Practical Software and Systems Measurement

User’s Group - 2012

Software Maintenance Life Cycle Cost Estimation Model

Workshop Materials and Discussion Topics

Jack McGarry - U.S. Army ARDEC
Dr. Robert Charette - ITABHI Corp.

1 August 2012
Workshop Objectives

• Present an integrated software maintenance life cycle estimation model that links software maintenance resource requirements to planned program sustainment profiles

• Examine the viability of the underlying model concepts and constructs

• Gather inputs focused on improving the design and implementation of the model
Participants

- Brad Clark
- Denton Tarbet
- Cari Pullen
- Chris Miller
- Bob Epps
- Paul McMahon
- Marc Gutleber
- Kim Elliott
- Rich Mabe
- Larry Osiecki

- Don Beckett
- Bob Ferguson
- Corinne Wallshein
- Joe Dean
- Jeramia Poland
- Bob McCann
- Thomas Harless
- Rich Schubert
- Arlene Minkiewicz
Workshop Agenda

• Review the overall estimation model requirements, architecture, and key concepts

• Examine the underlying model constructs
  - Underlying quantitative and context data
  - Software maintenance products and activities
  - Cost estimating relationships - cost drivers
  - Risk and uncertainty application constructs
  - Integrated cost allocation model

• Summarize inputs and recommendations
Notional Software Maintenance Life-Cycle Cost Model

- **EMD**
  - Development
- **PDSS**
  - Production
- **PPSS**
  - Maintenance

Cost

- **DEV**
- **Design Obsolescence**
- **Minimal Maintenance**

Software Maintenance Estimation Period

A B C
Outcomes

• Program context fundamentally impacts the baseline cost estimation process
  - Data structures and data availability
  - Domain estimation relationships - norms
  - Resource allocation - application structures

• The software maintenance WBS needs to be adaptable to each program - and still provide a common task-activity baseline

• Primary focus remains the release profile

• Need to refine the design and mapping of the prospective CERs
Problem Statement

Current software maintenance estimation methodologies do not support objective and defensible resource projections and allocations in the emerging Army systems sustainment environment
Practical Software and Systems Measurement

User’s Group - 2012

Software Maintenance Life Cycle Cost Estimation Model

Workshop Materials and Discussion Topics

Jack McGarry - U.S. Army ARDEC
Dr. Robert Charette - ITABHI Corp.

1 August 2012
Workshop Objectives

• Present an integrated software maintenance life cycle estimation model that links software maintenance resource requirements to planned program sustainment profiles

• Examine the viability of the underlying model concepts and constructs

• Gather inputs focused on improving the design and implementation of the model
Workshop Agenda

• Review the overall estimation model requirements, architecture, and key concepts

• Examine the underlying model constructs
  - Underlying quantitative and context data
  - Software maintenance products and activities
  - Cost estimating relationships - cost drivers
  - Risk and uncertainty application constructs
  - Integrated cost allocation model

• Summarize inputs and recommendations
Estimation Methodology Requirements

- Integrate software maintenance product and activity costs over the life cycle of a program
- Align the cost profile to the specific program life cycle software maintenance improvement and release strategy
- Encompasses both PDSS and PPSS efforts
- Ability to estimate at Milestones A, B, and C
- Integrates component cost estimates from multiple CER based product and activity estimates (WBS defined)
- Directly related to development technical parameters and development performance
- Can incorporate cost risk uncertainty at component and higher levels
- Practical - defensible - data derived - adaptable
Practical Software and Systems Measurement

Data Architecture
- Data Source Data
  - information requirements
  - program context data
  - planned/actual cost & technical data
  - risk-uncertainty data

Software Source Data
- Structured Cost Data
  - data set characterization
    - availability/content
    - integrity
    - usability
  - data demographics
  - data resource access

Work Breakdown Structure
- OPS-29 Funding Model
  - Terminology
    - program environment

Cost Risk Uncertainty Model
- Analysis
  - cost uncertainty factors
  - risk relationships

Cost Estimating Relationships
- Analysis
  - parametric relationships
  - cost ratios
  - product-activity specific
  - domain specific

Software Policy
- • best practices
- • requirements

Data & Reporting Requirements
- • data requirements
- • measures - data sources

Integrated Cost Estimation Model
- • estimation process model
- • product-activity-context driven estimation methodologies
- • risk based cost uncertainties
- • predictive cost performance
- • life cycle phase relationships

