Software Cost Estimation Issues for Future Ground Systems

Nancy Kern

Software Engineering Department
ETG/RSD
The Aerospace Corporation

Outline

1. Background
2. Software Cost Estimation Research
   - OO Software Development
   - Developing with Reuse
   - Developing with COTS
3. Conclusions

Background

- Multiple large efforts to replace and upgrade current ground systems
- Initial emphasis on modernizing systems using OO technology for better maintainability, flexibility, and extensibility
- Current focus on fielding systems with maximum re-utilization of legacy systems and COTS

Application of Software Cost Knowledge

- Need to have a good understanding of cost implications in order to
  - Develop independent cost estimates at conceptual design phases
  - Evaluate the reasonableness of contractor estimates (cost and schedule) at contract award
  - Identify and manage the risks involved in the development approaches selected
    - Cost Schedule Technical

Outline

1. Background
2. Software Cost Estimation Research
   - OO Software Development
   - Developing with Reuse
   - Developing with COTS
3. Conclusions

Cost Estimation Research
OO Software Development

- Research recently conducted on current practice and new methods for Object-Oriented (OO) SW cost estimation
  - Professional Organizations and Universities
  - DoD Support Organizations
  - Commercial Organizations
- Collection and analysis of data from large-scale OO Software developments planned

OO Software Sizing

- Most difficult task is estimating the size of the software system
- Many approaches to measuring software
  - Function Points and variations
  - Object Points and variations
  - Object Artifacts (e.g., Use Cases)
  - Source Lines of Code (SLOC)
- SLOC continues to be most widely used

OO Software Development Productivity

- Anecdotal evidence indicates productivity rates for OO languages similar to that for other 3GLs
- Productivity increases claimed in contractor historical data may be the result of over-counting for generated code (4GLs) and reused code
- Future Aerospace research into OO Cost Estimating Relationships (CERs)
  - Collection and analysis of OO project data (TBD)
  - Delphi Study of OO Cost Drivers (TBD)
  - OO Metrics Study (Ongoing)

Outline

1. Background
2. Software Cost Estimation Research
   - OO Software Development
   - Developing with Reuse
   - Developing with COTS
3. Conclusions

