Lessons Learned for Complex Software
Intensive Joint Navy Programs

Bob Hunt
Vice President, Services
bhunt@galorath.com
703-201-0651
Galaroth: Making Plans into Reality

Over 25 Years Solving Project Management, Scheduling, & Resource Analysis Problems For Government & Industry
- U.S. HQ in El Segundo, CA
- International HQ in Farnham, Surrey U.K.
- Washington DC, Phoenix, Melbourne, Raleigh, Saint Louis

Developers of the SEER Product Suite

Experts In Resource Analysis, Cost Analysis, EVM, & Program Tradeoff Studies (Our staff represents decades of experience in hardware and software analysis)
- Life Cycle Cost, Schedule, Risk, and Reliability Estimation at any stage Of The Life Cycle
- Cost/Schedule Tradeoffs, CAIV, Design To Cost Analyses
- Design for Manufacturing
- Project Monitoring and Control (Earned Value Management)
- Proposal Support - Including Strategy & Cost/Price Analysis
- Acquisition Cost Management
- Cost Process Improvement
- Custom Model Development & Implementation

Joint Programs require a robust mixture of cost and program skills

© 2008 Copyright Galaroth Incorporated
SEER Products Provide Complete
“Total Cost of Ownership Solutions”
An Estimate is a Distribution – Not a Point

• An *estimate* is the most knowledgeable statement you can make *at a particular point in time* regarding:
  - Effort / Cost
  - Schedule
  - Staffing
  - Risk
  - Reliability

• A well formed estimate is a distribution
• A well structured plan defines probability

Joint Programs need cost ranges and risk management
People, Process, Technology Are Keys
Source CMMI Tutorial

• Everyone realizes the importance of having a motivated, quality work force but...
• ...even our finest people can’t perform at their best when the process is not understood or operating “at its best.”

Major determinants of product cost, schedule, and quality
10 Step Software Estimation Process:
Consistent Processes = Reliable Estimates

1. Establish Estimate Scope
2. Establish Technical Baseline, Ground Rules, Assumptions
3. Collect Data
4. Estimate and Validate Software Size
5. Prepare Baseline Estimates
6. Review, Verify and Validate Estimate
7. Quantify Risks and Risk Analysis
8. Generate a Project Plan
9. Document Estimate and Lessons Learned
10. Track Project Throughout Development

© 2008 Copyright Galorath Incorporated
Mil/Aero Business Space

• General move to net-centric/systems-of-systems architectures
  – CANES, FCS, …

• Emphasis on joint, interoperable programs (between Services and nations)
  – JSF, JPALS, …

• More reliance on the contractor to lead efforts (LSI)?

• A great deal of talk about new acquisition strategies and developmental paradigms; and their potential reduction in costs – Agile, Use Cases

• Volatility in requirements and architecture – the more stakeholders there are the more volatility must be managed/expected

• Continued emphasis on process

• Increased use of COTS and open source products (SaaS and SOA)
## DoD Versus Commercial Development*

<table>
<thead>
<tr>
<th></th>
<th>Aerospace</th>
<th>Commercial</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Driving force</strong></td>
<td>Ability to support the warfighter</td>
<td>Ability to penetrate markets and make money</td>
</tr>
<tr>
<td><strong>Mindset</strong></td>
<td>Pessimistic and risk adverse</td>
<td>Optimistic and willing to take risk if rewards high</td>
</tr>
<tr>
<td><strong>Primary focus</strong></td>
<td>Customer satisfaction and cost control</td>
<td>Customer satisfaction and market dominance</td>
</tr>
<tr>
<td><strong>Process used</strong></td>
<td>Acquisition mindset with lots of oversight/discipline</td>
<td>R&amp;D mindset with some oversight/ discipline</td>
</tr>
<tr>
<td><strong>Driving issues</strong></td>
<td>• Understanding what is really wanted</td>
<td>• Being nimble, but in control</td>
</tr>
<tr>
<td></td>
<td>• Bureaucracy and overkill</td>
<td>• Devising discriminators</td>
</tr>
</tbody>
</table>

Taken from “Software Cost/Productivity/Quality: Have We Made Any Progress in the Past Decade?”; Donald J. Reifer; Reifer Consultants, Inc., May 2008