5 - 1 August 2012
Notional Software Maintenance Life-Cycle Cost Model

- EMD (Development)
- PDSS (Production)
- PPSS (Maintenance)

Cost

A B C

Software Maintenance Estimation Period

6 - 1 August 2012
Current Software Maintenance Estimation Methods

• **Parametric models**
  - *Product focused* - only include corrective, perfective, adaptive changes & enhancements
  - *Invalid assumptions* about sustaining engineering tasks
  - *Historical data* not visible in all models
  - *Not calibrated* - validated

• **Historical software maintenance funding - effort**
  - *Limited access* to historical FTE - Budget data-
  - *Represent “rolled-up” costs*
  - *Requirements or LOE funding?*
  - *Availability of correlated program development - sustainment cost data*
Current Estimation Methods (continued)

• **Number of lines of code per software engineer**
  - Each engineer can maintain 20K-25K LOC/ESLOC
  - Does not reflect the impact of software reuse or COTS

• **Software maintenance estimated as a percentage of development costs**
  - Rule(s) of thumb - development based:
    • S/W maintenance costs - 2/3 of total S/W life cycle costs
    • S/W maintenance costs - 60% to 75% of total S/W life cycle costs
    • Annual S/W maintenance costs - 5% to 10% of total S/W life cycle costs
  - Ignores total system life cycle software growth and maintenance requirements/strategy/tasks

All of these methods have significant limitations in the current Army environment
Practical Software and Systems Measurement

Data Architecture

Software Source Data
- information requirements
- program context data
- planned/actual cost & technical data
- risk-uncertainty data

Structured Cost Data
- data set characterization
  - availability/content
  - integrity
  - usability
- data demographics
- data resource access

Work Breakdown Structure

OPS-29 Funding Model

Terminology
- program environment

Cost Risk Uncertainty Model
- cost uncertainty factors
- risk relationships

Cost Estimating Relationships
- parametric relationships
- cost ratios
- product-activity specific
- domain specific

Analysis

Integrated Cost Estimation Model
- estimation process model
- product-activity-context driven estimation methodologies
- risk based cost uncertainties
- predictive cost performance
- life cycle phase relationships

Software Policy
- best practices
- requirements

Data & Reporting Requirements
- data requirements
- measures - data sources

9 - 1 August 2012
Categories of Data

- **Quantitative performance data** - effort, product measures, activity measures, etc.
- **Funding data** - budgets, expenditures, dollars and resource allocations, accounting structures, etc.
- **Technical data** - domain, mission, architecture, interfaces, etc.
- **Program profile data** - maintenance/release strategy, schedules, no. of deployed systems, etc.
- **Program context data** - technical and operational risks, environmental risks, mandates, policy, etc.

*Data is Time Dependent*
## Operating Environments - Domains

### Count of Operating Environment

<table>
<thead>
<tr>
<th></th>
<th>Helicopter</th>
<th>UAV</th>
<th>Ground</th>
<th>Missile</th>
<th>Grand Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mission Processing</td>
<td></td>
<td></td>
<td>5</td>
<td></td>
<td>5</td>
</tr>
<tr>
<td>Real Time</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Embedded - Other System</td>
<td>3</td>
<td></td>
<td>5</td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>Training</td>
<td></td>
<td></td>
<td>4</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>Vehicle Control</td>
<td></td>
<td>2</td>
<td>2</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>Vehicle Payload</td>
<td></td>
<td>2</td>
<td></td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Grand Total</td>
<td>3</td>
<td>2</td>
<td>22</td>
<td>2</td>
<td>29</td>
</tr>
</tbody>
</table>

### Count of Super-Domain

<table>
<thead>
<tr>
<th></th>
<th>Helicopter</th>
<th>UAV</th>
<th>Ground</th>
<th>Missile</th>
<th>Grand Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mission Critical - Embedded</td>
<td>3</td>
<td>2</td>
<td>7</td>
<td>2</td>
<td>14</td>
</tr>
<tr>
<td>Mission Critical - Non-Embedded</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9</td>
</tr>
<tr>
<td>Mission Support</td>
<td></td>
<td>6</td>
<td></td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>Grand Total</td>
<td>3</td>
<td>2</td>
<td>22</td>
<td>2</td>
<td>29</td>
</tr>
</tbody>
</table>
## Army Software Maintenance Data