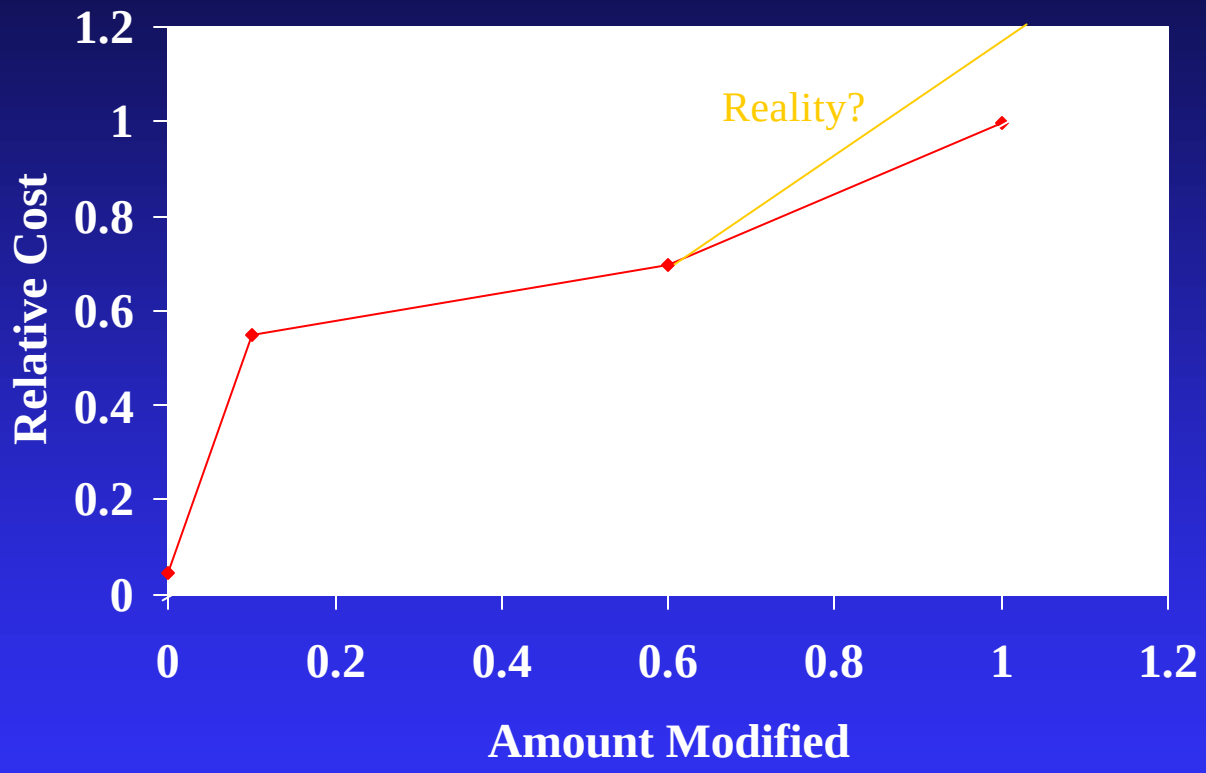
Developing with Reuse

- High expectations for cost and schedule savings by maximizing legacy reuse.
- Over-optimism regarding the amount, quality, and availability of “reusable” code.
- Creates a significant risk area that must be mitigated and managed through the entire life-cycle.
- Challenge in estimating amount of code that can be reused and the effort required to reuse it.

White Box vs. Black Box Reuse

- Black Box - Reuse with no modifications
  - Known inputs and outputs
  - Unknown/irrelevant contents
  - Cost of reuse is > 0
- White Box - Reuse with modifications
  - Requires knowledge of the contents to be used
  - Cost of reuse is >> 0
  - Cost of reuse vs % modification is non-linear

Reuse with Modifications



Source: NASA Selby Study, 1988

Reuse with No Modifications

- Numerous published studies estimate the effort required to reuse software unmodified
- Study results (as a percent of effort required for new development) range from 4.6% to 40%
- Results of Preferred Studies
  - Jeff Poulin: 20%
  - USC/CSE: 18-30%
  - Aerospace Corporation: 25-30%

Outline

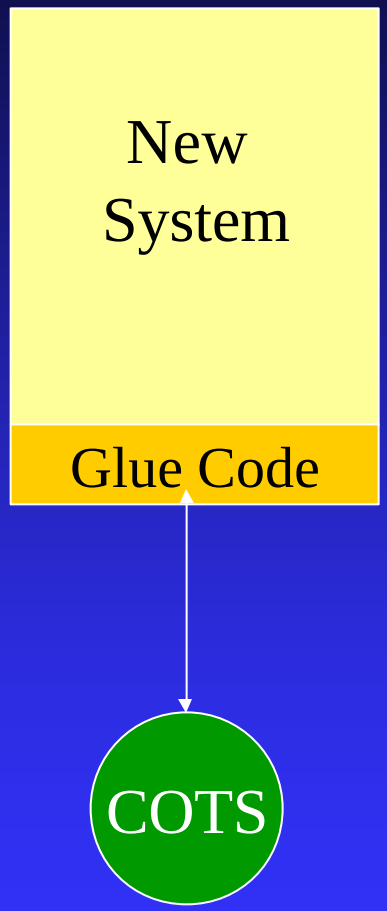
1. Background
2. Software Cost Estimation Research
   - OO Software Development
   - Developing with Reuse
   - Developing with COTS
3. Conclusions

Developing with COTS

- High expectations for cost/schedule savings by emphasizing COTS.
- Expected extent of benefits is rarely being met.
- Amounts of effort for integrating COTS packages are significantly under-estimated (or ignored).
- Expect additional cost and schedule to be required to complete developments involving COTS

