

Software in Acquisition Workshop

Dominant Air Power: Design For Tomorrow...Deliver Today



U.S. AIR FORCE

Software-Related Acquisition Challenges – An Aeronautical Systems Center Perspective

16 October 2007

*Mike Nicol
ASC/EN*

937-255-9566

Michael.Nicol@wpafb.af.mil

Air Force Aeronautical Systems Center, Engineering Directorate



Agenda

Aeronautical Systems Center

- **Today's Environment**
- **ASC Commander's Intent**
- **ASC Software Policy**
- **Why Is This So Hard?**
- **Way Ahead**



Today's Environment

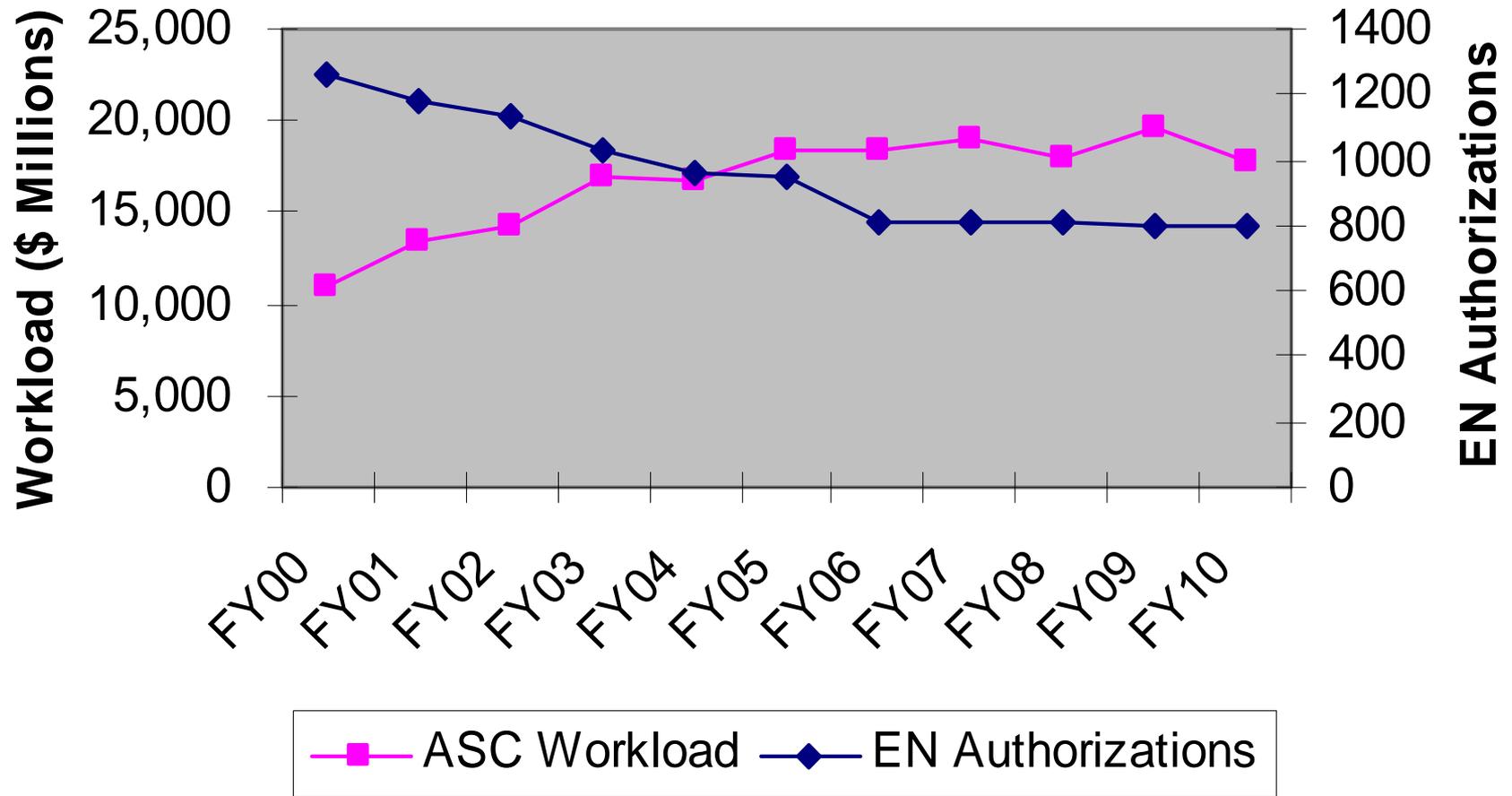
Aeronautical Systems Center

- **Leadership Is Totally Committed To Integrity, Credibility, And Transparency**
- **However,**
 - **Ever Increasing Pressure To Deliver Capability Quickly And Efficiently**
 - **Unprecedented Levels Of Non Developmental Software, Including COTS And Reuse**
 - **Execution Problems Make Programs Targets**
 - **Acquisition Workforce Continues To Decline**



