Ilities Tradespace and Affordability Program (iTAP)

By
Barry Boehm, USC
Russell Peak, GTRI

6th Annual SERC Sponsor Research Review
December 4, 2014
Georgetown University
School of Continuing Studies
640 Massachusetts Ave NW,
Washington, DC

www.sercuarc.org
• Models, Methods, Processes, and Tools (MMPTs)
  – MMPT Integration Framework: GTRI
  – SysML-COSYSMO Integration: GTRI, USC
  – Interactive, Model-Centric SE: MIT
  – ISR UAV Tradespace MMPTs: AFIT, NPS
  – Holistic Model-Centric SE: Stevens
  – Agile-Lean-Kanban SE: Stevens, USC, Auburn
  – Set-Based Design: WSU, PSU

• iTAP Foundations: ilities Ontology Views
  – Stakeholder Value-Based, Means-Ends View: USC
  – Change-Oriented View: MIT
  – Formal Methods Views: UVirginia
Critical nature of the ilities

- Or non-functional requirements; quality attributes
- Major source of project overruns, failures
- Significant source of stakeholder value conflicts
- Poorly defined, understood
- Underemphasized in project management
- Need for ilities ontology

• Ility synergies and conflicts analysis
  - Stakeholder value-based, means-ends hierarchy
  - Synergies and Conflicts matrix and expansions

• SysML-COSYSMO Integration: GTRI (Peak), USC (Lane)
Importance of ility Tradeoffs

Major source of DoD system overruns

• System ilities have systemwide impact
  – System elements generally just have local impact
• ilities often exhibit asymptotic behavior
  – Watch out for the knee of the curve
• Best architecture is a discontinuous function of ility level
  – “Build it quickly, tune or fix it later” highly risky
  – Large system example below
Value Conflicts: Security IPT

- Single-agent key distribution; single data copy
  - Reliability: single points of failure

- Elaborate multilayer defense
  - Performance: 50% overhead; real-time deadline problems

- Elaborate authentication
  - Usability: delays, delegation problems; GUI complexity

- Everything at highest level
  - Modifiability: overly complex changes, recertification
Proliferation of Definitions: Resilience


- Ecology and Society Organization Resilience variants: Original-ecological, Extended-ecological, Walker et al. list, Folke et al. list; Systemic-heuristic, Operational, Sociological, Ecological-economic, Social-ecological system, Metaphoric, Sustainabilty-related

- Variants in resilience outcomes
  - Returning to original state; Restoring or improving original state; Maintaining same relationships among state variables; Maintaining desired services; Maintaining an acceptable level of service; Retaining essentially the same function, structure, and feedbacks; Absorbing disturbances; Coping with disturbances; Self-organizing; Learning and adaptation; Creating lasting value
Example of Current Practice

• “The system shall have a Mean Time Between Failures of 10,000 hours”

• What is a “failure?”
  – 10,000 hours on liveness
  – But several dropped or garbled messages per hour?

• What is the operational context?
  – Base operations? Field operations? Conflict operations?

• Most management practices focused on functions
  – Requirements, design reviews; traceability matrices; work breakdown structures; data item descriptions; earned value management

• What are the effects on other –ilities?
  – Cost, schedule, performance, maintainability?
Need for ilities Ontology

A structural framework for organizing information about a topic of interest

- Oversimplified one-size-fits all definitions
  - ISO/IEC 25010, Reliability: the degree to which a system, product, or component performs specified functions under specified conditions for a specified period of time
  - OK if specifications are precise, but increasingly “specified conditions” are informal, sunny-day user stories. Satisfying just these will pass ISO/IEC, but fail on rainy-day use cases
  - Need to reflect that different stakeholders rely on different capabilities (functions, performance, flexibility, etc.) at different times and in different environments
- Proliferation of definitions, as with Resilience
- Weak understanding of inter-ility relationships
  - Synergies and Conflicts, as with Security
Initial SERC ilities Ontology

- Modified version of IDEF5 ontology framework
  - Classes, Subclasses, and Individuals
  - States, Processes, and Relations
- Top classes cover stakeholder value propositions
  - Mission Effectiveness, Resource Utilization, Dependability, Flexibility
- Subclasses identify means for achieving higher-class ends
  - Means-ends, one-to-many for top classes
  - Ideally mutually exclusive and exhaustive, but some exceptions
  - Many-to-many for lower-level subclasses
- States, Processes, and Relations cover sources of ility variation
  - States: Internal (beta-test); External (rural, temperate, sunny)
  - Processes: Operational scenarios (normal vs. crisis; experts vs. novices)
  - Relations: Impact of other ilities (security as above, synergies & conflicts)
• Critical nature of the ilities
  – Or non-functional requirements; quality attributes
  – Major source of project overruns, failures
  – Significant source of stakeholder value conflicts
  – Poorly defined, understood
  – Underemphasized in project management
  – Need for ilities ontology

