SYSTEMS ENGINEERING
Research Center

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

Context: SERC SEMT Research Area

Systems Engineering and Management Transformation

- 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

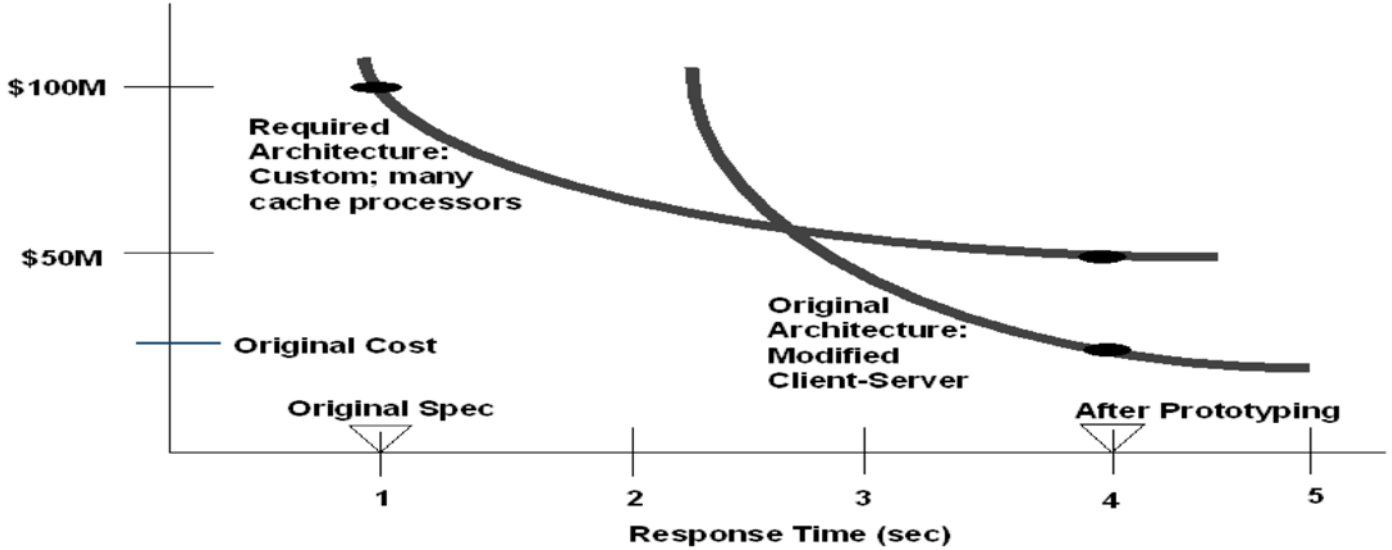
Ilities Tradespace and Affordability Analysis

- 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



SYSTEMS ENGINEERING Research Center

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

- **Wikipedia Resilience variants: Climate, Ecology, Energy Development, Engineering and Construction, Network, Organizational, Psychological, Soil**

- **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**

SYSTEMS ENGINEERING
Research Center

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

SYSTEMS ENGINEERING Research Center

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, Flexibiity
- **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)

SYSTEMS ENGINEERING Research Center

Ilities Tradespace and Affordability Analysis

- 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)