COTS Implementation Costs

- COTS product assessments
- COTS tailoring (e.g., customization via installation parameters)
- COTS Glue Code
- Increased application effort due to COTS volatility



Source: Chris Abts (USC-CSE / COCOTS POC)

COTS Integration Cost Exercise

- What is the average cost related to integration for a typical COTS product?
  - A. Insufficient data
  - B. $0
  - C. Cost of 1000 Glue SLOC
  - D. Cost of 1000 Glue SLOC + X ESLOC

A. Insufficient Data

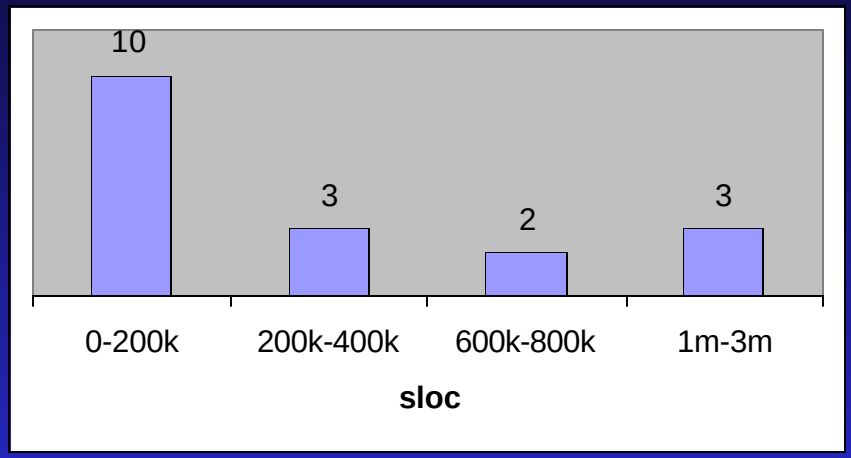
- Obviously the correct answer
- Unfortunately an unacceptable answer

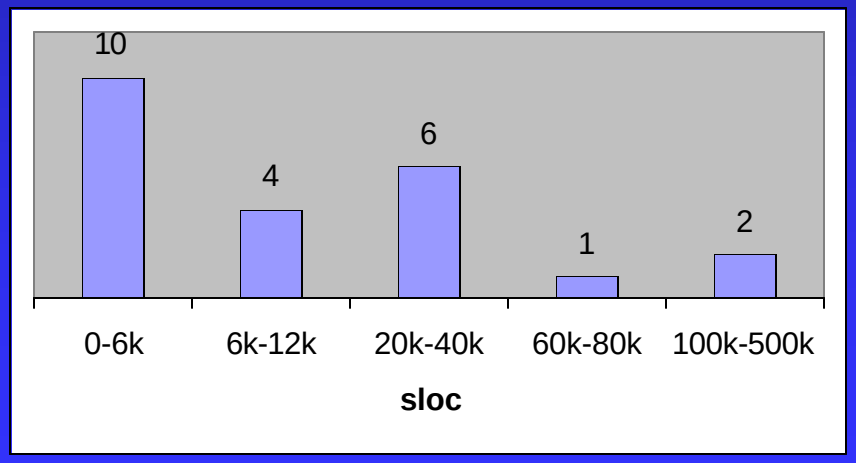
B. $ 0

- Historically, the contractor's answer of choice
- Wrong!