Ility synergies and conflicts analysis
  – Stakeholder value-based, means-ends hierarchy
  – Synergies and Conflicts matrix and expansions

• SysML-COSYSMO Integration: GTRI (Peak), USC (Lane)
Stakeholder value-based, means-ends hierarchy

• Mission operators and managers want improved Mission Effectiveness
  – Involves Physical Capability, Cyber Capability, Human Usability, Speed, Accuracy, Impact, Mobility, Scalability, Versatility, Interoperability

• Mission investors and system owners want Mission Cost-Effectiveness
  – Involves Cost, Duration, Personnel, Scarce Quantities (capacity, weight, energy, ...); Manufacturability, Sustainability

• All want system Dependability: cost-effective defect-freedom, availability, and safety and security for the communities that they serve
  – Involves Reliability, Availability, Maintainability, Survivability, Safety, Security

• In an increasingly dynamic world, all want system Flexibility: to be rapidly and cost-effectively changeable
  – Involves Modifiability, Tailorability, Adaptability
• Mission Effectiveness expanded to 4 elements
  – Physical Capability, Cyber Capability, Interoperability, Other
    Mission Effectiveness (including Usability as Human Capability)

• Synergies and Conflicts among the 7 resulting elements
  identified in 7x7 matrix
  – Synergies above main diagonal, Conflicts below

• Work-in-progress tool will enable clicking on an entry and
  obtaining details about the synergy or conflict
  – Ideally quantitative; some examples next