SYSTEMS ENGINEERING Research Center

Stakeholder value-based, means-ends hierarchy

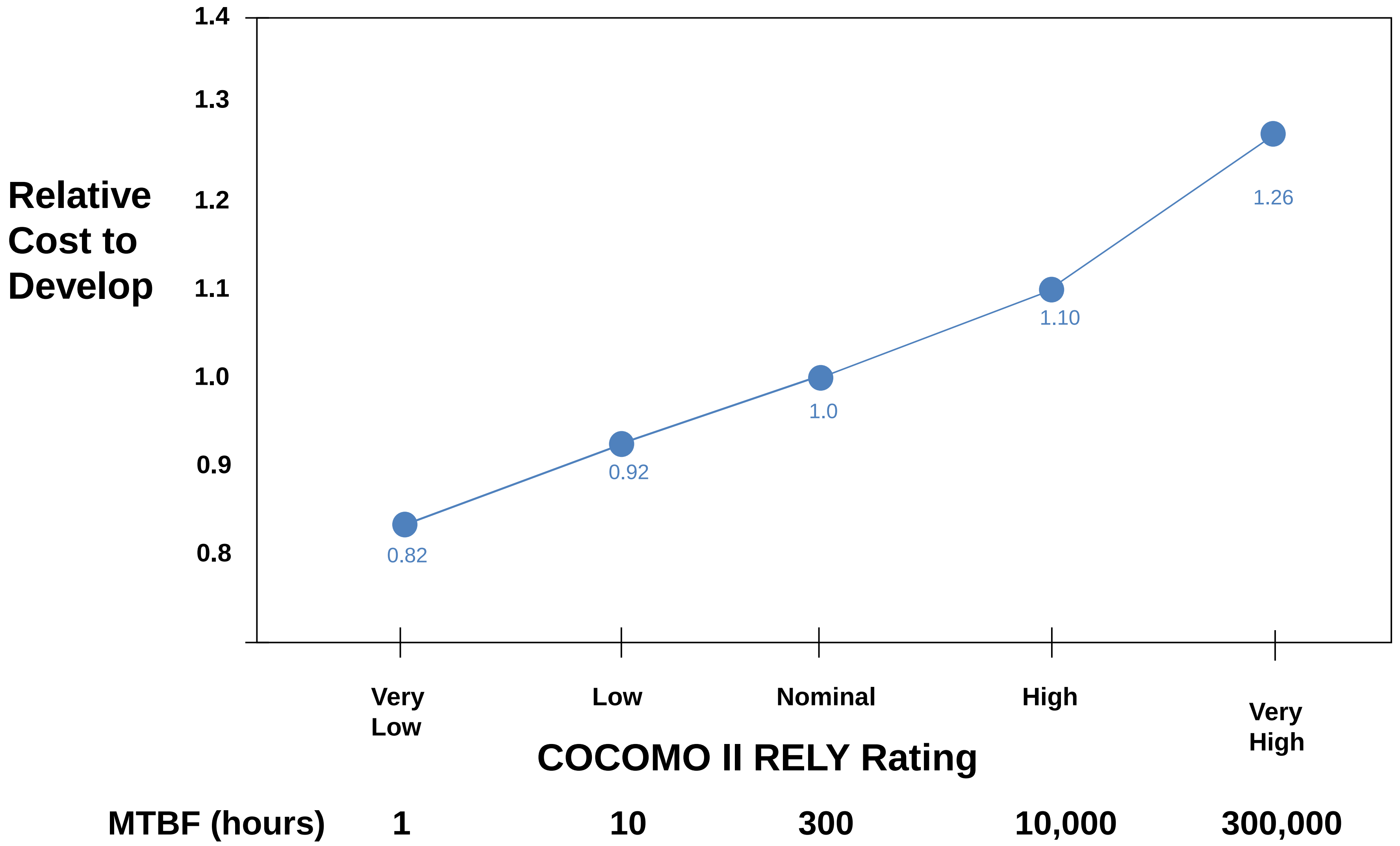
- **Mission operators and managers want improved Mission Effectiveness**
  - Involves Physical Capability, Cyber Capability, Human Usability, Speed, Accuracy, Impact, Mobility, Scalability, Versatility, Interoperability
- **Mission investors and system owners want Mission Cost-Effectiveness**
  - Involves 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

