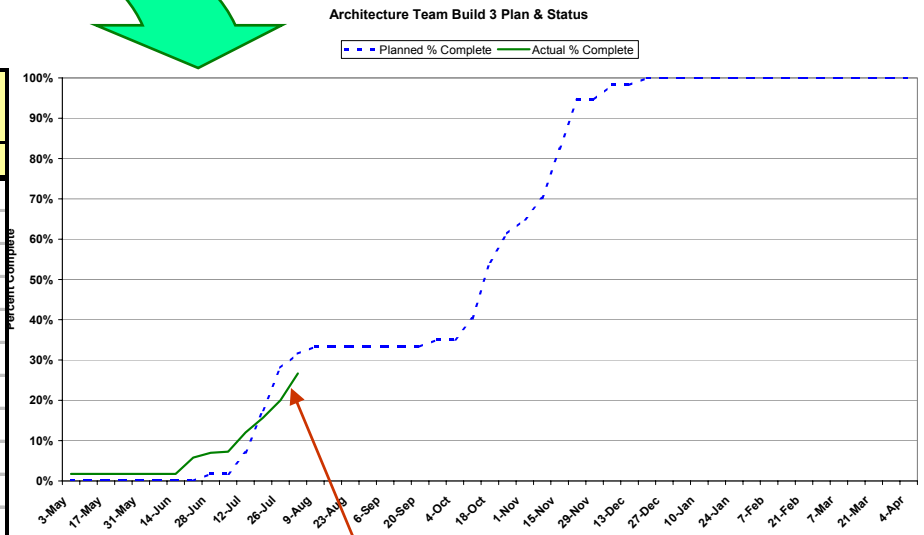
- **WBS level 5 plans generated based on detailed design activities for each Software Component**
  - **Plans and Status maintained by individual developers**
  - **Development Status updated weekly**
  - **CSC detailed design complete status is weighted summary of individual activities**

Architecture Team CSC	Detailed Design Complete		
	Plan	Actual	% Complete
CSC 1	12-Jul-02	30-Jul-02	100%
CSC 2	12-Jul-02	30-Jul-02	100%
CSC 3	24-Jul-02		78%
CSC 4	22-Jul-02		0%
CSC 5	19-Jul-02	31-Jul-02	100%
CSC 6	19-Jul-02	23-Jul-02	100%
CSC 7	28-Jun-02	11-Jul-02	100%
CSC 8	19-Jul-02		97%
CSC 9	19-Jul-02	31-Jul-02	100%
CSC 10	18-Jul-02		42%
CSC 11	19-Jul-02		97%
CSC 12	25-Jul-02		40%
CSC 13	22-Jul-02	22-Jul-02	100%
CSC 14	25-Jul-02		50%
CSC 15	12-Jul-02	22-Jul-02	100%
CSC 16	1-Aug-02		81%
CSC 17	2-Aug-02		84%
CSC 18	9-Aug-02		9%
CSC 19	20-Jul-02	2-Aug-01	100%
Work Package Summary			78%