• Still need synergies and conflicts within elements
  – Example 3x3 Dependability subset provided
<table>
<thead>
<tr>
<th>Flexibility</th>
<th>Dependability</th>
<th>Mission Effectiveness</th>
<th>Resource Utilization</th>
<th>Physical Capability</th>
<th>Cyber Capability</th>
<th>Interoperability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domain architcturing within domain</td>
<td>Modularity</td>
<td>Adaptable</td>
<td>Adaptable</td>
<td>Adaptable</td>
<td>Adaptable</td>
<td>Adaptable</td>
</tr>
<tr>
<td>Self Adaptive</td>
<td>Service oriented</td>
<td>Agile methods</td>
<td>Automated I/O validation</td>
<td>Spare capacity</td>
<td>Spare capacity</td>
<td>Spare capacity</td>
</tr>
<tr>
<td>Smart monitoring</td>
<td>Loose coupling for sustainability</td>
<td>User programmability</td>
<td>Product line architectures</td>
<td>Staffing, Empowering</td>
<td>Staffing, Empowering</td>
<td>Staffing, Empowering</td>
</tr>
<tr>
<td>Spare Capacity</td>
<td>Versatility</td>
<td>Automated aids</td>
<td>Automated aids</td>
<td>Automated aids</td>
<td>Automated aids</td>
<td>Automated aids</td>
</tr>
<tr>
<td>Use software vs. hardware</td>
<td>Reliability</td>
<td>Lightweight agility</td>
<td>Redundancy</td>
<td>Value prioritizing</td>
<td>Value prioritizing</td>
<td>Value prioritizing</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dependability</th>
<th>Accreditation</th>
<th>Agile methods assurance</th>
<th>Encryption</th>
<th>Many options</th>
<th>Multi-domain modifiability</th>
<th>Multi-level security</th>
<th>Survivability</th>
<th>User programmability</th>
<th>Use software vs. hardware</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accreditation</td>
<td>Adaptive</td>
<td>Automated aids</td>
<td>Automated I/O validation</td>
<td>Domain architcturing within domain</td>
<td>Domain architcturing within domain</td>
<td>Domain architcturing within domain</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Agile methods assurance</td>
<td>Lightweight</td>
<td>Automated aids</td>
<td>Automated aids</td>
<td>Staffing, Empowering</td>
<td>Staffing, Empowering</td>
<td>Staffing, Empowering</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Encryption</td>
<td>Many options</td>
<td>Agile methods</td>
<td>Automated I/O validation</td>
<td>Domain architcturing within domain</td>
<td>Domain architcturing within domain</td>
<td>Domain architcturing within domain</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Many options</td>
<td>Service oriented</td>
<td>Agile methods</td>
<td>Automated I/O validation</td>
<td>Domain architcturing within domain</td>
<td>Domain architcturing within domain</td>
<td>Domain architcturing within domain</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Multi-domain modifiability</td>
<td>松散耦合</td>
<td>用于可持续性的松散耦合</td>
<td>用户可编程性</td>
<td>产品线架构</td>
<td>雇佣，激励</td>
<td>雇佣，激励</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Multi-level security</td>
<td>Survivability</td>
<td>Agile methods</td>
<td>Automated I/O validation</td>
<td>Domain architcturing within domain</td>
<td>Domain architcturing within domain</td>
<td>Domain architcturing within domain</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Survivability</td>
<td>Value prioritizing</td>
<td>Agile methods</td>
<td>Automated I/O validation</td>
<td>Domain architcturing within domain</td>
<td>Domain architcturing within domain</td>
<td>Domain architcturing within domain</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>User programmability</td>
<td>Value prioritizing</td>
<td>Agile methods</td>
<td>Automated I/O validation</td>
<td>Domain architcturing within domain</td>
<td>Domain architcturing within domain</td>
<td>Domain architcturing within domain</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use software vs. hardware</td>
<td>Value prioritizing</td>
<td>Agile methods</td>
<td>Automated I/O validation</td>
<td>Domain architcturing within domain</td>
<td>Domain architcturing within domain</td>
<td>Domain architcturing within domain</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Mission Effectiveness</th>
<th>Autonomy vs. Usability</th>
<th>Modularity slowdowns</th>
<th>Multi-domain architecture interoperability conflicts</th>
<th>Versatility vs. Usability</th>
<th>Resource Utilization</th>
<th>Physical Capability</th>
<th>Cyber Capability</th>
<th>Interoperability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Autonomy vs. Usability</td>
<td>Anti-tamper</td>
<td>Agile methods scalability</td>
<td>Acquisition Cost</td>
<td>Agile methods scalability</td>
<td>Agile methods scalability</td>
<td>Agile methods scalability</td>
<td>Agile methods scalability</td>
<td>Agile methods scalability</td>
</tr>
<tr>
<td>Modularity slowdowns</td>
<td>Armor vs. Weight</td>
<td>Agile methods scalability</td>
<td>Certification</td>
<td>Agile methods scalability</td>
<td>Agile methods scalability</td>
<td>Agile methods scalability</td>
<td>Agile methods scalability</td>
<td>Agile methods scalability</td>
</tr>
<tr>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Easiest-first development</td>
<td>Agile methods scalability</td>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Agile methods scalability</td>
<td>Agile methods scalability</td>
<td>Agile methods scalability</td>
<td>Agile methods scalability</td>
<td>Agile methods scalability</td>
</tr>
<tr>
<td>Versatility vs. Usability</td>
<td>Redundancy</td>
<td>Agile methods scalability</td>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Agile methods scalability</td>
<td>Agile methods scalability</td>
<td>Agile methods scalability</td>
<td>Agile methods scalability</td>
<td>Agile methods scalability</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Physical Capability</th>
<th>Interoperability</th>
<th>Cyber Capability</th>
<th>Interoperability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Physical architecture or cyber architecture</td>
<td>Over-optimizing</td>
<td>Over-optimizing</td>
</tr>
<tr>
<td>Tight coupling</td>
<td>Physical architecture or cyber architecture</td>
<td>Over-optimizing</td>
<td>Over-optimizing</td>
</tr>
<tr>
<td>Use software vs. hardware</td>
<td>Physical architecture or cyber architecture</td>
<td>Over-optimizing</td>
<td>Over-optimizing</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Cyber Capability</th>
<th>Interoperability</th>
<th>Physical Capability</th>
<th>Interoperability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agile Methods scalability</td>
<td>Over-optimizing</td>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Over-optimizing</td>
</tr>
<tr>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Over-optimizing</td>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Over-optimizing</td>
</tr>
<tr>
<td>Over-optimizing</td>
<td>Over-optimizing</td>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Over-optimizing</td>
</tr>
<tr>
<td>Tight coupling</td>
<td>Over-optimizing</td>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Over-optimizing</td>
</tr>
<tr>
<td>Use software vs. hardware</td>
<td>Over-optimizing</td>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Over-optimizing</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Interoperability</th>
<th>Physical Capability</th>
<th>Cyber Capability</th>
<th>Interoperability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Encryption interoperability</td>
<td>Over-optimizing</td>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Over-optimizing</td>
</tr>
<tr>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Over-optimizing</td>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Over-optimizing</td>
</tr>
<tr>
<td>Over-optimizing</td>
<td>Over-optimizing</td>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Over-optimizing</td>
</tr>
<tr>
<td>Tight vs. Loose coupling</td>
<td>Over-optimizing</td>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Over-optimizing</td>
</tr>
<tr>
<td>Use software vs. hardware</td>
<td>Over-optimizing</td>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Over-optimizing</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Interoperability</th>
<th>Physical Capability</th>
<th>Cyber Capability</th>
<th>Interoperability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Encryption interoperability</td>
<td>Over-optimizing</td>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Over-optimizing</td>
</tr>
<tr>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Over-optimizing</td>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Over-optimizing</td>
</tr>
<tr>
<td>Over-optimizing</td>
<td>Over-optimizing</td>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Over-optimizing</td>
</tr>
<tr>
<td>Tight vs. Loose coupling</td>
<td>Over-optimizing</td>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Over-optimizing</td>
</tr>
<tr>
<td>Use software vs. hardware</td>
<td>Over-optimizing</td>
<td>Multi-domain architecture interoperability conflicts</td>
<td>Over-optimizing</td>
</tr>
</tbody>
</table>
Software Development Cost vs. Reliability