7x7 Synergies and Conflicts Matrix

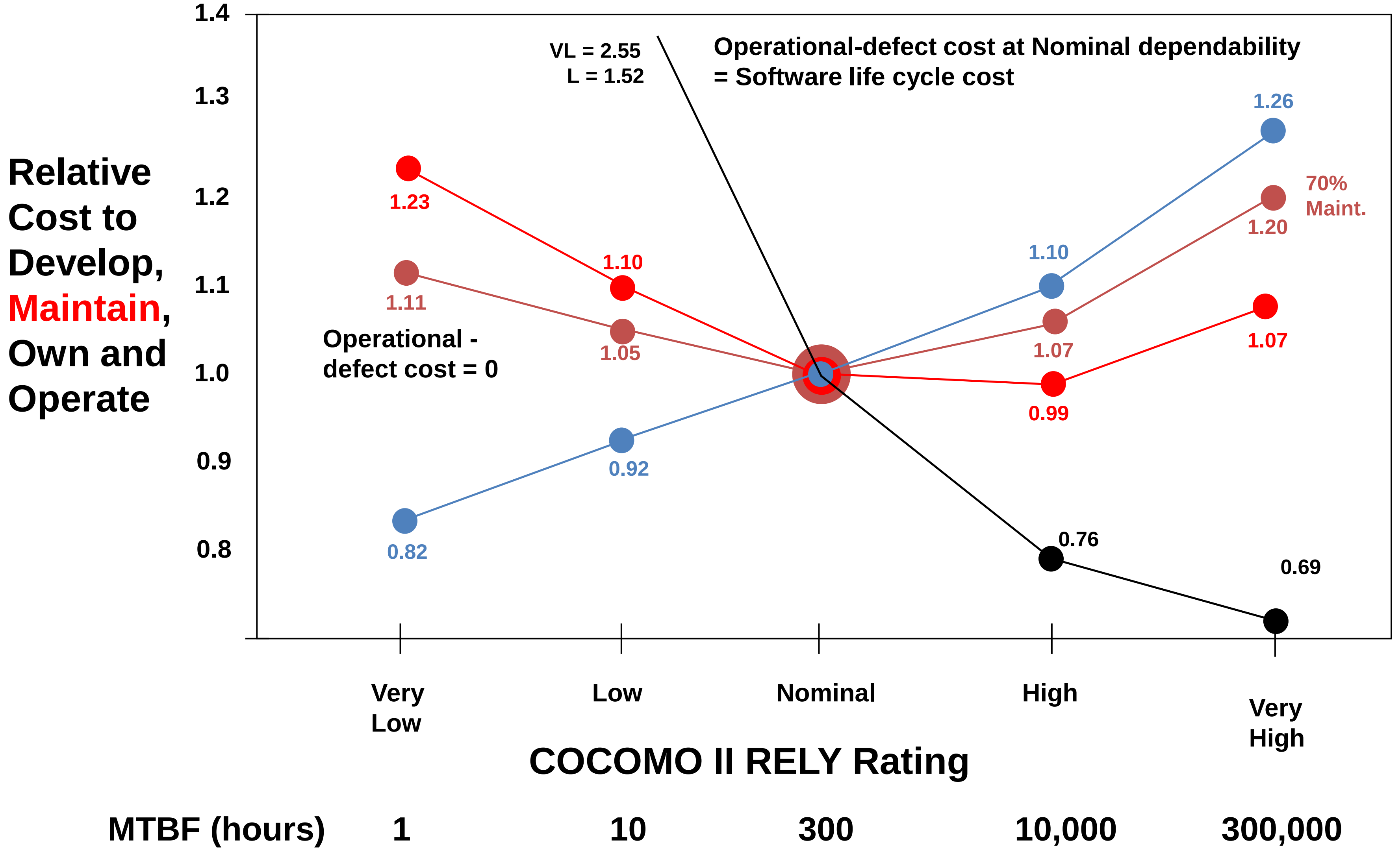
- Mission Effectiveness expanded to 4 elements
  - Physical Capability, Cyber Capability, Interoperability, Other Mission Effectiveness (including Usability as Human Capability)
- Synergies and Conflicts among the 7 resulting elements identified in 7x7 matrix
  - Synergies above main diagonal, Conflicts 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