<table>
<thead>
<tr>
<th>Org</th>
<th>Program</th>
<th>Number of Releases</th>
<th>WBS Task Performed</th>
<th>Interfaces</th>
<th>Effort</th>
<th>Cost ($)</th>
<th>Schedule</th>
<th>Software Change Requests (SCRs)</th>
<th>Requirements</th>
<th>Defects</th>
<th>Backlog</th>
<th>Source Lines of Code</th>
<th>OPS-29 Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Org 1</td>
<td>Program 1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Program 2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Program 3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>Program 4</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Program 5</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Program 6</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Program 7</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Program 8</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Program 9</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>Program 10</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Program 11</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Program 12</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>Program 13</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>Org 2</td>
<td>Program 1</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>Program 2</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Program 4</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>Program 5</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>Program 6</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>Program 7</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>Program 8</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>Program 9</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
</tr>
<tr>
<td>Org 3</td>
<td>Program 1</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
</tr>
<tr>
<td>Org 4</td>
<td>Program 1</td>
<td>19</td>
<td>19</td>
<td>19</td>
<td>19</td>
<td>19</td>
<td>19</td>
<td>19</td>
<td>19</td>
<td>19</td>
<td>19</td>
<td>19</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>Program 2</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Program 3</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>21</td>
</tr>
<tr>
<td></td>
<td>Program 4</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>Org 5</td>
<td>Program 1</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
</tr>
<tr>
<td></td>
<td>Program 2</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
</tr>
<tr>
<td></td>
<td>Program 3</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
<td>Multiple</td>
</tr>
<tr>
<td>Org 6</td>
<td>Program 1</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
</tr>
<tr>
<td></td>
<td>Program 2</td>
<td>24</td>
<td>24</td>
<td>24</td>
<td>24</td>
<td>24</td>
<td>24</td>
<td>24</td>
<td>24</td>
<td>24</td>
<td>24</td>
<td>24</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>Program 4</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
</tr>
<tr>
<td></td>
<td>Program 5</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
</tr>
</tbody>
</table>

12 - 1 August 2012
Data Analysis Summary

• Availability of data varies significantly by organization and program
• Execution data (actuals) is very limited - most data was planning data aligned with the OPS-29 funding model
• Multiple funding streams exist for similar work
• Data is cost based - not product/activity based
• Software engineering data derived from different source artifacts for a given release are sometimes inconsistent
• The aggregation levels of provided data are inconsistent (details vs. summary)
• The data that was available provided significant insight into potential cost estimating relationships and maintenance products and activity allocations
Practical Software and Systems Measurement

Data Architecture
- Information requirements
- Program context data
- Planned/actual cost & technical data
- Risk-uncertainty data

Software Source Data

Structured Cost Data
- Data set characterization
  - Availability/content
  - Integrity
  - Usability
- Data demographics
- Data resource access

Work Breakdown Structure

OPS-29 Funding Model

Terminology
- Program environment

Analysis

Cost Risk Uncertainty Model
- Cost uncertainty factors
- Risk relationships

Analysis

Cost Estimating Relationships
- Parametric relationships
- Cost ratios
- Product-activity specific
- Domain specific

Software Policy
- Best practices
- Requirements

Data & Reporting Requirements
- Data requirements
- Measures - data sources

Integrated Cost Estimation Model
- Estimation process model
- Product-activity-context driven estimation methodologies
- Risk based cost uncertainties
- Predictive cost performance
- Life cycle phase relationships
Work Breakdown Structure

Software Maintenance, Sustaining Engineering, and Operational Support

1.0 Software Maintenance
- Release N
- Planning - Management
- Software Requirements
- Architecture & Design
- Change Implementation
- Integration & Test
- Acceptance Test
- Rework
- Emergency Repairs
- Hardware Updates
- Release N+1
- Release N+2
- Release N+3

2.0 Software Licenses
- Deployed Systems
- Facility Systems

3.0 Information Assurance
- IAVA

4.0 Certification & Accreditations
- Mission, Safety, Performance
- DIACAP

5.0 Sustaining Engineering
- Analysis and Studies
- Test Support
- Software Delivery
- User Training
- User Support
- Field Support

6.0 Facilities & Infrastructure
- Development Facilities
- Integration and Test Facilities
- Tactical Equipment
- Test Equipment and Tools