MTBF (hours) | Very Low | Low | Nominal | High | Very High
-------------|----------|-----|---------|------|---------
1            | 0.82     | 0.92| 1.0     | 1.10 | 1.26    
10           |          |     |         |      |         
300          |          |     |         |      |         
10,000       |          |     |         |      |         
300,000      |          |     |         |      |         

COCOMO II RELY Rating

Relative Cost to Develop

0.8

0.9

1.0

1.1

1.2

1.3

1.4

Very Low

Low

Nominal

High

Very High
Software Ownership Cost vs. Reliability

Operational-defect cost at Nominal dependability = Software life cycle cost

Operational - defect cost = 0

MTBF (hours)       1                     10                300                 10,000           300,000

Relative Cost to Develop, Maintain, Own and Operate

Very Low    Low       Nominal    High          Very High

VL = 2.55
L = 1.52

COCOMO II RELY Rating

70% Maint.

12-04-2014 15SERC SSRR
<table>
<thead>
<tr>
<th></th>
<th>Security</th>
<th>Reliability</th>
<th>Maintainability</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Security</strong></td>
<td></td>
<td>Confidentiality, Integrity, Availabilty</td>
<td>Certification</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Assurance Cases</td>
<td>Diagnosability</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Certification</td>
<td>Integrity, Availability</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Failure Modes and Effects Analysis</td>
<td>Repairability</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fault Tree Analysis</td>
<td>Smart Monitoring</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Recertification</td>
<td>Spare Capacity</td>
</tr>
<tr>
<td><strong>Reliability</strong></td>
<td>Non-redundancy (For Security)</td>
<td></td>
<td>Accessibility</td>
</tr>
<tr>
<td></td>
<td>Redundancy (For Reliability)</td>
<td></td>
<td>Certification</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Diagnosability</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Repairability</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Smart Monitoring</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Spare Capacity</td>
</tr>
<tr>
<td><strong>Maintainability</strong></td>
<td>Accessibility</td>
<td>Armor</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Compartmentalization</td>
<td>Recertification</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Encryption</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Recertification</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Conclusions

• Ilities or non-functional requirements are success-critical
  – Major source of project overruns, failures
  – Significant source of stakeholder value conflicts
  – Poorly defined, understood
  – Underemphasized in project management

• Ilities ontology clarifies nature of ilities
  – Using value-based, means-ends hierarchy
  – Identifies sources of variation: states, processes, relations
  – Relations enable ility synergies and conflicts identification

• Continuing SERC research creating tools, formal definitions
• Critical nature of the ilities
  – Or non-functional requirements; quality attributes
  – Major source of project overruns, failures
  – Significant source of stakeholder value conflicts
  – Poorly defined, understood
  – Underemphasized in project management
  – Need for ilities ontology

• Ility synergies and conflicts analysis
  – Stakeholder value-based, means-ends hierarchy
  – Synergies and Conflicts matrix and expansions

SysML-COSYSMO Integration: GTRI (Peak), USC (Lane)
• Implemented cost modeling concepts as SysML building blocks
  – Concepts per SoS/COSYSMO work by Lane, Valerdi, Boehm, et al.
  – SysML knowledge capture that is reusable, modular, more complete

• Enables integration with complex SysML models

• Successfully applied building blocks to two healthcare SoS case studies

• Provides key step towards affordability trade studies involving diverse “-ilities”
COSYSMO/SoS Concepts Implemented as SysML Building Blocks: Selected SysML Diagrams

Generic and useful to analyze practically any SoS (see case study applications in next slides)
Healthcare SoS Case Study1 [Lane 2009]

Original Document & Spreadsheet Views

Recursive application of COSYSMO concepts for each constituent system in SoS, plus considerations specific to SoS top-level.

- Laboratory System
- Imaging Management System
- Pharmacy System

4 main systems; SoS top reqs: 50; CS reqs: 150 SoS, 20 non-SoS (220 reqs grand total)

**Table:**

<table>
<thead>
<tr>
<th>Aspect</th>
<th>Formula</th>
<th>Calculated Effort</th>
</tr>
</thead>
<tbody>
<tr>
<td>SoSE effort (Equation 5)</td>
<td>Effort = 38.55*(((SoSReq / SoSTop)* (SoSTop)* EMSoSReq) + ((SoSReq / SoSTop)* (SoSTop)* EMSoSReq)) / 152</td>
<td>40.41</td>
</tr>
<tr>
<td>Pharmacy System effort</td>
<td>Effort = 38.55*(((1.0CSReq) * ((CSReq)*EMCSReq)) * EMCSReq)) / 152</td>
<td>22.02</td>
</tr>
<tr>
<td>Laboratory System effort</td>
<td>Effort = 38.55*(((1.0CSReq) * ((CSReq)*EMCSReq)) * EMCSReq)) / 152</td>
<td>19.55</td>
</tr>
<tr>
<td>Imaging System effort</td>
<td>Effort = 38.55*(((1.0CSReq) * ((CSReq)*EMCSReq)) * EMCSReq)) / 152</td>
<td>19.55</td>
</tr>
<tr>
<td>New infrastructure component</td>
<td>Effort = 38.55<em>EM</em>size*152</td>
<td>33.43</td>
</tr>
</tbody>
</table>