	Flexibility	Dependability	Mission Effectivenss	Resource Utilization	Physical Capability	Cyber Capability	Interoperability
Flexibility		Domain architecting within domain Modularity Self Adaptive Smart monitoring Spare Capacity Use software vs. hardware	Adaptability Many options Service oriented Spare capacity User programmability Versatility	Adaptability Agile methods Automated I/O validation Loose coupling for sustainability Product line architectures Staffing, Empowering	Adaptability Spare capacity	Adaptability Spare capacity	Adaptability Loose coupling Modularity Product line architectures Service-oriented connectors Use software vs. Hardware User programmability
Dependability	Accreditation Agile methods assurance Encryption Many options Multi-domain modifiability Multi-level security Self Adaptive defects User programmability		Accreditation FMEA Multi-level security Survivability Spare capacity	Automated aids Automated I/O validation Domain architecting within domain Product line architectures Staffing, Empowering **Total Ownership Cost** Value prioritizing	Fallbacks Lightweight agility Redundancy Spare capacity Value prioritizing	Fallbacks Redundancy Value prioritizing	Assertion Checking Domain architecting within domain Service oriented
Mission Effectivenss	Autonomy vs. Usability Modularity slowdowns Multi-domain architecture interoperability conflicts Versatility vs. Usability	Anti-tamper Armor vs. Weight Easiest-first development Redundancy Scalability Spare Capacity Usability vs. Security		Automated aids Domain architecting within domain Staffing, Empowering Value prioritizing	Automated aids Domain architecting within domain Staffing, Empowering Value prioritizing	Automated aids Domain architecting within domain Staffing, Empowering Value prioritizing	Automated aids Domain architecting within domain Staffing, Empowering
Resource Utilization	Agile Methods scalability Assertion checking overhead Fixed cost contracts Modularity Multi-domain architecture interoperability conflicts Spare capacity Tight coupling Use software vs. hardware	Accreditation **Acquisition Cost** Certification Easiest-first development Fallbacks Multi-domain architecture interoperability conflicts Redundancy Spare Capacity, tools costs Usability vs. Cost savings	Agile methods scalability Cost of automated aids Many options Multi-domain architecture interoperability conflicts Spare capacity Usability vs. Cost savings Versatility		Automated aids Domain architecting within domain Staffing, Empowering Value prioritizing	Automated aids Domain architecting within domain Staffing, Empowering Value prioritizing	Automated aids Domain architecting within domain Rework cost savings Staffing, Empowering
Physical Capability	Multi-domain architecture interoperability conflicts Over-optimizing Tight coupling Use software vs. hardware	Lightweight agility Multi-domain architecture interoperability conflicts Over-optimizing	Multi-domain architecture interoperability conflicts Over-optimizing	Cost of automated aids Multi-domain architecture interoperability conflicts Over-optimizing		Automated aids Staffing, Empowering Value prioritizing	Automated aids Domain architecting within domain
Cyber Capability	Agile Methods scalability Multi-domain architecture interoperability conflicts Over-optimizing Tight coupling Use software vs. hardware	Multi-domain architecture interoperability conflicts Over-optimizing	Multi-domain architecture interoperability conflicts Over-optimizing	Cost of automated aids Multi-domain architecture interoperability conflicts Over-optimizing	Over-optimizing Physical architecture or cyber architecture		Automated aids Domain architecting within domain
Interoperability	Multi-domain architecture interoperability conflicts User-programmed interoperability	Encryption interoperability Multi-domain architecture interoperability conflicts	Multi-domain architecture interoperability conflicts	Assertion checking Cost, duration of added connectors	Over-optimizing Tight vs. Loose coupling	Reduced speed of Assertion checking Reduced speed of connectors, standards compliance Tight vs. Loose coupling	

Software Development Cost vs. Reliability



Software Ownership Cost vs. Reliability



	Security	Reliability	Maintainability
Security		Confidentiality, Integrity, Avalability Assurance Cases Certification Failure Modes and Effects Analysis Fault Tree Analysis Recertification	Certification Diagnosability Integrity, Avalability Repairability Smart Monitoring Spare Capacity
Reliability	Non-redundancy (For Security) Redundancy (For Reliability)		Accessibility Certification Diagnosability Repairability Smart Monitoring Spare Capacity
Maintainability	Accessibility Compartmentalization Encryption Recertification	Armor Recertification	

SYSTEMS ENGINEERING
Research Center

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

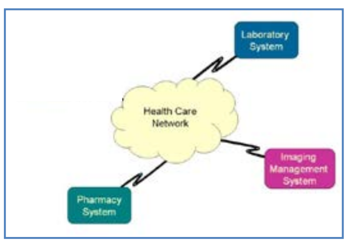
Ilities Tradespace and Affordability Analysis

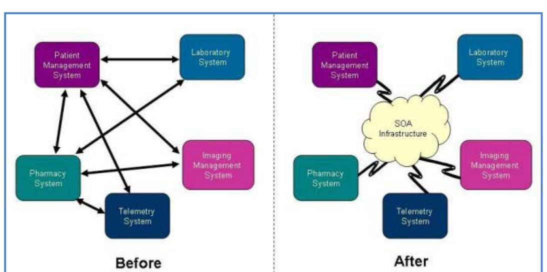
- Critical nature of the ilities
  - Or non-functional requirements; quality attributes
  - Major source of project overruns, failures
  - Significant source of stakeholder value conflicts
  - Poorly defined, understood
  - Underemphasized in project management
  - Need for ilities ontology
- Ility synergies and conflicts analysis
  - Stakeholder value-based, means-ends hierarchy
  - Synergies and Conflicts matrix and expansions
- ➡ SysML-COSYSMO Integration: GTRI (Peak), USC (Lane)