7.0 Management
- Program Management
- Contract Management
- Change Management
- Data Management
- Quality Assurance
- Process Management
- Personnel Management

Version 2.82

15 - 1 August 2012
Work Breakdown Structure

- Common structure that includes all potential products and activities - “what’s in - what’s out”
- Common definitions - terminology
- Emphasis on DOD weapons systems
- Basis for identifying specific cost elements attributable to a given program or system maintenance/sustaining engineering effort
- Product based - driven by changes to the software baseline(s)
- Release focused - primary software maintenance product
- Intended to be tailored and adapted for each program or organization:
  - addition/deletion of lower level cost elements
  - re-binning of lower level cost elements
- Foundation for cost estimation process/models
- Basis for defining cost estimating relationships
Work Breakdown Structure

1.0 Software Maintenance - products and activities associated with modifying an operational software product or system

2.0 Software Licenses - products and activities associated with the procurement and renewal of software licenses for operational software

3.0 Information Assurance - products and activities associated with ensuring that the software is compliant with externally defined information assurance requirements

4.0 Certifications and Accreditations - products and activities associated with verifying a software system against externally defined domain performance criteria

5.0 Sustaining Engineering - products and activities associated with supporting a deployed software product or system in its operational environment

6.0 Facilities & Infrastructure - products and activities associated with establishing and operating the facilities and processes required to modify, integrate, and test operational software products or systems

7.0 Management - products and activities associated with planning, organizing, funding, and controlling the resources required to support operational software products or systems
# Program WBS Element Coverage

| WBS   | Title                                | #  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 |
|-------|--------------------------------------|----|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1.0   | Software Maintenance                | 17 |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1   | Software Release (N)                | 7  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.1 | Planning and Management             | 19 |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.2 | Software Requirements              | 24 |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.3 | Architecture and Design             | 24 |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.4 | Change Implementation              | 23 |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.4.1| Change Development                 | 7  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.4.2| Unit Testing                       | 7  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.5 | Integration and Test                | 24 |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.5.1| Test Planning                      | 8  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.5.2| Test Scenario and Test Case         | 8  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.5.3| Test Tool Development              | 8  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.5.4| Test Conduct                       | 8  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.6 | Acceptance Testing                 | 29 |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.6.1| Test Planning                      | 7  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.6.2| Test Scenario and Test Case         | 7  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.6.3| Test Tool Development              | 7  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.6.4| Test Conduct                       | 7  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.7 | Rework                              | 7  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.8 | Emergency Repairs                  | 25 |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.1.9 | Hardware Updates                   | 15 |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.2   | Software Release (N+1)              | 0  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.3   | Software Release (N+2)              | 0  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 1.4   | Software Release (N+3)              | 0  |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 2.0   | Licenses                            | 11 |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 3.0   | Information Assurance               | 27 |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 3.1   | IAVA                                | 12 |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 3.2   | DIACAP                              | 11 |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 4.0   | Certifications & Accreditations     | 12 |   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
**Program WBS Element Coverage (cont.)**