**Total Effort:** 134.96
Healthcare SoS Case Study 1: Model Subset
Spreadsheet View vs. SysML DNA Signature View

Enter Size Parameters for System of Interest

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Easy</th>
<th>Nominal</th>
<th>Difficult</th>
</tr>
</thead>
<tbody>
<tr>
<td># of System Requirements</td>
<td>10</td>
<td>20</td>
<td>7</td>
</tr>
<tr>
<td># of System Interfaces</td>
<td>10</td>
<td>10</td>
<td>8</td>
</tr>
<tr>
<td># of Algorithms</td>
<td>20</td>
<td>15</td>
<td>10</td>
</tr>
<tr>
<td># of Operational Scenarios</td>
<td>5</td>
<td>5</td>
<td>4</td>
</tr>
</tbody>
</table>

Equivalent size:

- E: 60
- N: 89
- D: 221
- 223

Select Cost Parameters for System of Interest

<table>
<thead>
<tr>
<th>Parameter</th>
<th>VL</th>
<th>VL-L</th>
<th>L</th>
<th>H</th>
<th>E</th>
<th>N</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements Understanding</td>
<td>1.85</td>
<td>1.59</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Architecture Understanding</td>
<td>1.82</td>
<td>1.44</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level of Service Requirements</td>
<td>0.70</td>
<td>0.77</td>
<td></td>
<td></td>
<td>0.82</td>
<td>0.86</td>
<td></td>
</tr>
<tr>
<td>Migration Complexity</td>
<td>0.80</td>
<td>0.85</td>
<td></td>
<td></td>
<td>1.50</td>
<td>1.36</td>
<td></td>
</tr>
<tr>
<td>Technology Risk</td>
<td>1.48</td>
<td>1.34</td>
<td></td>
<td></td>
<td>1.46</td>
<td>1.33</td>
<td></td>
</tr>
<tr>
<td>Documentation</td>
<td>1.33</td>
<td>1.24</td>
<td></td>
<td></td>
<td>1.34</td>
<td>1.25</td>
<td></td>
</tr>
<tr>
<td>Stakeholder team cohesion</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
<td>1.00</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>Personnel/team capability</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
<td>1.00</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>Personnel experience/continuity</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
<td>1.00</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>Process capability</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
<td>1.00</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>Multisite coordination</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
<td>1.00</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>Tool support</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
<td>1.00</td>
<td>1.00</td>
<td></td>
</tr>
</tbody>
</table>

Pro forma parameter values:

- E: 0.5
- N: 1.0
- D: 5.0

Equivalent size:

- E: 0.5
- N: 1.0
- D: 5.0

System Engineering Person Months:

- Pro forma parameter values: 241.8
- Composite effort multiplier: 0.55 1.06

Healthcare IT Network Effort Model
(an infrastructure component; a primitive system;)

Healthcare SoS Case Study 1: Model Subset
Spreadsheet View vs. SysML DNA Signature View

Subset of full model

© Ricardo Valerdi, University of Southern California
### Healthcare SoS Case Study 1: Model Execution

#### Tool for Solving SysML Instance Structures

*(object-oriented spreadsheet-like tool)*

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Causality</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>SoS Total Effort Model</td>
<td>Real</td>
<td>given</td>
<td>38.55</td>
</tr>
<tr>
<td>d_A</td>
<td>Real</td>
<td>given</td>
<td>1.060</td>
</tr>
<tr>
<td>d_B</td>
<td>Real</td>
<td>given</td>
<td>1.060</td>
</tr>
<tr>
<td>constituent systems effort</td>
<td>person-months</td>
<td>ancillary</td>
<td>63.995</td>
</tr>
<tr>
<td>em_SoC-CR</td>
<td>Real</td>
<td>ancillary</td>
<td>2.500</td>
</tr>
<tr>
<td>em_SoC-MR</td>
<td>Real</td>
<td>ancillary</td>
<td>0.466</td>
</tr>
<tr>
<td>Infrastructure components effort</td>
<td>person-months</td>
<td>ancillary</td>
<td>33.433</td>
</tr>
<tr>
<td>ost</td>
<td>Real</td>
<td>given</td>
<td>0.100</td>
</tr>
<tr>
<td>sos_CR</td>
<td>Real</td>
<td>ancillary</td>
<td>50.000</td>
</tr>
<tr>
<td>sos_MR</td>
<td>Real</td>
<td>ancillary</td>
<td>20.000</td>
</tr>
<tr>
<td>sos_Treq</td>
<td>Real</td>
<td>given</td>
<td>52.000</td>
</tr>
<tr>
<td>sos_effort</td>
<td>person-months</td>
<td>ancillary</td>
<td>40.836</td>
</tr>
<tr>
<td>total effort</td>
<td>person-months</td>
<td>target</td>
<td>137.874</td>
</tr>
</tbody>
</table>

