COSYSMO 3.0:
Cost Estimation of Systems Engineering in a Context of Rapid Changes

Jim Alstad  USC Center for Systems and Software Engineering

Ground Systems Architecture Workshop
March 2, 2016

Jim Alstad*  310/344-0894  jalstad@usc.edu
COSYSMO 3.0 Objectives

• Context:
  – Current and future trends create challenges for full-system cost estimation of satellite systems
    • Emergent requirements, rapid change, net-centric systems of systems, COTS, clouds, apps, widgets, high assurance with agility, multi-mission systems
  – Current development practices can minimize cost of one phase, such as development, while raising ground station cost
• COSYSMO 3.0 is being developed to mitigate this situation by supporting accurate estimates of systems engineering costs, with benefits including:
  – Allowing thoughtful system-level systems engineering during development, which can result in, for example, choosing new technologies that reduce ground station cost
  – Allowing thoughtful systems engineering of ground stations to support life-cycle flexibility
Acknowledgements

1. The development of COSYSMO 3.0 has benefited greatly from GSAW Working Groups held in 2014 & 2015
2. This presentation was adapted from one with these authors:

   Jim Alstad
   Dr Barry W Boehm
   Dr Jo Ann Lane
   Mr Garry Roedler
   Dr Gan Wang
   Ms Marilee Wheaton

   USC Center for Systems and Software Engineering
   Lockheed Martin
   BAE Systems
   The Aerospace Corporation
Agenda:

- The motivation for COSYSMO 3.0
- History of COSYSMO 3.0
- Overview of the content of the COSYSMO 3.0 estimating model
- System-of-systems estimating: interoperability in COSYSMO 3.0
- Summary
COCOMO Family of Cost Models

Software Cost Models

- COCOMO 81
  - 1981
- COCOMO II
  - 2000
  - DBA COCOMO
    - 2004
  - COINCOMO
    - 2004, 2012

Other Independent Estimation Models

- COCOTS
  - 2000
- COSYSMO
  - 2005
- COSYSMO-SoS
  - 2007

Software Extensions

- COQUALMO
  - 1998
  - DBA COCOMO
    - 2004
- AGILE C II
  - 2003
- iDAVE
  - 2004
- COPLIMO
  - 2003
- COPSEMO
  - 1998
- COPROMO
  - 1998
- CORADMO
  - 1999, 2012

Legend:

- Model has been calibrated with historical project data and expert (Delphi) data
- Model is derived from COCOMO II
- Model has been calibrated with expert (Delphi) data

Dates indicate the time that the first paper was published for the model
History of COSYSMO Models

COSYSMO 1.0
Valerdi, 2005
- Identifies form of model
- Identifies basic cost drivers
- Identifies Size measure

With Reuse
Wang et al, 2008
- Adds weights to Size elements, reducing net Size in the presence of reuse

Req’ts Volatile
Pena, 2012
- Adds scale factor based on requirements volatility

For Reuse
Wang et al, 2014
- Adds weights to Size elements, reducing net Size when artifacts are only partially completed

Sys of Sys
Lane et al, 2011
- Adds effort multiplier when in the presence of system-of-systems

COSYSMO 3.0
Alstad, 2016?
- Integrates features of previous models
COSYSMO 3.0 Directions

Incorporate and harmonize existing COSYSMO model research and experience for estimating systems engineering effort:

• Several factors affecting the COSYSMO cost model have been shown to be valuable in increasing estimation accuracy (terminology from [1]):
  – Reuse (partial model—Development With Reuse) [3]
  – Reuse (with Development For Reuse) [1]
  – Requirements volatility (RV) [4]

The rating scales for these could be integrated into a comprehensive COSYSMO model.