Summarized Team Metrics

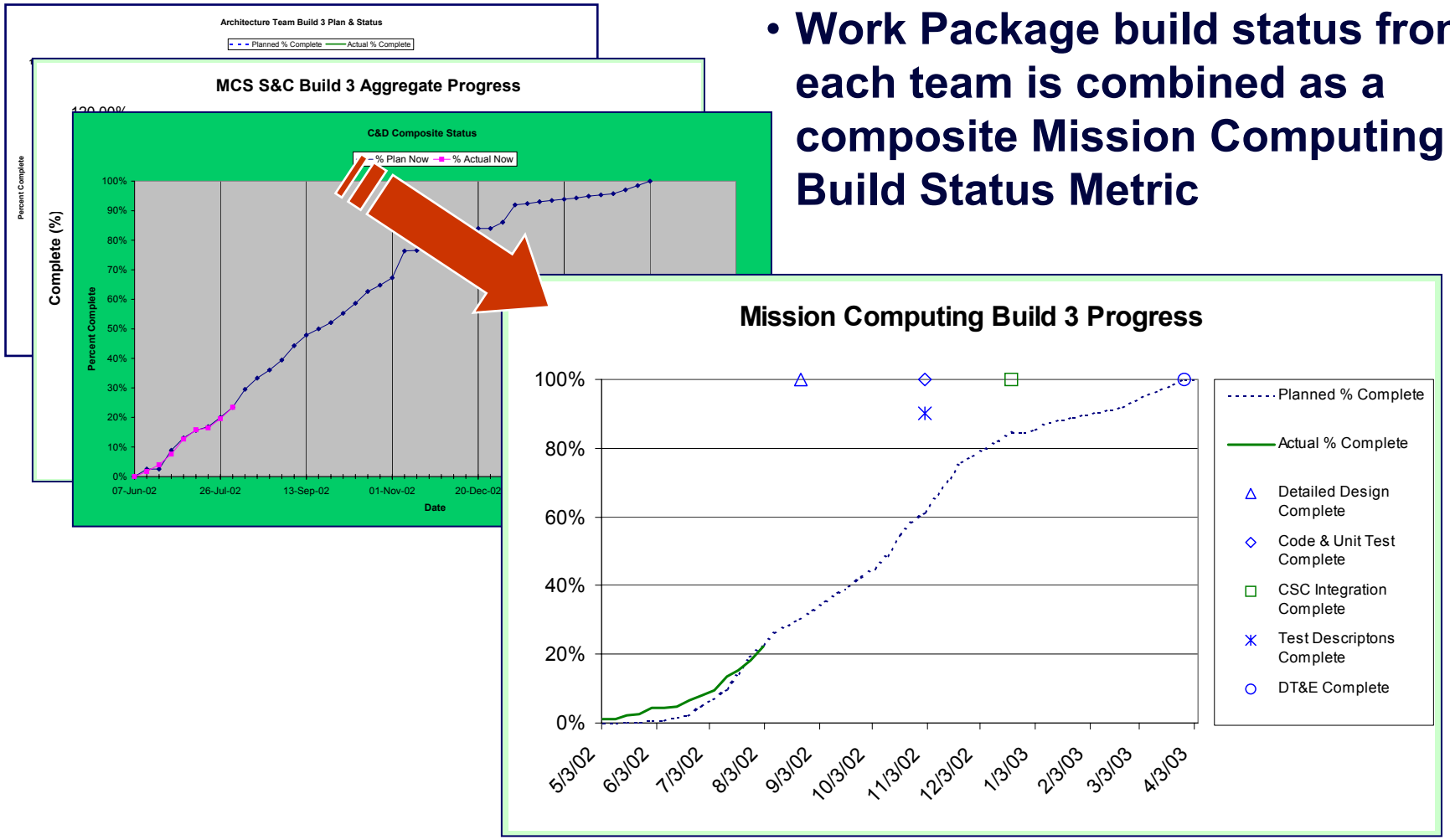
- **Work package summary plans & status combined and reported weekly in Build metric as % of build complete**

Architecture Team CSC	Detailed Design Complete		
	Plan	Actual	% Complete
CSC 1	12-Jul-02	30-Jul-02	100%
CSC 2	12-Jul-02	30-Jul-02	100%
CSC 3	24-Jul-02		78%
CSC 4	22-Jul-02		0%
CSC 5	19-Jul-02	31-Jul-02	100%
CSC 6	19-Jul-02	23-Jul-02	100%
CSC 7	28-Jun-02	11-Jul-02	100%
CSC 8	19-Jul-02		97%
CSC 9	19-Jul-02	31-Jul-02	100%
CSC 10	18-Jul-02		42%
CSC 11	19-Jul-02		97%
CSC 12	25-Jul-02		40%
CSC 13	22-Jul-02	22-Jul-02	100%
CSC 14	25-Jul-02		50%
CSC 15	12-Jul-02	22-Jul-02	100%
CSC 16	1-Aug-02		81%
CSC 17	2-Aug-02		84%
CSC 18	9-Aug-02		9%
CSC 19	20-Jul-02	2-Aug-01	100%
Work Package Summary			78%