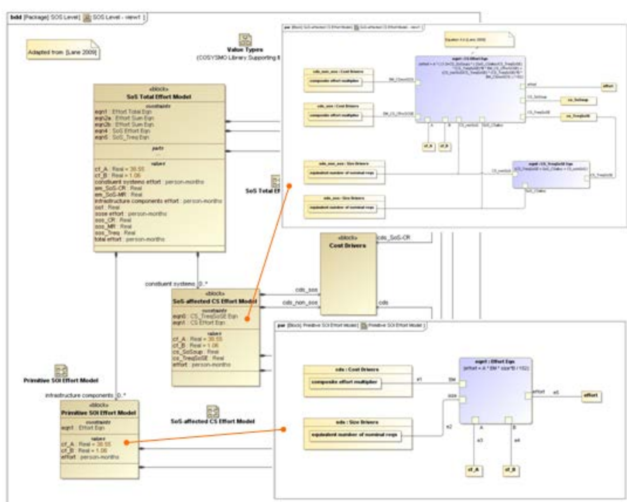
SysML Building Blocks for Cost Modeling

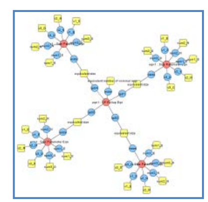
Summary (Oct 2013-Dec 2014) – R. Peak, GTRI; J. Lane, USC; R. Madachy, NPS

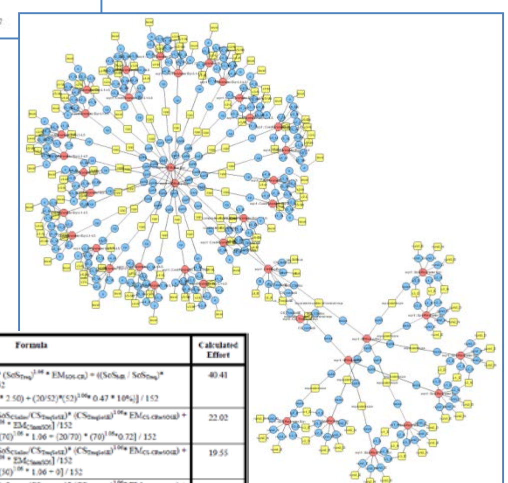
- Implemented cost modeling concepts as SysML building blocks
  - Concepts per SoS/COSYSMO work by Lane, Valerdi, Boehm, et al.
  - SysML knowledge capture that is reusable, modular, more complete
- Enables integration with complex SysML models
- Successfully applied building blocks to two healthcare SoS case studies
- Provides key step towards affordability trade studies involving diverse "-ilities"