Enhancement planned for inclusion:

• System-of-system considerations are hypothesized to affect system engineering costs:
  – Interoperability considerations [6]
Enhancements under discussion:

- Explore a model for total development cost based primarily on the COSYSMO parameters (following work led by Reggie Cole of Lockheed Martin [17, 7])
Agenda:

• The motivation for COSYSMO 3.0
• History of COSYSMO 3.0
• Overview of the content of the COSYSMO 3.0 estimating model
• System-of-systems estimating: interoperability in COSYSMO 3.0
• Summary
COSYSMO 3.0
Top-Level Model

$$PH = A \cdot (AdjSize)^E \cdot \prod_{j=1}^{15} EM_j$$

Elements of the COSYSMO 3.0 model:

- Calibration parameter A
- Adjusted Size model
  - eReq submodel, where 4 products contribute to size
  - Reuse submodel
- Exponent (E) model
  - Accounts for diseconomy of scale
  - Constant and 3 scale factors
- Effort multipliers EM
  - 15 cost drivers
Harmonized COSYSMO 3.0 Size Model

\[
AdjSize_{c3} = \sum_{SizeDrivers} eReq(\text{Type}(SD), \text{Difficulty}(SD)) \times \text{PartialDevFactor}(\text{AL}_{\text{Start}}(SD), \text{AL}_{\text{End}}(SD), \text{RTYPE}(SD))
\]

- **SizeDriver** is one of the system engineering products that determines size in the COSYSMO family (per [2]). Any product of these types is included:
  - System requirement
  - System interface
  - System algorithm
  - Operational scenario

- There are two submodels:
  - Equivalent nominal requirements ("eReq")
    - Raw size
  - Partial development
    - Adjusts size for reuse
Size Model – eReq Submodel

- The eReq submodel is unchanged from [2].
- The submodel computes the size of a SizeDriver, in units of eReq (“equivalent nominal requirements”)
- Each SizeDriver is evaluated as being easy, nominal, or difficult.
- The following table contains conversion factors for the conversion of a SizeDriver to a number of eReq:

<table>
<thead>
<tr>
<th>Size Driver Type</th>
<th>Easy</th>
<th>Nominal</th>
<th>Difficult</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Requirement</td>
<td>0.5</td>
<td>1.0</td>
<td>4.5</td>
</tr>
<tr>
<td>System Interface</td>
<td>1.9</td>
<td>3.9</td>
<td>9.0</td>
</tr>
<tr>
<td>System Algorithm</td>
<td>2.0</td>
<td>3.9</td>
<td>10.0</td>
</tr>
<tr>
<td>Operational Scenario</td>
<td>6.4</td>
<td>13.8</td>
<td>26.1</td>
</tr>
</tbody>
</table>
How Reuse Is Addressed

Reuse has two aspects [1]:

• Development with reuse (DWR): previously developed artifacts are reused on the current project
  – Addressed completely by the DWR partial development model

• Development for reuse (DFR): the current project is creating artifacts to be reused on other projects
  – One aspect of DFR development is that DFR costs more than ordinary development
    • Addressed by the DFR cost driver (below)
  – Another aspect of DFR is that the artifacts may be only partially completed, as during an IR&D project
    • Addressed by the DFR partial development model
Size Model – Partial Development Submodel

• (Concepts here are simplified a little)
• The basic DWR concept:
  – If a reused SizeDriver is being brought in, that saves effort, and so we adjust the size by multiplying the raw size by a PartialDevFactor less than 1.
  – The value of PartialDevFactor is based on the maturity of the reused SizeDriver, and is looked up in a table [24].
    • How fully developed was the SizeDriver?
  – If there is no reuse for this SizeDriver, then PartialDevFactor = 1 (no adjustment).