ASC Workload & EN Staffing

Aeronautical Systems Center



Current ASC/EN Authorizations are roughly 1/3 of 1991 levels



ASC Commander's Intent

Aeronautical Systems Center

- **Establish High Confidence, Low to Moderate Risk Programs (Including Software)**
 - Give No Broken Programs To Air Force Or DoD
 - Account For All Risks, Estimate “Conservatively”, And Crosscheck



ASC Software Policy

Aeronautical Systems Center

- **Define High Confidence, Low To Moderate Risk, Acquisition Strategies**
 - Provide Trade Space In Performance Requirements, Cost, And Schedule

- **Clearly Understand The Size Of The Software To Be Developed, Modified, Reused, And Integrated**
 - Develop High Confidence Software Cost And Schedule Estimates Which Reflect Historically Achieved Productivity And Realized Risk Areas Such As Software Size Growth, Incomplete Or Unstable Performance Requirements, And Proposed Software Reuse
 - Accommodate These High Confidence Estimates In All Program Baselines



ASC Software Policy (Cont.)

Aeronautical Systems Center

- **Understand And Evaluate The Software Related Strengths, Weaknesses, Risks, Domain Experience, Process Capability, And Development Capacity; Appropriately Treat Flight And Safety-critical Functions, Development Tools, & Past Performance**
 - Includes Evaluation Of The Draft Software Development Plan (SDP) And Offeror Commitment To The SDP Processes Through The Integrated Master Plan, Integrated Master Schedule, And Other Program Plans And Documents

- **Establish High Confidence That Existing Or Promised Software Envisioned For Reuse Will Be Available When Needed And Suitable Relative To Performance Requirements**



ASC Software Policy (Cont.)

Aeronautical Systems Center

- **Establish Objective Measures Of Software Performance**
 - Apply Earned Value Management
 - Use And Supplement Air Force Core Software Metrics
- **Place Early Attention On Sustainment Planning**
- **Seek Involvement And Assistance From The Acquisition Center Of Excellence (ACE), And Assist With Lessons Learned**



Why Is This So Hard?

Aeronautical Systems Center

- **Warfighter Need**
- **Culture**
- **Inadequate Understanding**
- **Changing Program Baselines**
- **Optimism**
- **Unrealistic Estimate Expectations**
- **Lack Of People In Key Areas**



Why Is This So Hard? Warfighter Need

Aeronautical Systems Center

- **Limited Resources/Budget**
- **Rapidly Bring Capability To The Warfighter**



Why Is This So Hard?

Culture

Aeronautical Systems Center

- **Decades Of Tacit Acceptance Of Medium Or Low Confidence Estimates**
- **Belief That Sticker Shock From High Confidence Estimates Will Cause Programs To Be Cancelled**
- **Belief That High Confidence Estimates Become Self-Fulfilling**
- **Lack Of Unwavering Commitment To High Confidence Estimates**