© 2008 Copyright Galorath Incorporated
## U.S. AVERAGE COSTS PER FUNCTION POINT IN 2008

<table>
<thead>
<tr>
<th>Software Type</th>
<th>Unburdened</th>
<th>Fully Burdened</th>
</tr>
</thead>
<tbody>
<tr>
<td>End-user Software</td>
<td>$350</td>
<td>$500</td>
</tr>
<tr>
<td>Web Software</td>
<td>$450</td>
<td>$800</td>
</tr>
<tr>
<td>Information Systems Software</td>
<td>$750</td>
<td>$1,100</td>
</tr>
<tr>
<td>Outsource Software</td>
<td>$700</td>
<td>$1,500</td>
</tr>
<tr>
<td>Commercial Software</td>
<td>$1,100</td>
<td>$1,800</td>
</tr>
<tr>
<td>Systems Software</td>
<td>$1,250</td>
<td>$2,100</td>
</tr>
<tr>
<td><strong>Military Software</strong></td>
<td><strong>$2,500</strong></td>
<td><strong>$5,000</strong></td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td><strong>$1,014</strong></td>
<td><strong>$1,829</strong></td>
</tr>
</tbody>
</table>

From MEASUREMENT, METRICS AND INDUSTRY LEADERSHIP; Capers Jones, Chief Scientist Emeritus; Software Productivity Research LLC, January 30, 2008
## CREEPING REQUIREMENTS IN 2008

<table>
<thead>
<tr>
<th>Domain</th>
<th>Average Monthly Rate of Creeping Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web software projects</td>
<td>4.0%</td>
</tr>
<tr>
<td>Commercial Software</td>
<td>3.5%</td>
</tr>
<tr>
<td>Information technology</td>
<td>2.5%</td>
</tr>
<tr>
<td>System, embedded software</td>
<td>2.0%</td>
</tr>
<tr>
<td><strong>Military Software</strong></td>
<td><strong>2.0%</strong></td>
</tr>
<tr>
<td>Outsourced Software</td>
<td>1.5%</td>
</tr>
<tr>
<td><strong>AVERAGE</strong></td>
<td><strong>2.6%</strong></td>
</tr>
</tbody>
</table>

From MEASUREMENT, METRICS AND INDUSTRY LEADERSHIP; Capers Jones, Chief Scientist Emeritus; Software Productivity Research LLC, January 30, 2008
Lessons Learned/Re-learned in the Last 12 Months

1. People are still people - software development is a very collaborative human process
2. Optimism is still the rule
3. Advances in developmental processes, languages, and tools are working to increase efficiencies in software development, but increases in requirements (system sophistication) are masking these efficiencies
4. While there is more talk about Software maintenance, it is still not a major acquisition consideration
5. System advocates are interested in more than just software development costs; the total system acquisition costs is critical (SEER for IT)
6. Cost estimates are entropic – they grow to be as large as available funding and seldom as large as they need to be
7. It is difficult to get people to invest in training; but the investment pays off
8. Failure to track actuals and record project/program data for future calibration is a massive lost opportunity

9. Even though people say "we've learned from the past and won't repeat those mistakes again" they usually do have the same overruns, but for different reasons

10. Cost savings from reused code is rarely realized to the extent planned – more to come on SOA and SaaS

11. Productivity gains from new techniques and automated tools are always overstated, and requirements volatility is always understated

12. People are as thorough and detailed as much as time allows them to be, not as much as their process says they should be

13. Independent checks (by another person) are more valuable than multiple rechecks by the person who created the estimate
Joint Programs

• Is it a Weapon Systems, C4ISR System, or Major AIS – probably some of each

• The important point is that the joint PM must be the central manager for all of the participating Components – control requirements growth – track cost growth

• Views of former joint PMs (Joint PM Handbook):
  ▪ A major cost driver is the inability to make decisions on joint requirements
  ▪ Contract problems can be traced back to technical issues and related to the ability to meet the requirements levied upon the system
  ▪ In development of the Capability Development Document, 50 percent of the time is spent with users discussing trade-offs
  ▪ Interoperability is the number one concern among all military Components/Services

© 2008 Copyright Galorath Incorporated
Specifics Issues for Cost Management of Joint Programs