<table>
<thead>
<tr>
<th>5.0</th>
<th>Software Sustaining Engineering</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1</td>
<td>Analysis and Studies</td>
<td>18</td>
</tr>
<tr>
<td>5.2</td>
<td>Test Support</td>
<td>16</td>
</tr>
<tr>
<td>5.3</td>
<td>Delivery</td>
<td>15</td>
</tr>
<tr>
<td>5.4</td>
<td>User Training</td>
<td>15</td>
</tr>
<tr>
<td>5.5</td>
<td>User Support</td>
<td>16</td>
</tr>
<tr>
<td>5.6</td>
<td>Field Support</td>
<td>26</td>
</tr>
<tr>
<td>6.0</td>
<td>Software Facilities and Infrastructure</td>
<td>16</td>
</tr>
<tr>
<td>6.1</td>
<td>Development Facilities</td>
<td>11</td>
</tr>
<tr>
<td>6.1.1</td>
<td>Equipment</td>
<td>7</td>
</tr>
<tr>
<td>6.1.2</td>
<td>Licenses</td>
<td>11</td>
</tr>
<tr>
<td>6.1.3</td>
<td>Operations</td>
<td>7</td>
</tr>
<tr>
<td>6.2</td>
<td>Integration and Test Facilities</td>
<td>27</td>
</tr>
<tr>
<td>6.2.1</td>
<td>Equipment</td>
<td>7</td>
</tr>
<tr>
<td>6.2.2</td>
<td>Licenses</td>
<td>14</td>
</tr>
<tr>
<td>6.2.3</td>
<td>Operations</td>
<td>11</td>
</tr>
<tr>
<td>6.3</td>
<td>Tactical Equipment</td>
<td>11</td>
</tr>
<tr>
<td>6.4</td>
<td>Test Equipment and Tools</td>
<td>27</td>
</tr>
<tr>
<td>7.0</td>
<td>Management</td>
<td>11</td>
</tr>
<tr>
<td>7.1</td>
<td>Program Management</td>
<td>29</td>
</tr>
<tr>
<td>7.1.1</td>
<td>Project Release Management</td>
<td>12</td>
</tr>
<tr>
<td>7.1.2</td>
<td>Risk Management</td>
<td>11</td>
</tr>
<tr>
<td>7.1.3</td>
<td>Measurement/ Analysis</td>
<td>26</td>
</tr>
<tr>
<td>7.2</td>
<td>Contract Management</td>
<td>17</td>
</tr>
<tr>
<td>7.3</td>
<td>Change Management</td>
<td>27</td>
</tr>
<tr>
<td>7.4</td>
<td>Data Management</td>
<td>24</td>
</tr>
<tr>
<td>7.5</td>
<td>Quality Assurance</td>
<td>29</td>
</tr>
<tr>
<td>7.5.1</td>
<td>V&amp;V</td>
<td>26</td>
</tr>
<tr>
<td>7.6</td>
<td>Process Management</td>
<td>12</td>
</tr>
<tr>
<td>7.7</td>
<td>Personnel Management</td>
<td>0</td>
</tr>
</tbody>
</table>
Cost Estimating Relationships

- Types of CERs
  - product
  - activity
- CER constructs
  - parametric
  - cost ratios
  - simple relationships
  - historical norms
- Types of costs
  - variable
  - semi-variable
  - fixed
- Key considerations
  - domain
  - life cycle phase
  - change drivers
Cost Estimating Relationships

• Ratios used to identify performance relationships

• Product costs structured into software releases - estimated parametrically based on functional change content

• Activity costs estimated using simple CER relationships - unique variables and drivers

• Fixed costs based on historical resource expenditures

• Related factors influence and/or modify the outcomes in all cases
## WBS Based Cost Estimating Relationships

<table>
<thead>
<tr>
<th>WBS</th>
<th>Title</th>
<th>Army OPS-29 Mapping</th>
<th>Variability</th>
<th>Cost Drivers Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Software Release</td>
<td></td>
<td></td>
<td>Lower level cost roll-up</td>
</tr>
<tr>
<td>1.1.1 to 1.1.5</td>
<td>Planning, Req'ts, Design, Implementation, Development, Unit Testing, Integration &amp; Test (typical SW Development)</td>
<td>Capability Sets FY(XX/XX) System Mission Capability</td>
<td>Variable Cost</td>
<td>Parametric modeling - Number of Requirements (Enhancements), Defects, Test Cases - added, reused, modified; App. Domain, Complexity, Reliability, Personnel Factors</td>
</tr>
<tr>
<td>1.1.6</td>
<td>Acceptance Test</td>
<td>Capability Sets FY(XX/XX) System Mission Capability</td>
<td>Variable Cost</td>
<td>Number of requirements / capabilities / … / test cases / etc.)</td>
</tr>
<tr>
<td>2.0</td>
<td>Licenses</td>
<td>Licenses</td>
<td>Semi-Var</td>
<td>Type and number of COTS products.</td>
</tr>
<tr>
<td>3.1</td>
<td>IAVA</td>
<td>IAVA</td>
<td>Semi-Var Cost</td>
<td>Parametric model (percentage distribution); Application domain, size, effort staffing, duration, productivity</td>
</tr>
<tr>
<td>4.0</td>
<td>Certifications &amp; Accreditations</td>
<td>C&amp;A</td>
<td>Fixed</td>
<td>Cost per C&amp;A</td>
</tr>
<tr>
<td>5.0</td>
<td>SW Sustaining Engineering</td>
<td></td>
<td>Roll-up</td>
<td>Lower level cost roll-up</td>
</tr>
<tr>
<td>5.1 to 5.5</td>
<td>Analysis, Test Support, Delivery, Training</td>
<td>Organic Labor</td>
<td>Semi-Var Cost</td>
<td>LOE</td>
</tr>
<tr>
<td>5.6</td>
<td>Field Support</td>
<td>FSEs</td>
<td>Semi-Var</td>
<td>LOE, No. of Field Sites</td>
</tr>
<tr>
<td>6.0</td>
<td>Software Facilities &amp; Infrastructure</td>
<td>System Infrastructure or System Open Door</td>
<td>Fixed</td>
<td># People at facility, Simulation / Test Equip Maintenance</td>
</tr>
<tr>
<td>7.0</td>
<td>Management</td>
<td>Organic</td>
<td>Roll-up</td>
<td>Lower level cost roll-up</td>
</tr>
<tr>
<td>7.1 to 7.7</td>
<td>Program / Release / Risk / Contract Management</td>
<td></td>
<td>Fixed Cost</td>
<td>LOE</td>
</tr>
</tbody>
</table>
Types of Software Maintenance Costs