Why Is This So Hard? Inadequate Understanding

Aeronautical Systems Center

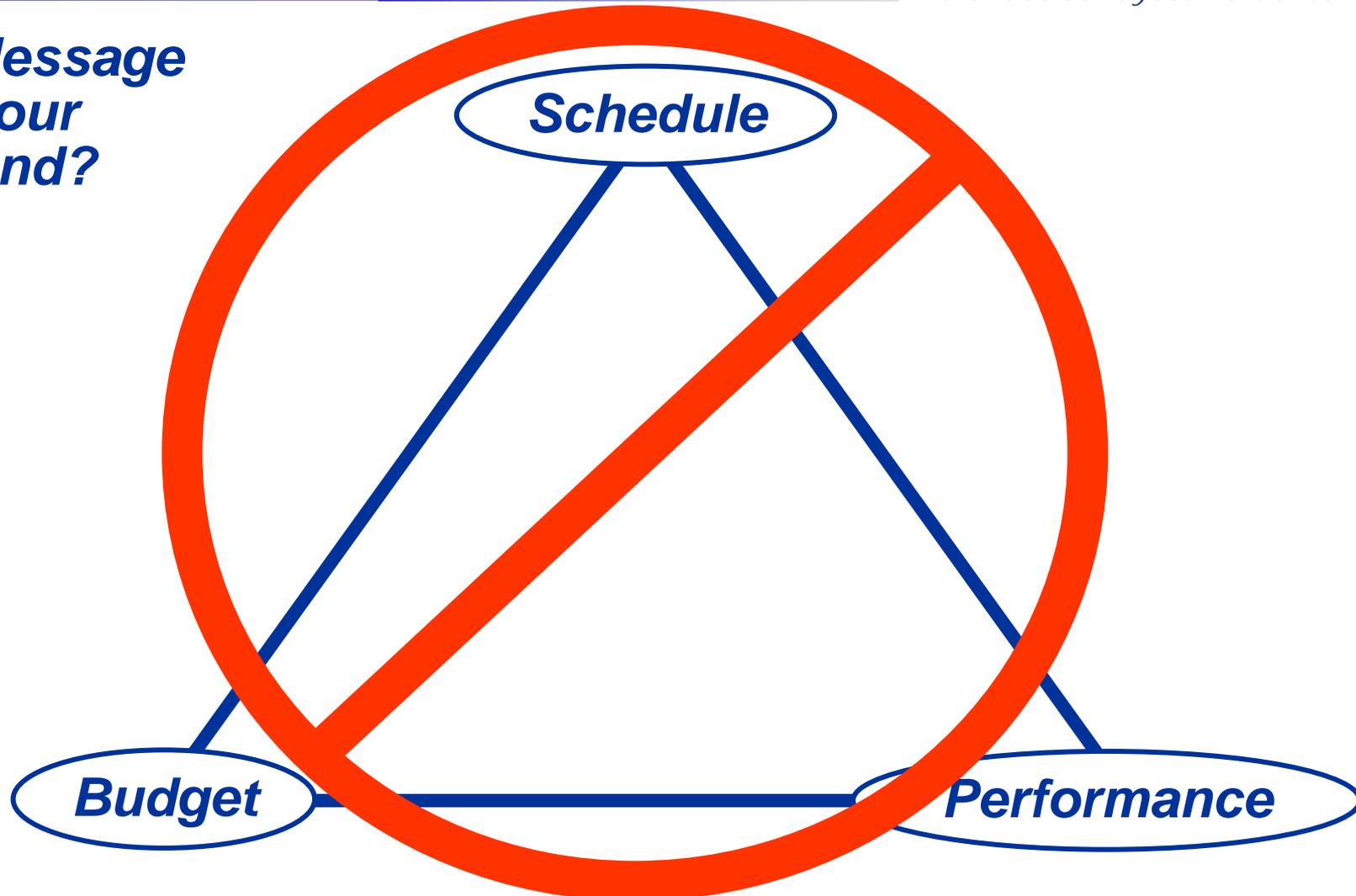
- **Baselines Are Established Before The Size Of The Development Is Understood**
- **Unprecedented Systems/Capabilities Drive Reliance On Expert Judgment**
- **Lack Of Hard Data And/Or Reliance On Expert Judgment Makes Estimate Difficult To Defend**



The Iron Triangle

Aeronautical Systems Center

**What Message
Does Your
RFP Send?**





Establishing Baselines

Aeronautical Systems Center

“One Of The Best Ways To Improve USAF Cost Estimates Is To Wait Until A Preliminary Design Review Has Been Completed, Rather Than Trying To Guess A Weapon’s Price Before The Final Requirements Are Defined”

*Sue Payton
Air Force Assistant Secretary for Acquisition*



Why Is This So Hard? Changing Program Baselines

Aeronautical Systems Center

- **Seemingly Constant Churn In Funding, Requirements, Etc.**
- **Sometimes Difficult To Determine The Source Of Change And Stop The Damage**



Why Is This So Hard?