USC Glue Code Data

Total SLOC





Glue SLOC

Source: Chris Abts (USC-CSE)

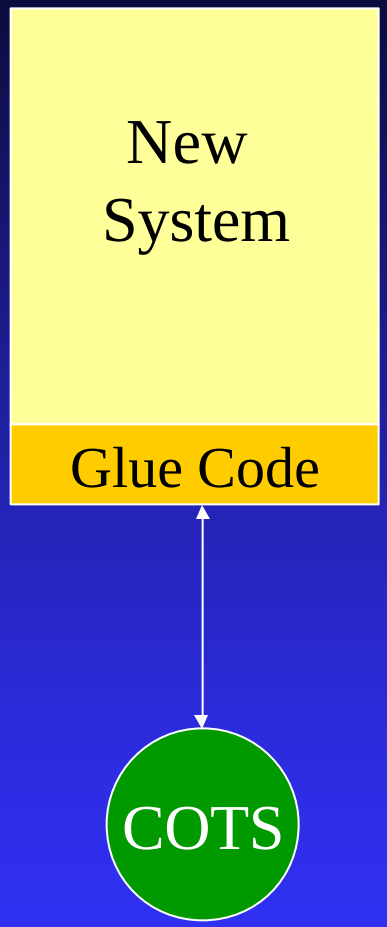
C. Cost of 1000 SLOC of Glue Code

- 1000 ± 300 Glue SLOC on average per COTS product *
- Based on USC-CSE data collected for COCOTS
- In line with 1200 SLOC observed in recent ICEs
- Partial Credit
  - This answer represents a ROM estimation of a portion of the glue-related development costs

* A very rough rule of thumb!

COTS Implementation Costs - Expanded

- COTS product assessments
- COTS tailoring (e.g., customization via installation parameters)
- COTS Glue Code
  - Cost of Glue Code development
  - Cost of additional system design, integration, and test efforts related to the COTS
- Increased application effort due to COTS volatility