Variable

Semi Variable

Fixed

Cost

Time
## Correlation Coefficients Matrix

<table>
<thead>
<tr>
<th></th>
<th>Sample size</th>
<th>Critical value (10%)</th>
<th>1.94</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Interfaces</th>
<th>Age</th>
<th>Est_Total_KSLOC</th>
<th>Req_Fund_FY11</th>
<th>Exec_Fund_FY11</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interfaces</td>
<td>Pearson Correlation Coefficient</td>
<td>1.00</td>
<td>0.70</td>
<td>0.39</td>
<td>0.50</td>
</tr>
<tr>
<td></td>
<td>R Standard Error</td>
<td></td>
<td>1.00</td>
<td>0.87</td>
<td>1.00</td>
</tr>
<tr>
<td></td>
<td>t</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>p-value</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>H0 (10%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Age</td>
<td>Pearson Correlation Coefficient</td>
<td>0.47</td>
<td>1.00</td>
<td>0.30</td>
<td>0.59</td>
</tr>
<tr>
<td></td>
<td>R Standard Error</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>t</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>p-value</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>H0 (10%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Est_Total_KSLOC</td>
<td>Pearson Correlation Coefficient</td>
<td>0.70</td>
<td>0.23</td>
<td>0.30</td>
<td>0.59</td>
</tr>
<tr>
<td></td>
<td>R Standard Error</td>
<td></td>
<td>1.00</td>
<td>0.87</td>
<td>1.00</td>
</tr>
<tr>
<td></td>
<td>t</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>p-value</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>H0 (10%)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Req_Fund_FY11</td>
<td>Pearson Correlation Coefficient</td>
<td>0.39</td>
<td>0.15</td>
<td>0.44</td>
<td>0.81</td>
</tr>
<tr>
<td></td>
<td>R Standard Error</td>
<td></td>
<td>0.14</td>
<td>0.15</td>
<td>0.04</td>
</tr>
<tr>
<td></td>
<td>t</td>
<td></td>
<td>1.04</td>
<td>0.77</td>
<td>4.41</td>
</tr>
<tr>
<td></td>
<td>p-value</td>
<td></td>
<td>0.34</td>
<td>0.47</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>H0 (10%)</td>
<td></td>
<td>accepted</td>
<td>accepted</td>
<td>rejected</td>
</tr>
<tr>
<td>Exec_Fund_FY11</td>
<td>Pearson Correlation Coefficient</td>
<td>0.50</td>
<td>0.11</td>
<td>0.12</td>
<td>0.94</td>
</tr>
<tr>
<td></td>
<td>R Standard Error</td>
<td></td>
<td>0.13</td>
<td>0.11</td>
<td>0.06</td>
</tr>
<tr>
<td></td>
<td>t</td>
<td></td>
<td>1.41</td>
<td>1.78</td>
<td>3.40</td>
</tr>
<tr>
<td></td>
<td>p-value</td>
<td></td>
<td>0.21</td>
<td>0.13</td>
<td>0.01</td>
</tr>
<tr>
<td></td>
<td>H0 (10%)</td>
<td></td>
<td>accepted</td>
<td>accepted</td>
<td>rejected</td>
</tr>
</tbody>
</table>
Funding vs. Software Size

\[ y = 272.34x^{0.491} \]
\[ R^2 = 0.864 \]

\[ y = 32.975x^{0.5976} \]
\[ R^2 = 0.8956 \]
Preliminary Cost Relationships

• Results
  - Estimated total system KSLOC and number of interfaces had the highest correlation to both funded and executed funding
  - Requested and executed funding varied by a factor of 3
  - There is an implied prioritization of applied funding by subsystem (domain - weapons system vs. training system)
  - System age had no appreciable cost impact
  - There is a non-linear relationship between software system size and cost for software maintenance (resource constraints)
  - Measurable software development characteristics available early in the life cycle may be early predictors of software maintenance life cycle cost
Software Change Drivers
(Product CER Calibration Factors)