Optimism

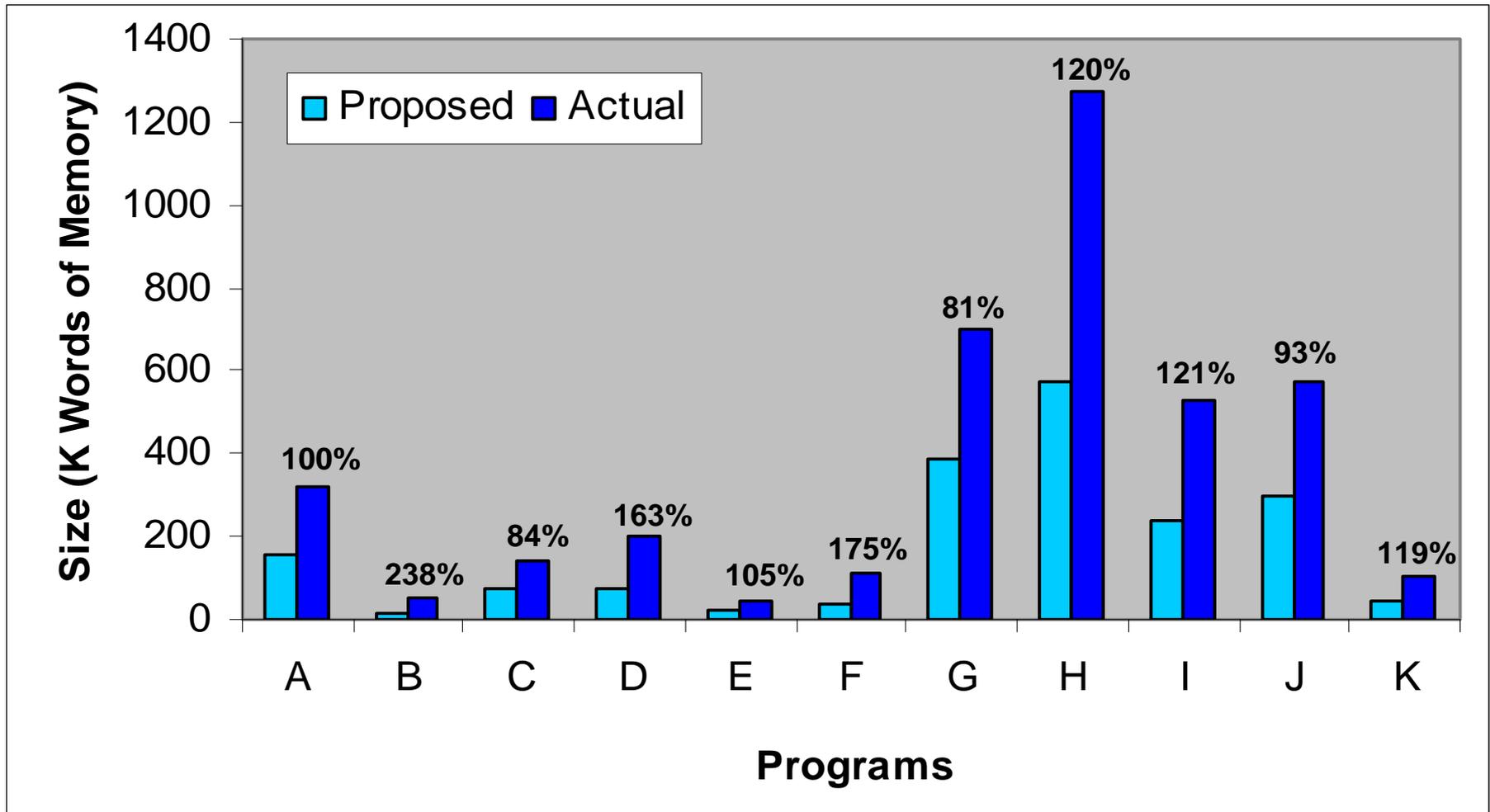
Aeronautical Systems Center

- **Belief That The Size Estimate Is Correct And “Typical” Growth Will Not Occur**
- **Belief That High (Unprecedented?) Levels Of Reuse Can Be Achieved As Proposed**
- **Belief That Better People, Processes, Etc. Bring Significant Increases In Productivity**
- **Program Plans Based On Perfect Execution**
- **Temptation To Doubt The High Confidence Estimate**



Embedded Software Size Growth

Aeronautical Systems Center



Data from ASC Weapon System Programs, circa 1990



Sources Of Software Growth

Aeronautical Systems Center

■ Primary Sources Of Software Growth

- Requirements Changes/Additions
- Improved Understanding Of The Problem

■ ASC/EN Study (Early 90's)

- Roughly 25% Of Changes Introduced By Government
- Roughly 75% Of Changes Introduced By Contractor/Developer
 - ~ 1/3 Of These Introduced By Contractor-Derived Specification Changes
 - ~ 2/3 Of These Due To Evolution Of Lower-level Design Requirements As The Development Proceeds