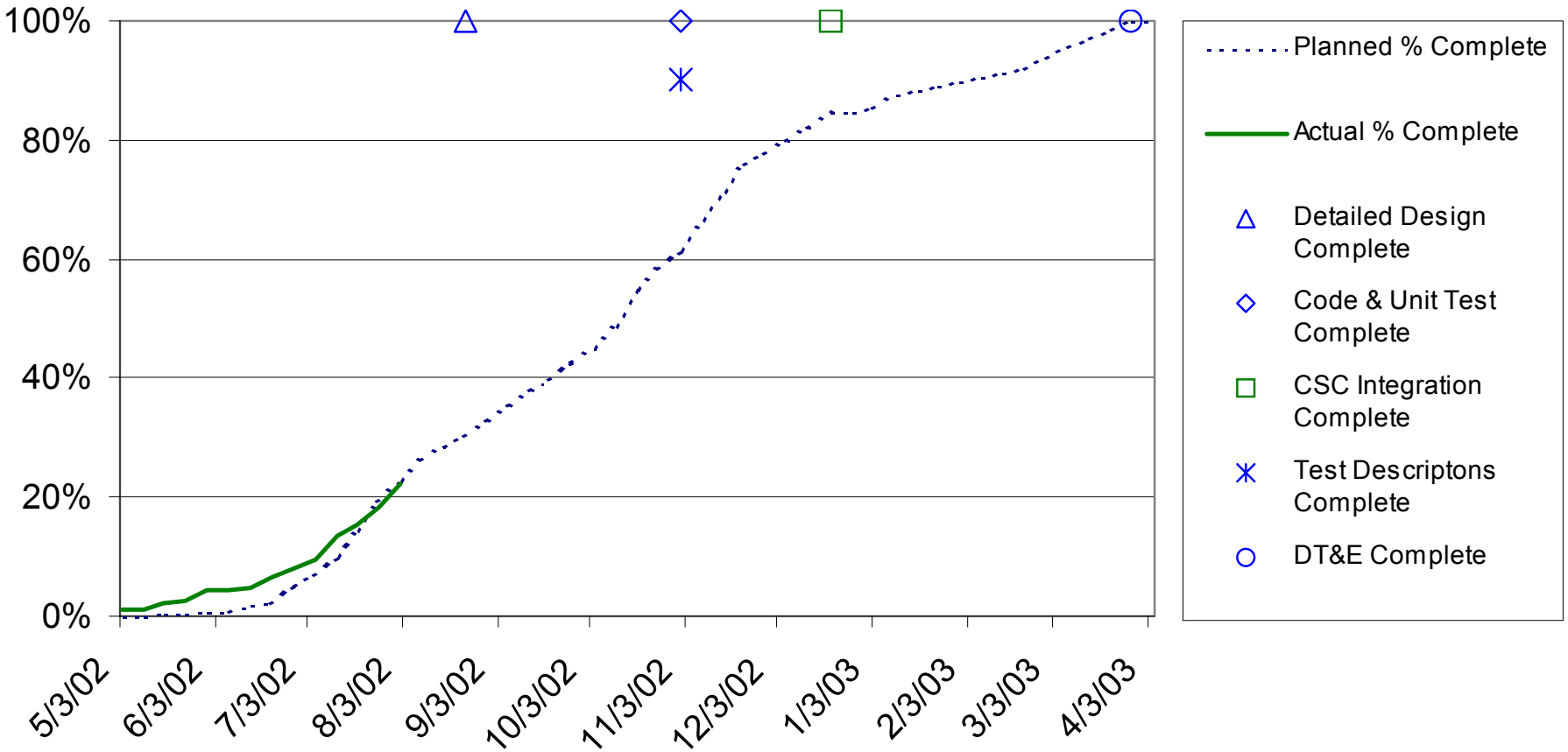
Detailed design complete status is weighted portion of overall build 3 development status for this team