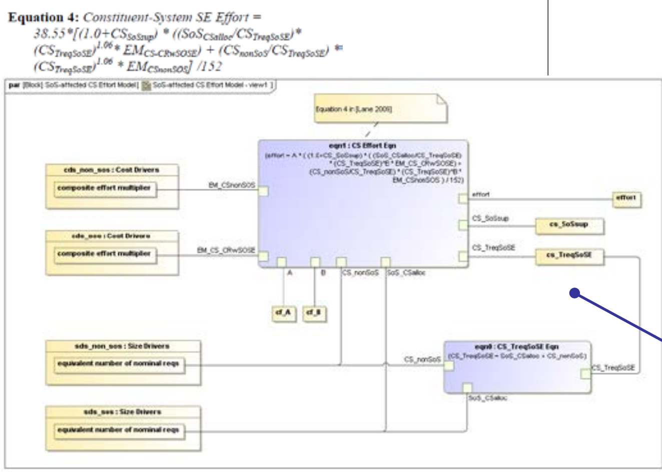


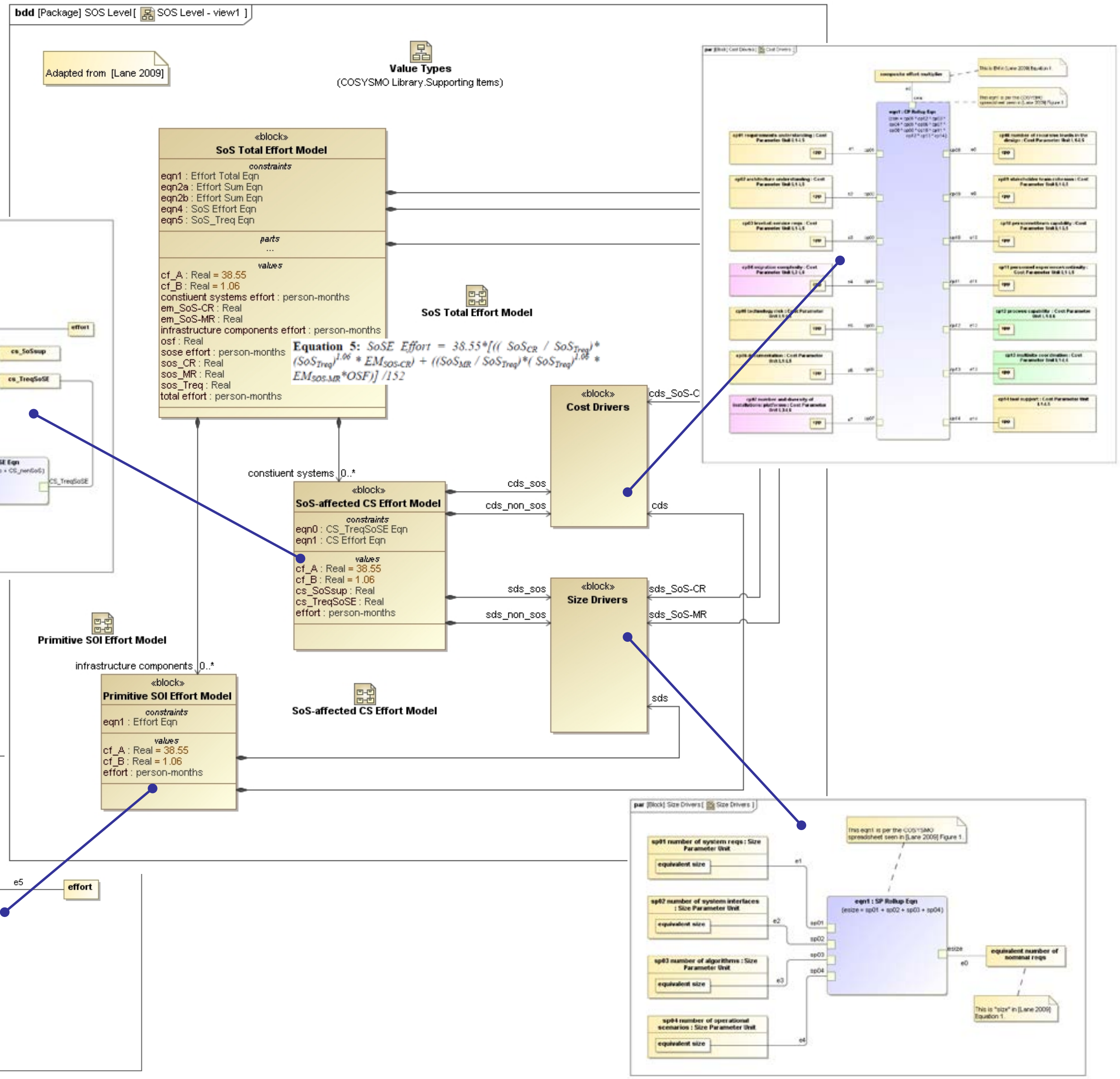
Aspect	Formula	Calculated Effort
SoSE effort (Equation 5)	Effort = 38.55*[((SoS_{CR} / SoS_{Treq})* $(SoS_{Treq})^{1.06}$ * EM_{SOS-CR}) + ((SoS_{MR} / SoS_{Treq})* $(SoS_{Treq})^{1.06}$ * EM_{SOS-MR}*OSF)] /152 = 38.55*[((50 / 52) * $(52)^{1.06}$ * 2.50) + (20/52)*$(52)^{1.06}$* 0.47 * 10%)] / 152	40.41
Pharmacy System effort (Equation 4)	Effort = 38.55*[(1.0+CS_{SoSreq}) * (($SoS_{CSalter}$/$CS_{TreqSoSE}$)* $(CS_{TreqSoSE})^{1.06}$* $EM_{CS-CRwSOSE}$) + (CS_{nonSoS}/$CS_{TreqSoSE}$) * $(CS_{TreqSoSE})^{1.06}$ * $EM_{CSnonSOS}$] /152 = 38.55 *[(1.15) * ((50/70)*$(70)^{1.06}$ * 1.06 + (20/70) * $(70)^{1.06}$*0.72] / 152	22.02
Laboratory System effort (Equation 4)	Effort = 38.55*[(1.0+CS_{SoSreq}) * (($SoS_{CSalter}$/$CS_{TreqSoSE}$)* $(CS_{TreqSoSE})^{1.06}$* $EM_{CS-CRwSOSE}$) + (CS_{nonSoS}/$CS_{TreqSoSE}$) * $(CS_{TreqSoSE})^{1.06}$ * $EM_{CSnonSOS}$] /152 = 38.55 *[(1.15) * ((30/30)*$(30)^{1.06}$ * 1.06 + 0] / 152	19.55
Imaging System effort (Equation 4)	Effort = 38.55*[(1.0+CS_{SoSreq}) * (($SoS_{CSalter}$/$CS_{TreqSoSE}$)* $(CS_{TreqSoSE})^{1.06}$* $EM_{CS-CRwSOSE}$) + (CS_{nonSoS}/$CS_{TreqSoSE}$) * $(CS_{TreqSoSE})^{1.06}$ * $EM_{CSnonSOS}$] /152 = 38.55 *[(1.15) * ((30/30)*$(30)^{1.06}$ * 1.06 + 0] / 152	19.55
New infrastructure component effort (Equation 1)	Effort = 38.55*EM*$(size)^{1.05}$/152 = 38.55 * 1.0 * $(100)^{1.05}$ / 152	33.43
	Total Effort:	114.96