<table>
<thead>
<tr>
<th>DWR Activity Level:</th>
<th>New</th>
<th>Design Modified</th>
<th>Design Implemented</th>
<th>Adapted for Integration</th>
<th>Adopted for Integration</th>
<th>Managed</th>
</tr>
</thead>
<tbody>
<tr>
<td>DWR % of full-project cost:</td>
<td>100.00%</td>
<td>85.59%</td>
<td>71.18%</td>
<td>58.80%</td>
<td>39.75%</td>
<td>22.52%</td>
</tr>
</tbody>
</table>

• The basic development-for-reuse (DFR) concept is analogous:
  – A product to be reused may be not be taken through the full development cycle (e.g., an IR&D project)

<table>
<thead>
<tr>
<th>DFR Activity Level:</th>
<th>Conceptualized for Reuse</th>
<th>N/A</th>
<th>Designed for Reuse</th>
<th>Constructed for Reuse</th>
<th>N/A</th>
<th>Validated for Reuse</th>
</tr>
</thead>
<tbody>
<tr>
<td>DFR % of full-project cost</td>
<td>32.92%</td>
<td>54.91%</td>
<td>81.27%</td>
<td></td>
<td></td>
<td>96.79%</td>
</tr>
</tbody>
</table>
COSYSMO 3.0
Exponent Model

- Exponent model is expanded from Peña [4, 9]

\[ E = E_{COSYSMO1} + SF_{ROR} + SF_{PC} + SF_{RV} \]

Where:
- \( E_{COSYSMO1} = 1.06 \) [2]
- SF = scale factor
- ROR = Risk/Opportunity Resolution
- PC = Process Capability
- RV = Requirements Volatility

The effect of a large exponent is more pronounced on bigger projects
Harmonized COSYSMO 3.0
Cost Driver Model

Here are the 15 cost drivers:

<table>
<thead>
<tr>
<th>Driver Name</th>
<th>Data Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONOPS &amp; requirements understanding</td>
<td>Subjective assessment of the CONOPS &amp; the system requirements</td>
</tr>
<tr>
<td>Architecture understanding</td>
<td>Subjective assessment of the system architecture</td>
</tr>
<tr>
<td>Stakeholder team cohesion</td>
<td>Subjective assessment of all stakeholders</td>
</tr>
<tr>
<td>Level of service requirements</td>
<td>Subjective difficulty of satisfying the key performance parameters</td>
</tr>
<tr>
<td>Technology risk</td>
<td>Maturity, readiness, and obsolescence of technology</td>
</tr>
<tr>
<td># of Recursive levels in the design</td>
<td>Number of applicable levels of the Work Breakdown Structure</td>
</tr>
<tr>
<td>Development for reuse</td>
<td>Is this project developing artifacts for later reuse?</td>
</tr>
<tr>
<td># and Diversity of installations/platforms</td>
<td>Sites, installations, operating environment, and diverse platforms</td>
</tr>
<tr>
<td>Migration complexity</td>
<td>Influence of legacy system (if applicable)</td>
</tr>
<tr>
<td>Interoperability</td>
<td>Degree to which this system has to interoperate with others</td>
</tr>
<tr>
<td>Personnel/team capability</td>
<td>Subjective assessment of the team's intellectual capability</td>
</tr>
<tr>
<td>Process capability</td>
<td>CMMI level or equivalent rating</td>
</tr>
<tr>
<td>Personnel experience/continuity</td>
<td>Subjective assessment of staff consistency</td>
</tr>
<tr>
<td>Multisite coordination</td>
<td>Location of stakeholders and coordination barriers</td>
</tr>
<tr>
<td>Tool support</td>
<td>Subjective assessment of SE tools</td>
</tr>
</tbody>
</table>
## Harmonized COSYSMO 3.0 Cost Driver Impacts

<table>
<thead>
<tr>
<th>Cost Driver Impacts (EMRs) in COSYSMO 3.0 v35</th>
</tr>
</thead>
<tbody>
<tr>
<td>DFR</td>
</tr>
<tr>
<td>Tool Support</td>
</tr>
<tr>
<td># and Diversity of Installations/ Platforms</td>
</tr>
<tr>
<td>Interoperability</td>
</tr>
<tr>
<td>Multisite Coordination</td>
</tr>
<tr>
<td>Migration Complexity</td>
</tr>
<tr>
<td># of Recursive Levels in the Design</td>
</tr>
<tr>
<td>Personnel Experience/ Continuity</td>
</tr>
<tr>
<td>Stakeholder Team Cohesion</td>
</tr>
<tr>
<td>Architecture Understanding</td>
</tr>
<tr>
<td>Process Capability</td>
</tr>
<tr>
<td>Personnel/Team Capability</td>
</tr>
<tr>
<td>Level of Service Requirements</td>
</tr>
<tr>
<td>Technology Risk</td>
</tr>
<tr>
<td>CONOPS &amp; Requirements Understanding</td>
</tr>
</tbody>
</table>