Building Confidence In Proposed Reuse

Aeronautical Systems Center

- **The Software Is Operational In A Similar Mission Application And Architecture**
- **Performance And Interface Requirements Are Verified To Be Identical**
- **Integration Is Not Complex And Software Can Be Used Essentially “As-is” With Little Or No Development**
- **The Software Is Free Of Known Defects**
- **The Offeror Has Full Control Over The Future Evolution Of The Software**
- **Adequate Property Rights & Licenses Are Established**
- **Proposed Approach For Achieving Predicted Reuse Is Sound, And Planned Verification Is Adequate**
- **The Offeror Has An Institutionalized Process For Making Reuse Decisions**
- **The Offeror Has Significant Familiarity/Experience With The Software**
- **The Offeror Has Access To The Source Code**



Why Is This So Hard?

Unrealistic Estimate Expectations

Aeronautical Systems Center

- **Difficult To Develop High-Confidence Estimates**
- **Numerous Software-Related Risks To Consider**



Attributes of High Confidence Estimates

Aeronautical Systems Center

- **Estimate Is Based On Well Defined, Stable Requirements**
- **Estimate Is Based On A Comprehensive, Detailed, Well Documented, And Realistic Software Development Schedule With Durations And Logic**
- **Actual Cost, Productivity, SLOC, Etc. Data Is Available On The Same Program Or Very Similar Program At The Same Contractor Facility**
- **Program Can Accurately Estimate Software Source Lines Of Code (SLOCS) Given Known State Of Requirements At Estimate Timeframe**
- **Estimate Is Consistent With Historical Program Experience On Size, Code Growth, And Ability To Achieve Planned Levels Of Reuse**
- **Estimate Includes Cost Associated With Modifying And Integrating Any Planned COTS Software**
- **Estimate Includes The Ability To Honestly Characterize Developer Capability And Program Environment**
- **Phases And Activities Included In The Estimate Are Consistent With Other Program Component Estimating Approaches And Content**
- **Estimating Techniques Are Appropriate To The Program Situation And Comprehensiveness Of Available Data**
- **Estimate Is Independently Reviewed & Cross Checked At Aggregate Level**



Typical Software-Related Risks

Aeronautical Systems Center

- 1. Rapid Staff Buildup At The Start Of New Development Programs**
- 2. Complex, Poorly Defined, Incomplete, Or Unstable System Or Software Requirements**
- 3. Hand-off Of Software Requirements From Systems Engineering Without Adequate Interaction**
- 4. Inability To Agree On And Control Build Or Spiral Content (AKA, Lack Of A Baseline)**
- 5. COTS/GOTS Availability, Suitability, Integration, And Sustainment**
- 6. Integration-Heavy Effort (Significant Integration Effort For Existing Components)**
- 7. Concurrent Hardware Development Or Requirements That Drive The Use Of Unproven Tools Or Technology**
- 8. Extensive Security Requirements (Multi-Level Security)**



Typical Software-Related Risks ***(Cont.)***

Aeronautical Systems Center

- 9. Unprecedented System And Software Architectures**
- 10. Long-Duration Development Timeframes**
- 11. Technical Obsolescence Of Computing Architectures And Hardware**
- 12. Safety-Critical Requirements**
- 13. Uncontrolled, Unknown, Or Untrusted Sources Of Software (Foreign Developers, Open Source, Etc.)**
- 14. Government Furnished Equipment/Property (GFE/GFP) With Unknown Performance Capability**
- 15. Use Of Tools, Methods, And Technologies With Which The Developer Has No Previous Experience**
- 16. Multiple Developers And Subcontractors Teaming To Develop Complex Software Intensive Systems Which Must Be Tailored And Integrated Into A Total System Capability**



Why Is This So Hard? Lack of People In Key Areas

Aeronautical Systems Center

- **Estimators, Especially Those Trained In Software Estimation**
- **Schedule Risk Evaluators**
- **Software Engineers**



Way Ahead

Aeronautical Systems Center

- **Continue The Focus On Establishing Realistic Program Baselines**
- **Needs:**
 - Tools That Help Us With Early Estimates Where Non-Developmental Software Is A Primary Driver
 - Acquisition / Contracting Approaches That Allow Program Baselines To Evolve Along With The Understanding Of The System Requirements
 - Across The Board And Consistent Commitment To High Confidence Programs