Build Summarized Metrics

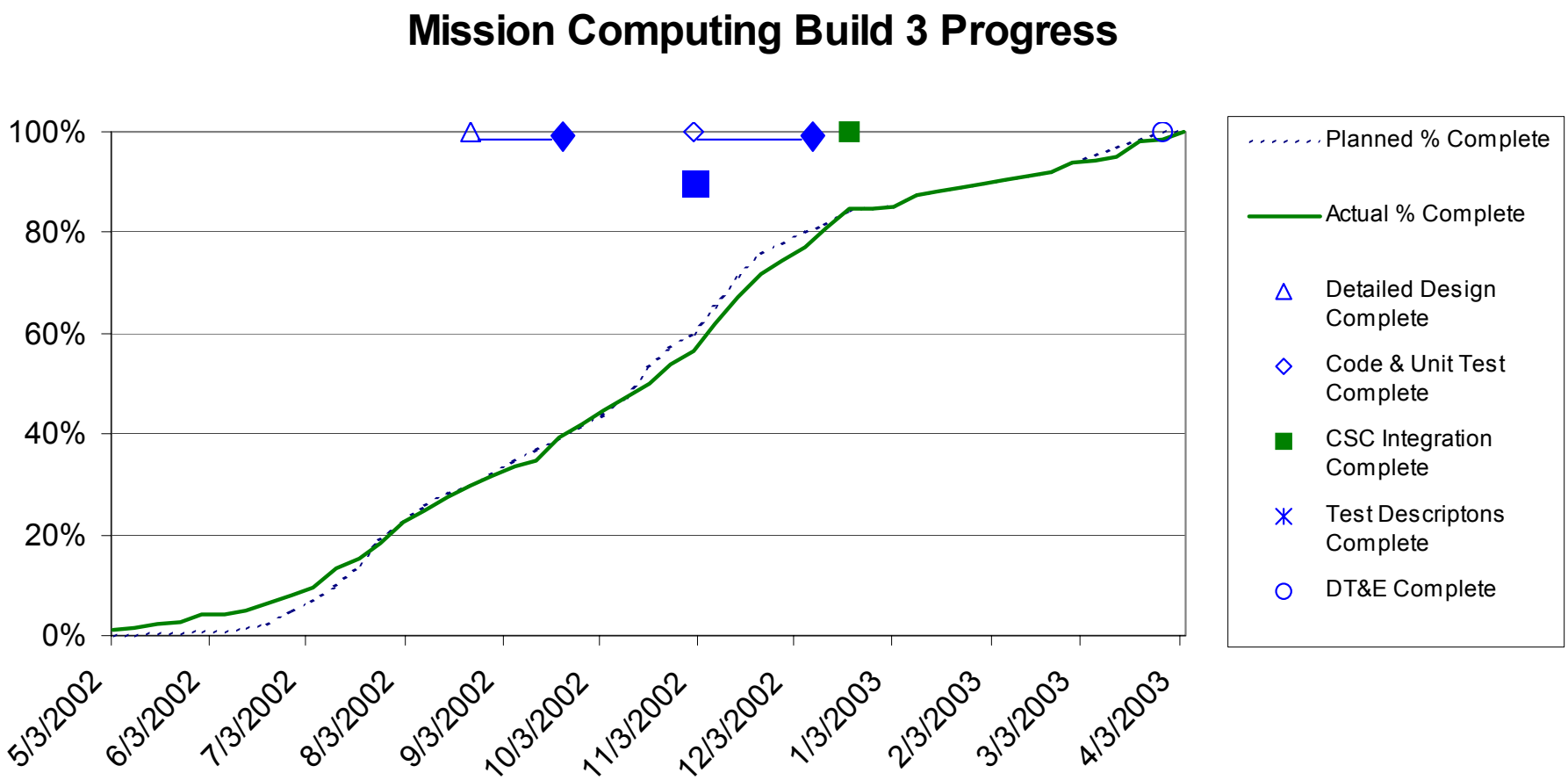


- **Work Package build status from each team is combined as a composite Mission Computing Build Status Metric**