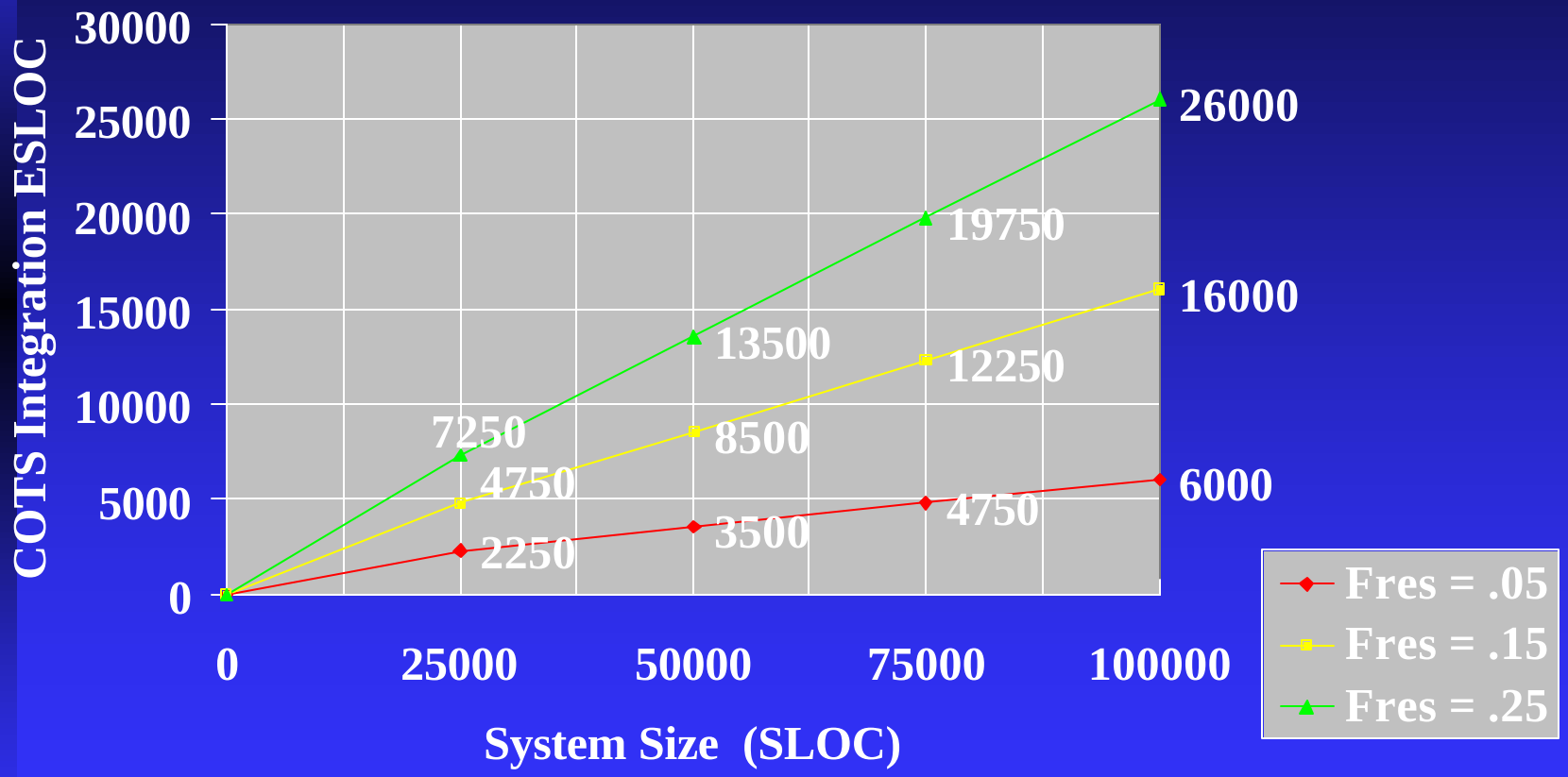
D. Cost of (1000 Glue SLOC + X ESLOC

- ESLOC
  - Equivalent (or Effective) SLOC
  - Adjusts the SLOC estimate to account for additional non-coding effort that will decrease productivity
- Value of X depends on
  - System (or subsystem) size
  - Ease of integrating the COTS product into the system
    - Represented as F_{res} factor in 1997 publication by R. Jensen
    - Factor typically ranges from .05 (easy) to .25 (difficult)
    - Recent estimates using PRICE range from 0.09 to .15

Calculation of COTS Integration Effort

- $S_e = S_{en} * (1 + F_{res})$
  - S_e = Effective size of system including component integration
  - S_{en} = Effective Size of system ignoring COTS effort
  - F_{res} = Ratio of the effort to integrate the component into the product to the development effort ignoring component integration effort
- $X = S_e$ + Glue SLOC
- X increases quickly as either S_{en} or F_{res} increases