#### Top-Level SysML Instances

*(bdd view - after solving in ParaMagic)*

- **sos1**: SoS Total Effort Model
  - total effort: person-months = 137.874
- **cs1-pharmacy-sys**: SoS-affected CS Effort Model
  - effort: person-months = 24.731
- **cs2-lab-mgt-sys**: SoS-affected CS Effort Model
  - effort: person-months = 19.618
- **cs3-imaging-sys**: SoS-affected CS Effort Model
  - effort: person-months = 19.618
- **ic1-hc-network**: Primitive SoS Effort Model
  - effort: person-months = 33.434

No. of variables: 1166
No. of equations: 204
# Healthcare SoS Case Study 1: Implementation Results

Good verification compared to original results

## Original Results Summary [Lane 2009]

(subject to known corrections & round-off)

<table>
<thead>
<tr>
<th>Aspect</th>
<th>Formula</th>
<th>Calculated Effort</th>
</tr>
</thead>
<tbody>
<tr>
<td>SoSE effort (Equation 5)</td>
<td>[ 38.55 \times \left( \frac{\text{SoS}<em>{\text{CR}} / \text{SoS}</em>{\text{Treq}}}{} \right) \times (\text{SoS}<em>{\text{Treq}}^{0.06} \times \text{EM}</em>{\text{SOS,CR}}) + (\text{SoS}<em>{\text{CR}} / \text{SoS}</em>{\text{Treq}})^2 ]</td>
<td>40.41</td>
</tr>
<tr>
<td>Pharmacy System effort (Equation 4)</td>
<td>[ 38.55 \times \left( \frac{1.0 + \text{CS}<em>{\text{Sofreq}}}{1.0} \right) \times (\text{SoS}</em>{\text{SM}} / \text{SoS}<em>{\text{Treq}}) \times (\text{CS}</em>{\text{Treq}}^{0.06} \times \text{EM}<em>{\text{Sofreq,SM}}) + (\text{CS}</em>{\text{Sofreq}} / \text{CS}<em>{\text{Treq}})^2 \times (\text{CS}</em>{\text{Treq}}^{0.06} \times \text{EM}<em>{\text{Sofreq,SM}}) \times \text{EM}</em>{\text{CS,CR,SM}} / 152 ]</td>
<td>24.65</td>
</tr>
<tr>
<td>Laboratory System effort (Equation 4)</td>
<td>[ 38.55 \times \left( \frac{1.0 + \text{CS}<em>{\text{Sofreq}}}{1.0} \right) \times (\text{SoS}</em>{\text{SM}} / \text{SoS}<em>{\text{Treq}}) \times (\text{CS}</em>{\text{Treq}}^{0.06} \times \text{EM}<em>{\text{Sofreq,SM}}) + (\text{CS}</em>{\text{Sofreq}} / \text{CS}<em>{\text{Treq}})^2 \times (\text{CS}</em>{\text{Treq}}^{0.06} \times \text{EM}<em>{\text{Sofreq,SM}}) + (\text{CS}</em>{\text{Treq}}^{0.06} \times \text{EM}<em>{\text{CS,CR,SM}}) \times \text{EM}</em>{\text{CS,SM}} / 152 ]</td>
<td>19.55</td>
</tr>
<tr>
<td>Imaging System effort (Equation 4)</td>
<td>[ 38.55 \times \left( \frac{1.0 + \text{CS}<em>{\text{Sofreq}}}{1.0} \right) \times (\text{SoS}</em>{\text{SM}} / \text{SoS}<em>{\text{Treq}}) \times (\text{CS}</em>{\text{Treq}}^{0.06} \times \text{EM}<em>{\text{Sofreq,SM}}) + (\text{CS}</em>{\text{Sofreq}} / \text{CS}<em>{\text{Treq}})^2 \times (\text{CS}</em>{\text{Treq}}^{0.06} \times \text{EM}<em>{\text{Sofreq,SM}}) + (\text{CS}</em>{\text{Treq}}^{0.06} \times \text{EM}<em>{\text{CS,CR,SM}}) \times \text{EM}</em>{\text{CS,SM}} / 152 ]</td>
<td>19.55</td>
</tr>
<tr>
<td>New infrastructure component effort (Equation 1)</td>
<td>[ 38.55 \times \text{EM}^{*} \times (\text{size})^{0.06} / 152 ]</td>
<td>33.43</td>
</tr>
</tbody>
</table>