**External Drivers**
- Operational User
  - Functional modifications
  - Functional additions
  - Functional deletions
- Stakeholder
  - Threat
  - Mission doctrine
  - System interoperability
  - External testing/IV&V
  - External audits
- Mandate
  - Legal/Regulatory/Policy
- Technology
  - Technology obsolescence
  - Infrastructure changes

**Internal Drivers**
- Legacy
  - Technical debt
  - Deferred functionality
- Maintenance Operations
  - Maintainer skill set

**Types of Changes**
- Corrective (identified defects)
- Preventive (latent faults)
- Perfective (functional enhancements)
- Adaptive (new HW/SW environments)
Software Maintenance Change Drivers

- The factors that cause the changes that are made to an operational software baseline

- Software changes characterized by:
  - source of change
  - type of change
  - impact of change (scope, complexity, etc.)
  - priority of implementation

- Change drivers can impact the operational software configuration, associated sustaining engineering activities, and the implemented maintenance infrastructure

- Most change requirements are allocated to planned releases per time period

- Deferred change requirements are defined as “backlog”

- Drivers with different characteristics are costed differently
Software Maintenance Product Sizing

- Overall concept of “E size” for maintenance products
- E Size for types of changes (different drivers)
- Composite E Size for different types of releases
- Concept - profile the program maintenance strategy based on numbers and types of releases (variable costs) - add semi-fixed costs related to supporting the fielded software (costs not based on amount of work to construct a release)
- Variable costs driven by type and number of product changes
- Semi-fixed costs driven by number of deployed platforms, domain, operating tempo, etc.
- Similar to software development sizing “E Sloc” but focused on change drivers/release content and support scope
Integrated Cost Estimation Life Cycle Model

- **Two key components**

- **Software maintenance cost risk-uncertainty model**
  - Risk information model
  - Risk interdependency analysis model
  - Cost estimation application process model

- **Integrated life cycle cost allocation model**
  - Program software life cycle product-activity profile
    - Post-development software releases
    - Fielded software sustaining engineering support
    - Software maintenance site infrastructure
  - Maintenance resource-funding allocation model(s)
  - Integrated cost estimation results
Notional Software Maintenance Life-Cycle Cost Allocation Model
Program Software Maintenance Release Profile

Cycles are different - platform dependent
User needs drive release content
Program Information Requirements

- System and software maintenance strategy
- Included products and activities
- Software release profile - PDSS/PPSS
  - Release type and schedule (high level)
  - Release profile stability
  - User base - deployment strategy
  - Configuration information
  - Performing organizations
  - Projected operational environment
  - Program and software risk analysis
  - Historical profile information (later)
Implementation Guidance for Depot-Level Maintenance

- 10 U.S.C. § 2460 - Definition of depot-level maintenance and repair has been changed (as of 05 April 2012) to state that software maintenance is defined as:

"The repair, adaptive modifications or upgrades, change events made to operational software, integration and testing."

Interpretation/Guidance: The scope of this statute, as indicated by the title, is limited to depot-level maintenance and repair. Therefore, the language does not apply to hardware or software modifications that are not maintenance in nature. Specifically, modifications designed to enhance performance or add functional capability are not considered maintenance and may be excluded in the application of all depot maintenance statutory provisions, however, they may be included if the modifications are being executed consistent with past depot practices.
Acquisition Life-Cycle Model

System Life Cycle Acquisition Process

Combat Developer
PM Total Life Cycle System Manager
Air Force Materiel Command

Acquisition Framework

High Ability to Influence LCC (70-75% of Cost Decisions Made)
Less Ability to Influence LCC (85% of Cost Decisions Made)
Little Ability to Influence LCC (95-95% of Cost Decisions Made)
Minimum Ability to Influence LCC (85% of Cost Decisions Made)

28% Life Cycle Cost

72% Life Cycle Cost
10 USC 2366a states that:

“Milestone Decision Authority must certify that a determination of applicability of core depot-level maintenance and repair capabilities requirements has been made before a Major Defense Acquisition Program (MDAP) may receive Milestone A approval.