COTS Integration Effort Range Estimates



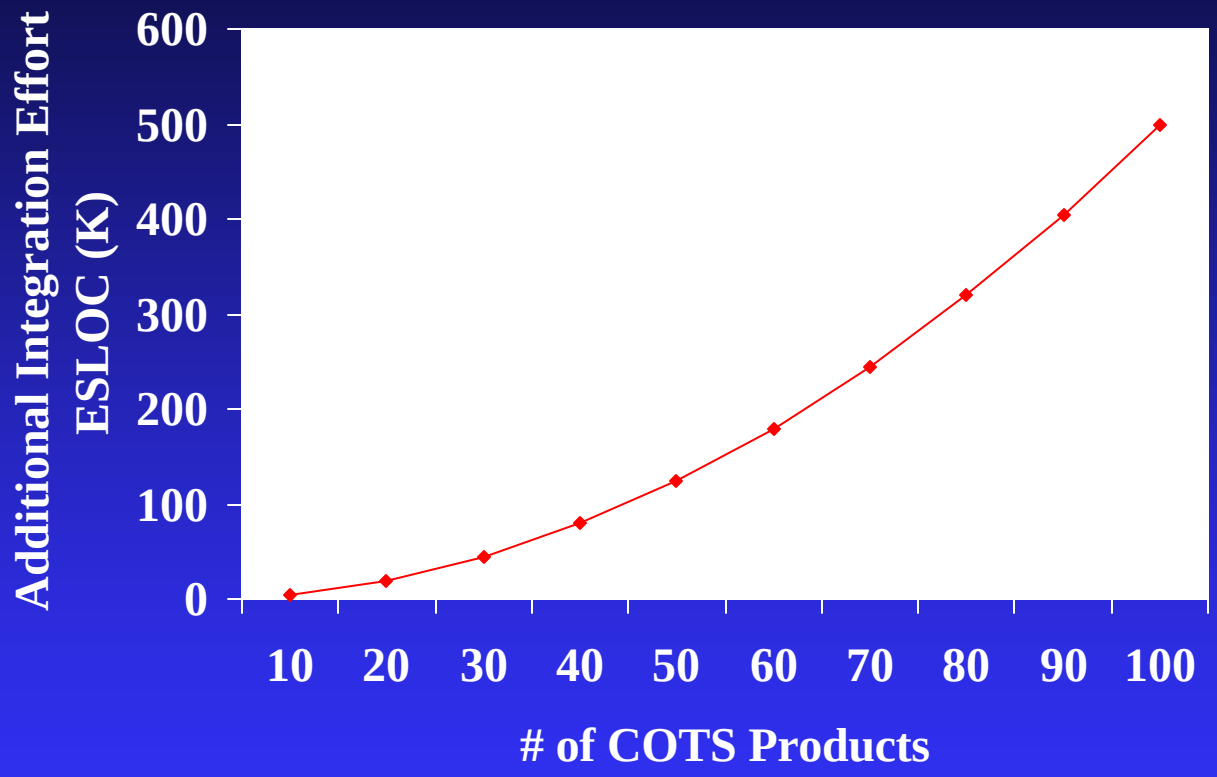
Impact of Multiple COTS Products on Cost

- As the number of COTS products to be integrated into a system increases, the effort involved
  - is probably not linear
  - is likely to grow at least geometrically
  - may grow exponentially
- If the average cost of integrating 1 COTS product into a system is 5,000 ESLOC:
  - Integrating 10 products costs >>50,000 ESLOC
  - Integrating 100 products costs >>>> 500,000 ESLOC

Estimation of Additional Effort to Integrate Multiple COTS Products

- Need to re-adjust the effective size (ESLOC) for additional effort due to COTS interactions.
- For a very conservative estimate:
  - Use total glue code SLOC instead of system size for Sen to calculate X
  - Use a low Fres value of .05
- Example: To integrate 10 COTS products
  - Sen = Total Glue SLOC = 10 * 1000 SLOC = 10000 SLOC
  - Each COTS: Se =10000*1.05 = 10500 ESLOC
  - Additional effort per COTS = 10500 - 10000 = 500 ESLOC
  - Total additional effort = 10*500 ESLOC = 5000 ESLOC

Effort (due to COTS interactions) to integrate multiple COTS products



Outline

1. Background
2. Software Cost Estimation Research
   - OO Software Development
   - Developing with Reuse
   - Developing with COTS
3. Conclusions

Conclusions

- Use of OO, Legacy Reuse, and COTS in large Ground System developments have not met expectations for cost/schedule reductions
- Further studies needed to develop good OO Cost Estimating Relationships (CERs)
- Good tools and methods are available for estimating costs associated with Legacy Reuse
- Hidden costs of COTS development are significant
  - Need to closely scrutinize COTS-related estimates
  - USC working to address these issues with COCOTS
- Recommend Cost-Risk analyses to develop distributions for Low, High, and Most Likely estimates

Bibliography

- Research in OO Metrics, Sizing, and Cost Estimation for GST, Aerospace Report No. TOR-99(3133)-1
- Software Acquisition and Software Engineering Best Practices, Aerospace Report No. TOR-2000(8550)-1
- Cost Estimating Relationships (CERs) for Software Development, Aerospace Report No. TOR-96(8504)-3
- Estimating the Cost of Software Reuse, Randall Jensen, Crosstalk, May1997
- COCOMO II / COCOTS briefings and user manuals, USC-CSE