Total Effort: **137.59**

## SysML-Based Results Summary

<table>
<thead>
<tr>
<th>Effort Model</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>sos1 : SoS Total Effort Model</td>
<td>[ 40.4873466062356 \times \text{person-months} = 137.87437862723192 ]</td>
</tr>
<tr>
<td>cs1-pharmacy-sys : SoS-affected CS Effort Model</td>
<td>[ 24.73153958895236 \times \text{person-months} = 19.61184247237522 ]</td>
</tr>
<tr>
<td>cs2-lab-mgt-sys : SoS-affected CS Effort Model</td>
<td>[ 19.61184247237522 \times \text{person-months} = 19.61184247237522 ]</td>
</tr>
<tr>
<td>cs3-imaging-sys : SoS-affected CS Effort Model</td>
<td>[ 19.61184247237522 \times \text{person-months} = 19.61184247237522 ]</td>
</tr>
<tr>
<td>ic1-hc-network : Primitive SOI Effort Model</td>
<td>[ 33.433419257466774 \times \text{person-months} = 33.433419257466774 ]</td>
</tr>
</tbody>
</table>

No. of variables: **1166**  
No. of equations: **204**
Healthcare SoS Case Study 1: Full Model
SysML DNA Signature View

Healthcare SoS Effort Model (a top-level SoS)

Healthcare IT Network Effort Model
(an infrastructure component; a primitive system;)

Pharmacy System Effort Model
(a constituent system)

Cost Drivers of SoS Capability Reqs

Size Drivers of SoS Capability Reqs
Healthcare SoS Case Study2 [Lane et al.]

Original Document & Spreadsheet Views

6 main systems; SoS top reqs: 130; CS reqs: 375 SoS, 175 non-SoS (680 reqs grand total)

Recursively uses COSYSMO, and adds SoS aspects.

Using the values in Table 6-3, the total SE effort calculation is as shown

Equation 6-6: SoSE Effort

\[
\text{SE Effort} = SE_{\text{Infrastructure}} + \text{SE}_{\text{Pat Management}} + \text{SE}_{\text{Pharmacy}} + \text{SE}_{\text{Telemetry}} + \text{SE}_{\text{Imaging}} + \text{SE}_{\text{Laboratory}}
\]

Table 6-2. Health Care SoS SOA Capability Size Overview

<table>
<thead>
<tr>
<th>SoS Product (CS)</th>
<th>SoS SOA Requirements (SoS, etc.)</th>
<th>Concurrent Product Upgrade Size</th>
<th>Total Requirements for CS (CS\text{reqs} + SoS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOA Infrastructure</td>
<td>Infrastructure characteristics: 20</td>
<td>None (new constituent system)</td>
<td>30</td>
</tr>
<tr>
<td>Patient Management System</td>
<td>Shared data standards: 10</td>
<td>30 requirements</td>
<td>140</td>
</tr>
<tr>
<td>Pharmacy System</td>
<td>Infrastructure characteristics: 10</td>
<td>60 requirements</td>
<td>145</td>
</tr>
<tr>
<td>Telemetry System</td>
<td>Shared data standards: 10</td>
<td>25 requirements</td>
<td>75</td>
</tr>
<tr>
<td>Imaging Management System</td>
<td>Shared data standards: 10</td>
<td>50 requirements</td>
<td>95</td>
</tr>
<tr>
<td>Laboratory System</td>
<td>Shared data standards: 10</td>
<td>10 requirements</td>
<td>65</td>
</tr>
</tbody>
</table>

Table 6-3. SoSE Effort Equation Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>CoSE</th>
<th>SE_{\text{Infrastructure}}</th>
<th>SE_{\text{Patient Management System}}</th>
<th>SE_{\text{Pharmacy}}</th>
<th>SE_{\text{Telemetry}}</th>
<th>SE_{\text{Imaging}}</th>
<th>SE_{\text{Laboratory}}</th>
</tr>
</thead>
<tbody>
<tr>
<td>CoSE_{\text{Base}}</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CoSE_{\text{Opt}}</td>
<td>5</td>
<td>140</td>
<td>145</td>
<td>75</td>
<td>95</td>
<td>65</td>
<td>65</td>
</tr>
</tbody>
</table>

\[
\text{SE Effort} = 38.55 (\text{CoSE}_{\text{Base}} + \text{CoSE}_{\text{Opt}}) + 38.55 (\text{SE}_{\text{Infrastructure}} + \text{SE}_{\text{Patient Management System}} + \text{SE}_{\text{Pharmacy}} + \text{SE}_{\text{Telemetry}} + \text{SE}_{\text{Imaging}} + \text{SE}_{\text{Laboratory}})
\]
Healthcare SoS Case Study 2: Results Verification

Original Calculations & Results [Lane et al.]

Using the values in Table 6-3, the total SE upgrade effort calculation is 309.5 person months which is the sum of equations 6-6 through 6-12 shown below.

