

# estimate

estimate • analyze • plan • control

## Lessons Learned for Complex Software Intensive Joint Navy Programs

Bob Hunt  
Vice President, Services  
[bhunt@galorath.com](mailto:bhunt@galorath.com)  
703-201-0651



# Galorath: 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

# SEER Products Provide Complete "Total Cost of Ownership Solutions"

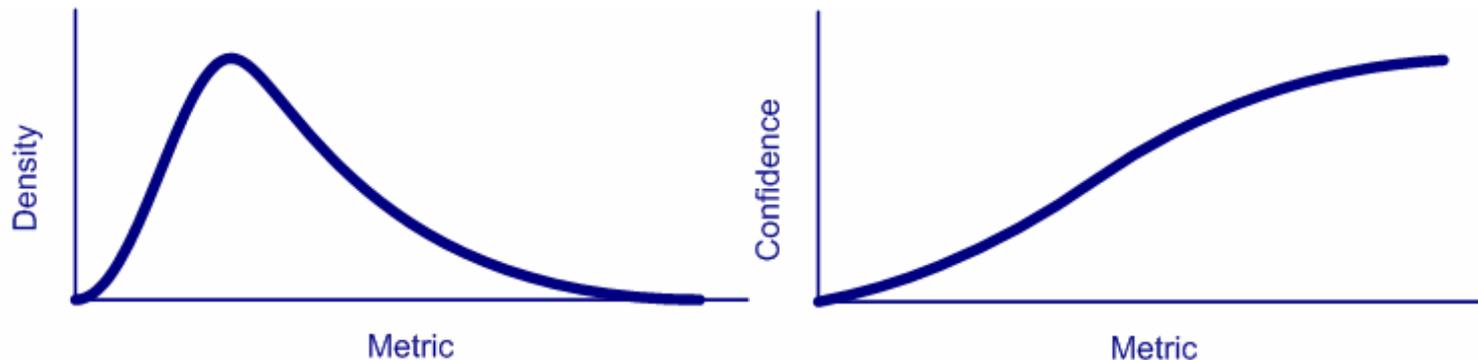


# An Estimate is a Distribution – Not a Point



- 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**

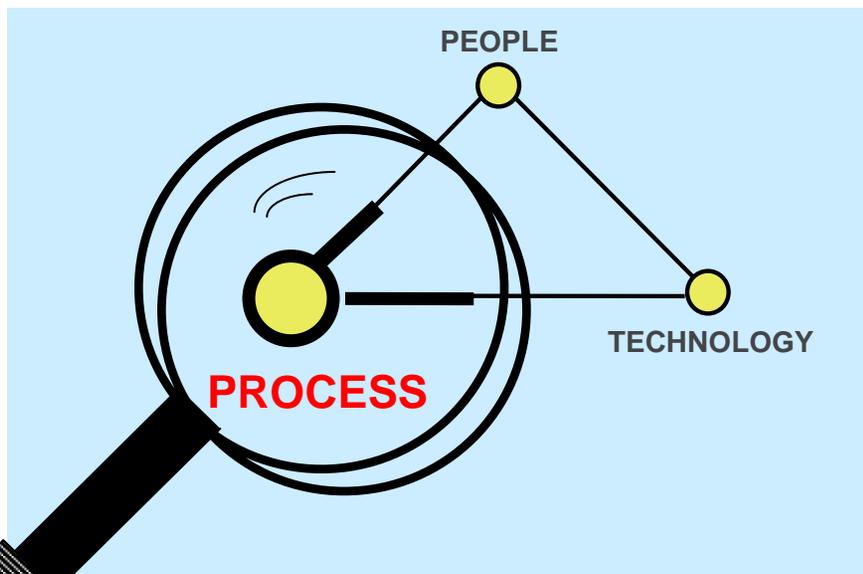


# People, Process, Technology Are Keys

Source CMMI Tutorial

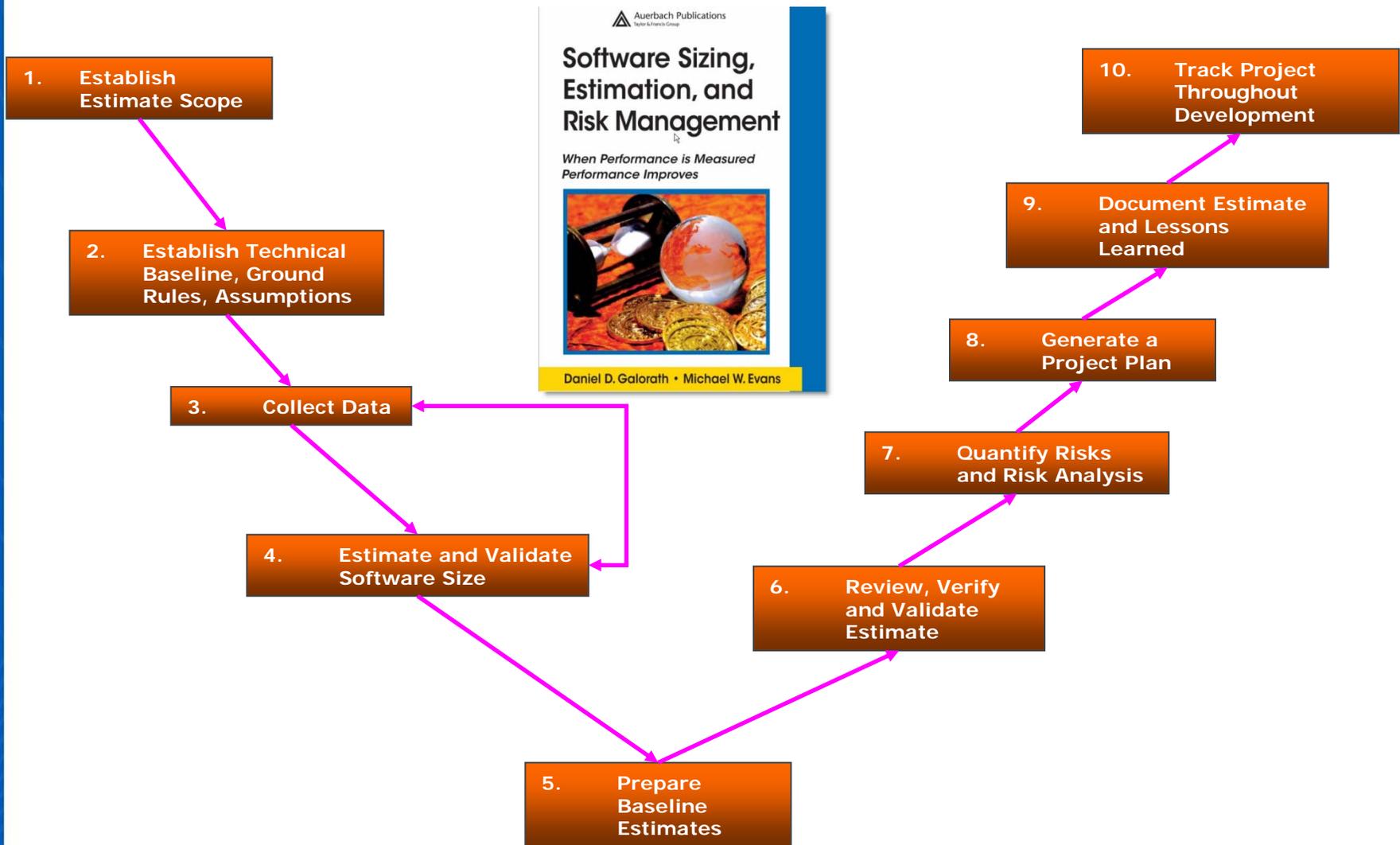


- Everyone realizes the importance of having a motivated, quality work force but...
- ...even our finest people can't perform at their best when the process is not understood or operating "at its best."



**Major determinants of product cost,  
schedule, and quality**

# 10 Step Software Estimation Process: Consistent Processes = Reliable Estimates



# 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\*



	<b>Aerospace</b>	<b>Commercial</b>
Driving force	Ability to support the warfighter	Ability to penetrate markets and make money
Mindset	Pessimistic and risk adverse	Optimistic and willing to take risk if rewards high
Primary focus	Customer satisfaction and cost control	Customer satisfaction and market dominance
Process used	Acquisition mindset with lots of oversight/discipline	R&D mindset with some oversight/ discipline
Driving issues	<ul style="list-style-type: none"><li>• Understanding what is really wanted</li><li>• Bureaucracy and overkill</li></ul>	<ul style="list-style-type: none"><li>• Being nimble, but in control</li><li>• Devising discriminators</li></ul>

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

## U.S. AVERAGE COSTS PER FUNCTION POINT IN 2008



	Unburdened	Fully Burdened
End-user Software	\$350	\$500
Web Software	\$450	\$800
Information Systems Software	\$750	\$1,100
Outsource Software	\$700	\$1,500