- Develop an Acquisition Program Baseline (APB) of cost, schedule, and performance (capability) parameters – consider software specific metrics, e.g., defect discovery and removal
- Use a parametric model to track and monitor these cost, schedule, and performance parameters
- Define (or as a minimum document) who can create changes
## Tracking A Joint Navy Program
### Top 10 Cost Drivers

<table>
<thead>
<tr>
<th>Rank</th>
<th>First Estimate</th>
<th>Second Estimate (14 months later)</th>
<th>Third Estimate (7 months Later)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Specification level - Reliability</td>
<td>Specification</td>
<td>Specification level - Reliability</td>
</tr>
<tr>
<td>2</td>
<td>Test Level</td>
<td>Test</td>
<td>Test Level</td>
</tr>
<tr>
<td>3</td>
<td>Time Constraints</td>
<td>Requirements Volatility</td>
<td>Requirements Volatility (Change)</td>
</tr>
<tr>
<td>4</td>
<td>Real Time Code</td>
<td><strong>Time Constraints</strong></td>
<td><strong>Time Constraints</strong></td>
</tr>
<tr>
<td>5</td>
<td>Special Display Requirements</td>
<td>Real Time Code</td>
<td>Real Time Code</td>
</tr>
<tr>
<td>6</td>
<td>Quality Assurance Level</td>
<td>Special Display Requirements</td>
<td>Special Display Requirements</td>
</tr>
<tr>
<td>7</td>
<td>Development System Volatility</td>
<td>Development System Volatility</td>
<td>Quality Assurance Level</td>
</tr>
<tr>
<td>8</td>
<td>Security Requirements</td>
<td>Security Requirements</td>
<td>Development System Volatility</td>
</tr>
<tr>
<td>9</td>
<td>Target System Volatility</td>
<td>Target System Volatility</td>
<td>Security Requirements</td>
</tr>
<tr>
<td>10</td>
<td>Process Volatility</td>
<td>Quality Assurance Level</td>
<td>Target System Volatility</td>
</tr>
</tbody>
</table>
An Acquisition Strategy to Reduce Software Maintenance Costs

- Software maintenance is often under estimated
- JSF is employing a unique strategy – the software developer must also be responsible for the maintenance – a fixed price bid
- Software Maintenance is “carefully” being defined and estimated
- It is unlikely that a LOE maintenance support strategy will be employed
What Keeps a Joint PM Awake at Night

Did we estimate the effort correctly?

Will my team deliver on time & budget?

How should I monitor progress?

Do we understand the requirements?

How do I justify a Change Request estimate?

Solution (partial)....

© 2008 Copyright Galorath Incorporated
Estimate Early & Often

- Use Parametric Estimation
- Validate with Engineering Build-up (WBS) estimate
- Calibrate estimates based on actual costs
- Re-estimate at end of each Phase & Iteration
- Use estimates to identify productivity improvement opportunities

“A good estimate is an estimate that provides a clear enough view of the project reality to allow the project leadership to make good decisions about how to control the project to hit its targets”

Steve McConnell
The Good, the Bad and the Ugly

- Parametric estimation provides consistency
- Less bias – “facts on the table”
- Requires organizational training & discipline

“Software estimation is neither hard nor new. What is hard, is accepting that the easy-to-digest answer we seek when estimating is simply not there...

Using these [estimation] tools turns the practice of estimation ... that engrains a disciplined approach and pays heed to underlying behavioral and attitude challenges, into an exercise that is simply about playing with numbers”

Final Things to Remember

• Processes and tools don’t provide the thinking; you do
• Communication is key; good management and poor communication are mutually exclusive
• Document every assumption and decision
• Find the balance between necessary detail and unnecessary complexity
• Size is the biggest driver of every software estimate; it deserves the most attention (expect size growth)
• Save every project baseline in order to Track Program Changes over time
• Hold a post-mortem project review; data is highly perishable
• A well developed WBS is critical to a well structured estimates and to crosswalk back to the requirement
• Expect change
Contact Information

Please contact us at:

info@galorath.com
(310) 414-3222 US
Bob Hunt 703-201-0651
Bhunt@Galorath.com
+44 (0) 1252 720 629 UK
www.galorath.com