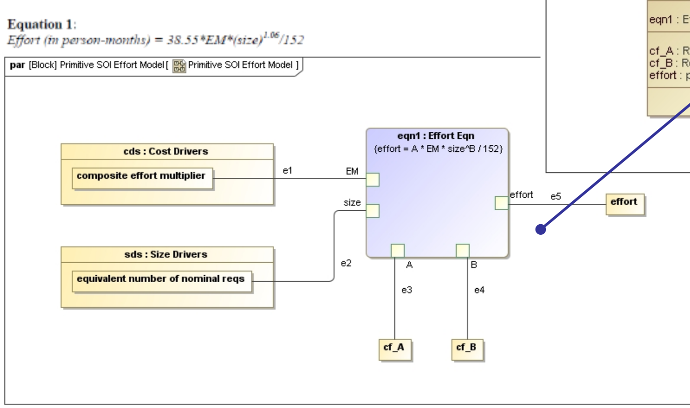
COSYSMO/SoS Concepts Implemented as SysML Building Blocks: *Selected SysML Diagrams*

SYSTEMS ENGINEERING Research Center

Generic and useful to analyze practically any SoS (see case study applications in next slides)



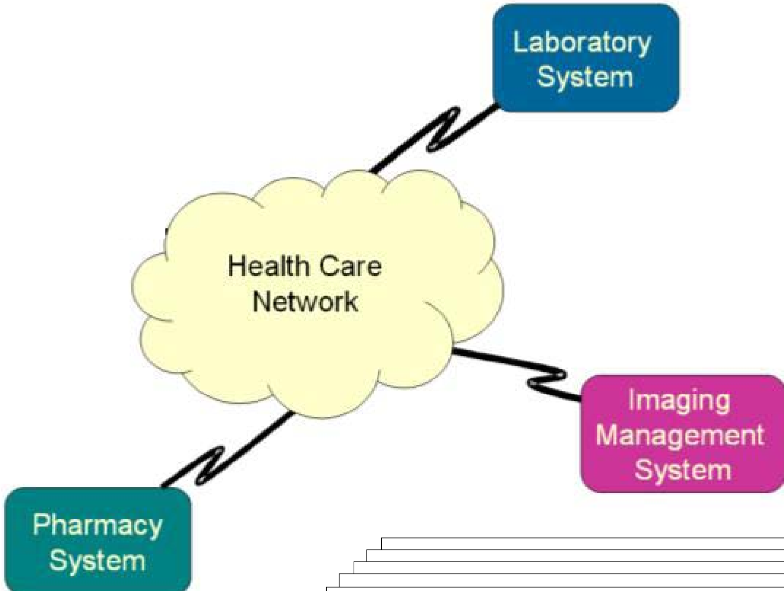




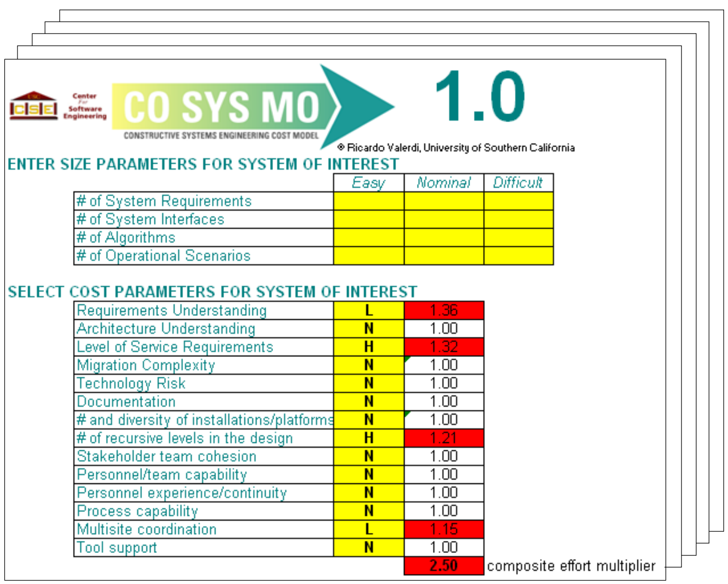
Healthcare SoS Case Study1 [Lane 2009]

Original Document & Spreadsheet Views

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



Aspect	Formula	Calculated Effort
SoSE effort (Equation 5)	$Effort = 38.55*[((SoS_{CR} / SoS_{Treq})*(SoS_{Treq})^{1.06} * EM_{SOS\text{-}CR}) + ((SoS_{MR} / SoS_{Treq})*(SoS_{Treq})^{1.06} * EM_{SOS\text{-}MR}*OSF)] /152$ $= 38.55*[((50 / 52) * (52)^{1.06} * 2.50) + (20/52)*(52)^{1.06} * 0.47 * 10\%)] / 152$	40.41
Pharmacy System effort (Equation 4)	$Effort = 38.55*[(1.0+CS_{SoSsup}) * ((SoS_{CSalloc}/CS_{TreqSoSE})*(CS_{TreqSoSE})^{1.06} * EM_{CS\text{-}CRwSOSE}) + (CS_{nonSoS}/CS_{TreqSoSE}) * (CS_{TreqSoSE})^{1.06} * EM_{CSnonSOS}] /152$ $= 38.55 *[(1.15) * ((50/70)*(70)^{1.06} * 1.06 + (20/70) * (70)^{1.06}*0.72] / 152$	22.02
Laboratory System effort (Equation 4)	$Effort = 38.55*[(1.0+CS_{SoSsup}) * ((SoS_{CSalloc}/CS_{TreqSoSE})*(CS_{TreqSoSE})^{1.06} * EM_{CS\text{-}CRwSOSE}) + (CS_{nonSoS}/CS_{TreqSoSE}) * (CS_{TreqSoSE})^{1.06} * EM_{CSnonSOS}] /152$ $= 38.55 *[(1.15) * ((50/50)*(50)^{1.06} * 1.06 + 0] / 152$	19.55