Commercial Software	\$1,100	\$1,800
Systems Software	\$1,250	\$2,100
<b>Military Software</b>	<b>\$2,500</b>	<b>\$5,000</b>
Average	\$1,014	\$1,829

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

# CREEPING REQUIREMENTS IN 2008



Domain	Average Monthly Rate of Creeping Requirements
Web software projects	4.0%
Commercial Software	3.5%
Information technology	2.5%
System, embedded software	2.0%
<b>Military Software</b>	<b>2.0%</b>
Outsourced Software	1.5%
<b>AVERAGE</b>	<b>2.6%</b>

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

## Lessons Learned/Re-learned in the Last 12 Months - continued



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*

# 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



Rank	First Estimate	Second Estimate (14 months later)	Third Estimate (7 months Later)
1	Specification level - Reliability	Specification	Specification level - Reliability
2	Test Level	Test	Test Level
3	Time Constraints	Requirements Volatility	Requirements Volatility (Change)
4	Real Time Code	Time Constraints	Time Constraints
5	Special Display Requirements	Real Time Code	Real Time Code
6	Quality Assurance Level	Special Display Requirements	Special Display Requirements
7	Development System Volatility	Development System Volatility	Quality Assurance Level
8	Security Requirements	Security Requirements	Development System Volatility
9	Target System Volatility	Target System Volatility	Security Requirements
10	Process Volatility	Quality Assurance Level	Target System Volatility

## 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



# 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”**

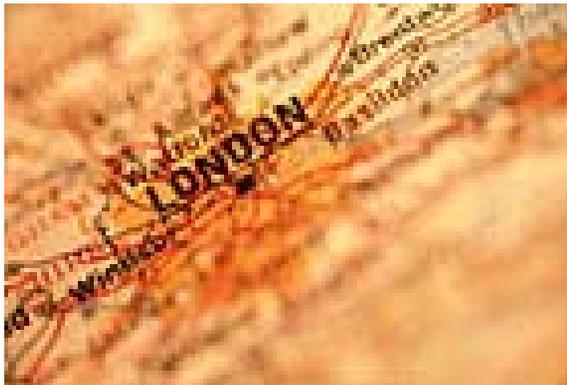
**Douglas Muir, *The New Relevance Of Estimation*, Software Productivity Center**

# 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](mailto:info@galorath.com)

(310) 414-3222 US

Bob Hunt 703-201-0651

[Bhunt@Galorath.com](mailto:Bhunt@Galorath.com)

+44 (0) 1252 720

629 UK

[www.galorath.com](http://www.galorath.com)