- **Interpretation/Guidance:** The scope of this analysis is limited to the determination whether the criteria detailed in 10 U.S.C. § 2464 apply to the weapon system or item of military equipment being procured. This analysis should be limited to determining whether the entire system, or particular subsets of the system, is considered to require core depot maintenance capability.

- The Cost Analysis Requirements Description (CARD) should have described or outlined a system sustainment concept, but details are not required for Milestone A.
**Milestone B**

- The Cost Analysis Requirements Description (CARD) should now describe:
  - System sustainment concept
  - System logistics concept
    - Maintenance concept
    - Supply management concept
    - Transportation concept
  - Software maintenance concept
  - System training concept

- Develop Initial Capabilities Document (ICD) / Capability Development Document (CDD) / Capability Production Document (CPD)

- System Engineering Plan

- Life Cycle Sustainment Plan (DoD 5000.02) documenting the PM's plan for formulating, implementing and executing the sustainment strategy is required to achieving and maintaining the Sustainment KPP/KSAs needs to be developed as part of the official program/project Acquisition Strategy.
Initial Capabilities Document (ICD) / Capability Development Document (CDD) / Capability Production Document (CPD)

These documents are the sponsor’s means to specify authoritative and testable, performance capabilities for the program. The ICD prefaces a system materiel decision and evolves into the CDD, which prioritizes KPP and subset KSA performance capability design and development parameters. The baseline CPD is finalized after the system level Critical Design Review and before Milestone C. In addition to supportability related KPP/KSAs, the ICD, CDD, and CPD narrative should also address the following:

- System maintenance/support concepts and usage scenarios
- Operational and support environments. This should include the general support categories relative to the logistics support infrastructure (remote sites, organic depots, commercial facilities, air bases or ship yards, etc. without naming specific locations)
- Expected durations of support
- Support or maintenance effectiveness metrics and key enablers, such as diagnostics/prognostics
- Conditions conducive to joint sustainment and to performance-based support strategies.
Milestone C

- **The Cost Analysis Requirements Description (CARD) is updated** (and before Full Rate Production Decision Review)

- **Initial Capabilities Document (ICD) / Capability Development Document (CDD) / Capability Production Document (CPD) are updated**

- **The Systems Engineering Plan is updated**

- **Life Cycle Sustainment Plan (DoD 5000.02) is updated**
Post Milestone C

- Life Cycle Sustainment Plan (DoD 5000.02) is updated at a minimum every 5 years, or when:
  - Subsequent increments are approved and funded to reflect how the support strategy will evolve to support multiple configurations
  - Significant changes are required to the product support package to achieve the objective sustainment metrics including major support provider changes.

- If a decision has been made that a system will replace another, a Replaced System Sustainment Plan for the existing system is required that includes at least the budget estimates required to sustain the existing system until the new system assumes the majority of mission responsibility.
Software Maintenance Cost-Capability Framework

Cost
- applied resources
- maintenance process performance
- risk

Personnel

Facilities

Software Products

Software Services

Capability
- functional domain
- system interfaces
- operational environment

Software Changes
- software baseline architecture - design
- requirements - change drivers
How much should software maintenance cost?

"It's All About the Money", Dr. Chien Huo, CAPE, November 2011
Army Software Maintenance Study

- Sponsored by U.S. Army Office of the Deputy Assistant Secretary of the Army for Cost & Economics (ODASA-CE)
- Collaborative environment - diverse perspectives
- Army, Air Force, Navy, OSD, Industry participation
- Initial focus on ACAT-1 weapons system software maintenance costs
- Technical approach based on measurement and estimation best practices
Contact Information

Jack McGarry  
U.S. Army RDECOM - ARDEC  
Quality Engineering & System Assurance  
RDAR-QES-A  
Building 62  
Picatinny Arsenal, NJ 07806-5000  
(973) 724-7007  
john.mcgarry@us.army.mil

Dr. Robert Charette  
ITABHI Corporation  
11609 Stonewall Jackson Drive  
Spotsylvania, VA 22551-4668  
(540) 972-8150  
charette@itabhi.com