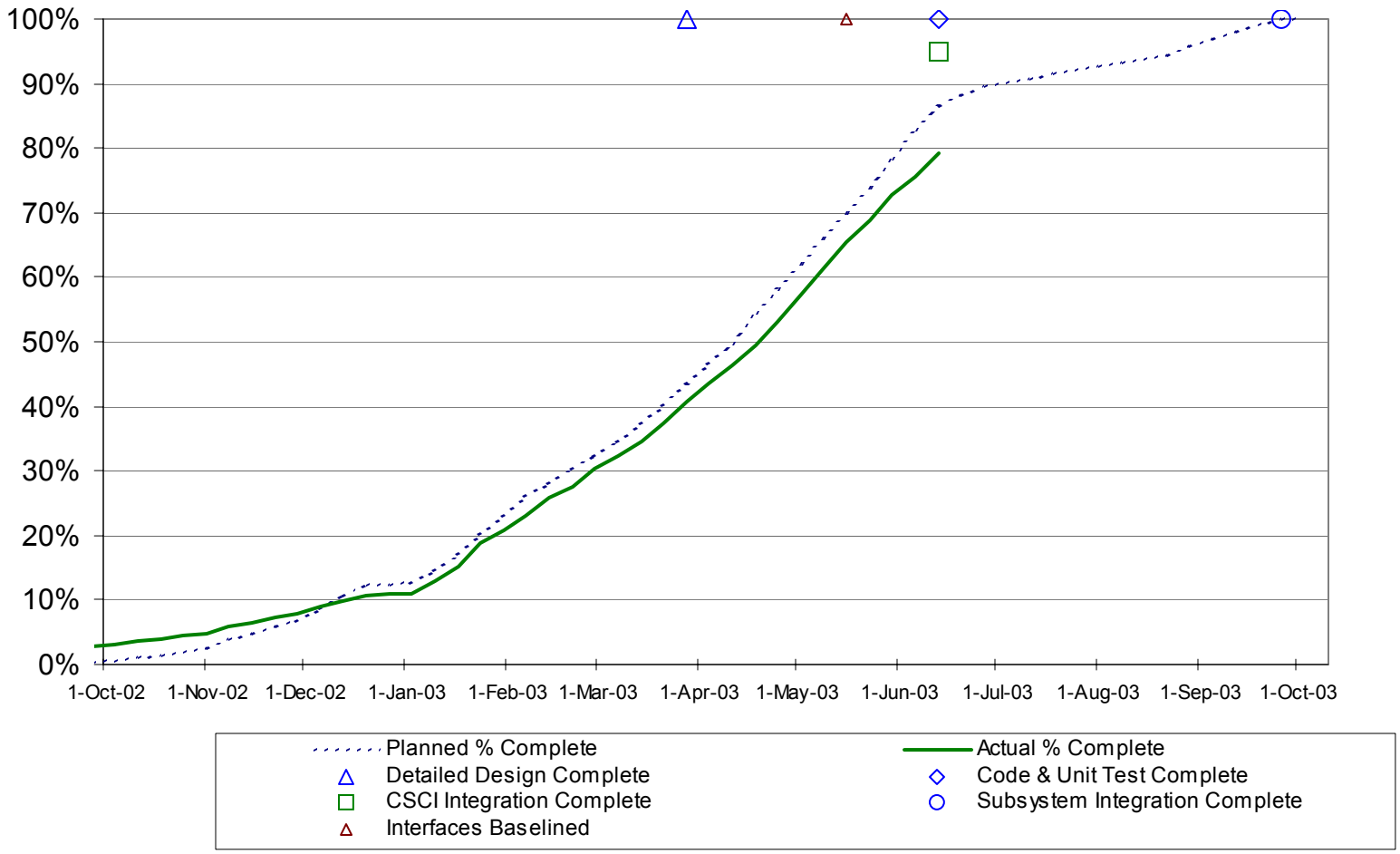
Mission Computing Build 3 Progress



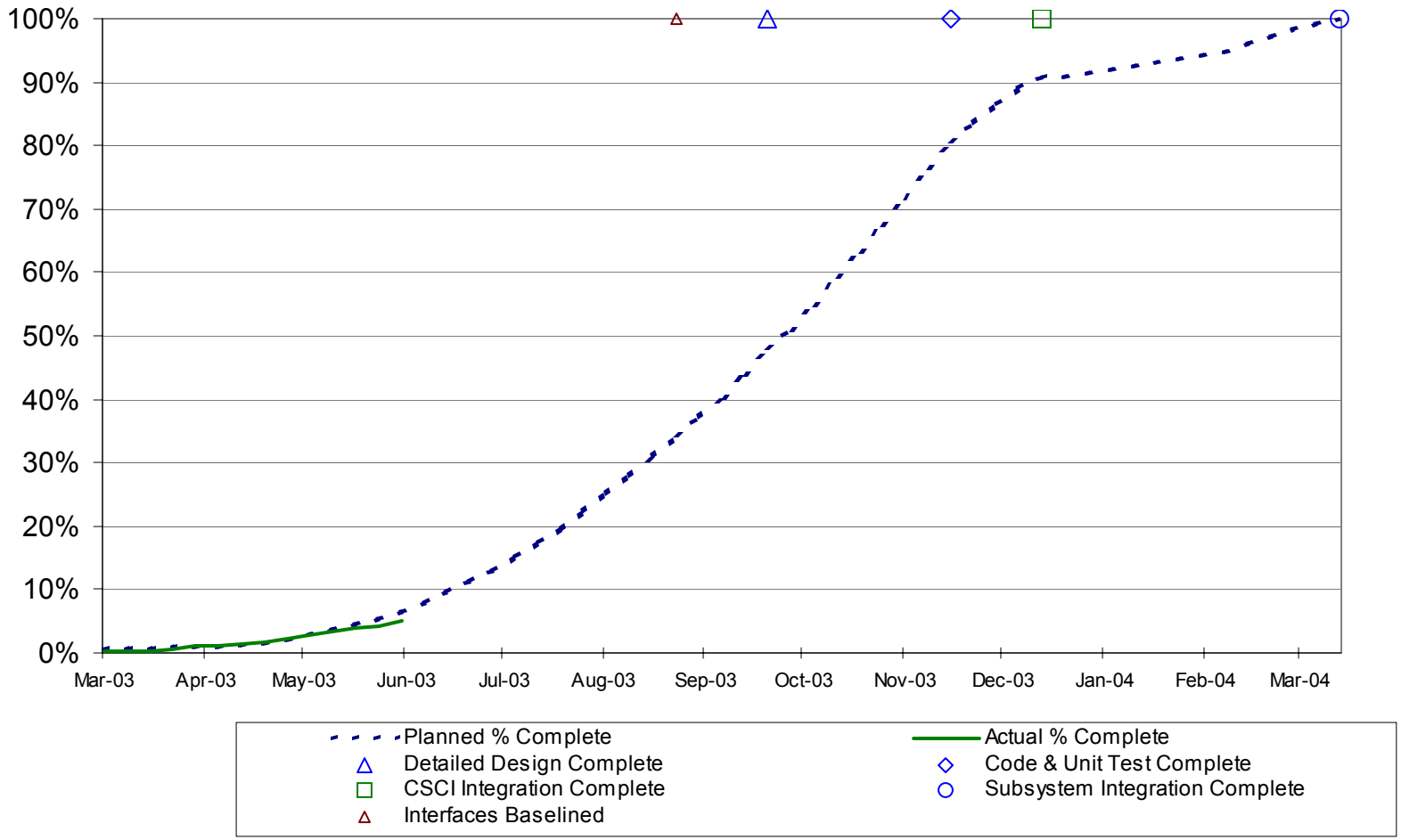
Actual Progress



Mission Computing Build 4 Progress



Mission Computing Build 5 Progress



Brisbane - Australia

Typical evening in Brisbane

Day at the office - Australian style

Use of Metrics within CoA

- Provides insight into software development process
- Confirmation of understandings from meetings, documentation, reviews
- Insight into hardware performance
- Insight into maturity of design
- Understanding of effectiveness and status of recovery plans

Use of Metrics within CoA

- Metrics reviewed in toto each month
- Analysing for unexpected trends, or unexplained spikes (degradation)
- Seeking confirmation of information obtained from design, management and technical reviews
- Findings are consolidated into a monthly report to the Program Governance Board

Value of Metrics Program

- Unprecedented insight into software development activities
- Invaluable tool for management to make informed decisions re resources
- Difficult, if not impossible, to successfully manage a large complex software intensive project without an effective metrics program

Summary

- Metrics provide specific insight into software development activities
- Allows managers to assess progress
- Identify problems - before they become unmanageable
- Plan (*timely*) solutions
- Obtain additional resources, as required
- Monitor the effectiveness of remedial actions
- Reduce overall risk to the program - excellent Risk Management / Reduction Strategy

Questions?