**Equation 6-6: SoSE Effort**

\[
\text{SoSE Effort} = 38.55*[(130/138.75) * (138.75)^{1.06} * 2.50] + [(8.75/138.75) * (138.75)^{1.06} * 0.47]] / 152
\]

= 112 person months

**Equation 6-7: SOA Infrastructure Constituent System SE Effort**

\[
\text{SOA Infrastructure Constituent System SE Effort} = 38.55*[(1.0+0.30) * ((30/30) * (30)^{0.6} * 1.62)] / 152
\]

= 19.05 person months

**Equation 6-8: Patient Management Constituent System SE Effort**

\[
\text{Patient Management Constituent System SE Effort} = 39.47*[(1.0+0.10) * ((110/140) * (140)^{0.6} * 1.06) + (30/140) * (140)^{0.6} * 0.72]] / 152
\]

= 52.44 person months

**Equation 6-9: Pharmacy Constituent System SE Effort**

\[
\text{Pharmacy Constituent System SE Effort} = 39.25*[(1.0+0.10) * ((85/145) * (145)^{0.6} * 1.06) + (60/145) * (145)^{0.6} * 0.67]] / 152
\]

= 46.62 person months

**Equation 6-10: Telemetry Constituent System SE Effort**

\[
\text{Telemetry Constituent System SE Effort} = 36.82*[(1.0+0.10) * ((50/75) * (75)^{0.6} * 1.06) + (25/75) * (75)^{0.6} * 0.831]] / 152
\]

= 24.57 person months

**Equation 6-11: Imaging Management Constituent System SE Effort**

\[
\text{Imaging Management Constituent System SE Effort} = 38.55*[(1.0+0.10) * ((45/95)]]
\]

= 29.74 person months

**Equation 6-12: Laboratory Constituent System SE Effort**

\[
\text{Laboratory Constituent System SE Effort} = 38.95*[(1.0+0.10) * ((55/65)]]
\]

= 24.48 person months

Good comparison (subject to errors in round-off)

Top-Level SysML Instances

(bdd view - after solving in ParaMagic)

No. of variables: 1830
No. of equations: 320

---

Original Schematic

- Patient Management System
- Laboratory System
- Imaging Management System
- Pharmacy System
- Telemetry System
Applications & Candidate Future Case Studies

• Analysis of alternatives
  – Subsystem/component upgrades
  – Levels of capability option performance within SoS
  – Interoperability assessments for alternatives
• System/component retirement (or replacement) assessments
• Capabilities vs. costs

Case: Emergency Response SoS

Case: Military Operations SoS
Summary & Impact

**SysML/MBSE Approach**

- Created cost modeling building blocks in SysML
  - Leveraging COSYSMO/COCOMO legacy and experiences
- Successfully validated via two healthcare SoS case studies:
  - Base complexity (Case 1) and increased complexity (Case 2)
- Benefits
  - Enables better knowledge capture
    - More modular, reusable, precise, maintainable, complete (e.g., units), ...
    - Acausal; better verification & validation vs. spreadsheets; ...
  - Enables swapping in/out alternative subsystem designs
  - Provides patterns that are easy-to-apply with many systems/SoS
- Can integrate with existing body of system models
  - Executable system models in SysML, DoDAF/UPDM, ...
  - Methods to automate this integration are WIP in RT113/ITAP (CY2015)
- Provides key step towards affordability trade studies involving diverse “-ilities”
Backup charts
COCOMO II-Based Tradeoff Analysis
Better, Cheaper, Faster: Pick Any Two?

- For 100-KSLOC set of features
- Can “pick all three” with 77-KSLOC set of features

(RELY, MTBF (hours))
- (VL, 1)
- (L, 10)
- (N, 300)
- (H, 10K)
- (VH, 300K)
Costing Insights: COCOMO II Productivity Ranges

Scale Factor Ranges: 10, 100, 1000 KSLOC

Product Complexity (CPLX)
Analyst Capability (ACAP)
Programmer Capability (PCAP)
Time Constraint (TIME)
Personnel Continuity (PCON)
Required Software Reliability (RELY)
Documentation Match to Life Cycle Needs (DOCU)
Multi-Site Development (SITE)
Applications Experience (AEXP)
Platform Volatility (PVOL)
Use of Software Tools (TOOL)
Platform Experience (PEXP)
Architecture and Risk Resolution (RESL)
Precedentedness (PREC)
Develop for Reuse (RUSE)
Team Cohesion (TEAM)
Development Flexibility (FLEX)

Staffing
Teambuilding
Continuous Improvement

Productivity Range

Scale Factor Ranges: 10, 100, 1000 KSLOC
COSYSMO Sys Engr Cost Drivers

Teambuilding
- Documentation: 1.64
- # and diversity of installations/platforms: 1.87
- Tool Support: 1.93
- Multisite coordination: 1.93
- Migration Complexity: 1.93
- # of recursive levels in the design: 1.93

Continuous Improvement
- Process capability: 2.16
- Personnel experience/continuity: 2.21
- Stakeholder team cohesion: 2.3
- Personnel/team capability: 2.3
- Architecture Understanding: 2.52
- Technology Risk: 2.61
- Level of Service Requirements: 2.98

Staffing
- Requirements Understanding: 3.12

Effort Multiplier Ratio (EMR)