### Key Drivers and Impacts

- **Teambuilding**:
  - Teambuilding
  - Continuous improvement

- **Staffing**:
  - Personnel Experience/ Continuity
  - Stakeholder Team Cohesion

- **Continuous improvement**:
  - Architecture Understanding
  - Process Capability
  - Personnel/Team Capability
  - Level of Service Requirements
  - Technology Risk
  - CONOPS & Requirements Understanding

**Cost Impacts (EMRs)**:

- DFR: 1.80
- Tool Support: 1.84
- # and Diversity of Installations/ Platforms: 1.84
- Interoperability: 2.05
- Multisite Coordination: 2.06
- Migration Complexity: 2.07
- # of Recursive Levels in the Design: 2.12
- Personnel Experience/ Continuity: 2.26
- Stakeholder Team Cohesion: 2.42
- Architecture Understanding: 2.45
- Process Capability: 2.53
- Personnel/Team Capability: 2.53
- Level of Service Requirements: 2.63
- Technology Risk: 2.65
- CONOPS & Requirements Understanding: 3.09
Agenda

Agenda:

• The motivation for COSYSMO 3.0
• History of COSYSMO 3.0
• Overview of the content of the COSYSMO 3.0 estimating model
• System-of-systems estimating: interoperability in COSYSMO 3.0
• Summary
System-of-Systems and Interoperability

• Suppose that SE work is being done on a system that is a constituent system in a system-of-systems. How is that context manifested in the SE project?
  – Answer: As interoperability requirements
  – Interoperability: The ability of a system to work with another system or group of systems.

• COSYSMO 3.0 includes interoperability as an influence on cost
Lane & Valerdi [6] propose that interoperability be considered a cost influence in the COSYSMO family. Propose this influence could be manifested in two ways:

- Method 1: Add a new cost driver (covered there)
- Method 2: Adjust the easy/medium/difficult rating scale for system interfaces (part of the Size model)

The working COSYSMO 3.0 includes both methods; only one would be retained in final COSYSMO 3.0.
Size Model –
Adjustment for Interoperability

Adjustment for interoperability (Method 2):

• [6] proposes (in its Table 3) that the table that defines the easy/medium/hard rating scale for a system interface (from [2]) be adjusted by adding a new row (the last row in this table):

<table>
<thead>
<tr>
<th>Easy</th>
<th>Medium</th>
<th>Difficult</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple messages and protocols</td>
<td>Moderate communication complexity</td>
<td>Complex protocol(s)</td>
</tr>
<tr>
<td>Uncoupled</td>
<td>Loosely coupled</td>
<td>Tightly coupled</td>
</tr>
<tr>
<td>Strong consensus among stakeholders</td>
<td>Moderate consensus among stakeholders</td>
<td>Low consensus among stakeholders</td>
</tr>
<tr>
<td>Well behaved</td>
<td>Predictable behavior</td>
<td>Emergent behavior</td>
</tr>
<tr>
<td>Domain or enterprise standards employed</td>
<td>Functional standards employed</td>
<td>Isolated or connected systems with few or no standards</td>
</tr>
</tbody>
</table>
Agenda:

• The motivation for COSYSMO 3.0
• History of COSYSMO 3.0
• Overview of the content of the COSYSMO 3.0 estimating model
• System-of-systems estimating: interoperability in COSYSMO 3.0
• Summary
COSYSMO 3.0 will provide independent estimates of the cost of thorough systems engineering required based on project parameters.
- Thereby assisting in facing the challenge of the rapid rate of change.


Bibliography (4/4)