Imaging System effort (Equation 4)	$Effort = 38.55*[(1.0+CS_{SoSsup}) * ((SoS_{CSalloc}/CS_{TreqSoSE})*(CS_{TreqSoSE})^{1.06} * EM_{CS\text{-}CRwSOSE}) + (CS_{nonSoS}/CS_{TreqSoSE}) * (CS_{TreqSoSE})^{1.06} * EM_{CSnonSOS}] /152$ $= 38.55 *[(1.15) * ((50/50)*(50)^{1.06} * 1.06 + 0] / 152$	19.55
New infrastructure component effort (Equation 1)	$Effort = 38.55*EM*(size)^{1.06}/152$ $= 38.55 * 1.0 * (100)^{1.06} / 152$	33.43
	Total Effort:	134.96



ENTER SIZE PARAMETERS FOR SYSTEM OF INTEREST

	Easy	Nominal	Difficult
# of System Requirements			
# of System Interfaces			
# of Algorithms			
# of Operational Scenarios			

SELECT COST PARAMETERS FOR SYSTEM OF INTEREST

Requirements Understanding	L	1.36
Architecture Understanding	N	1.00
Level of Service Requirements	H	1.32
Migration Complexity	N	1.00
Technology Risk	N	1.00
Documentation	N	1.00
# and diversity of installations/platforms	N	1.00
# of recursive levels in the design	H	1.21
Stakeholder team cohesion	N	1.00
Personnel/team capability	N	1.00
Personnel experience/continuity	N	1.00
Process capability	N	1.00
Multisite coordination	L	1.15
Tool support	N	1.00
		2.50 composite effort multiplier

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

Healthcare SoS Case Study1: Model Subset

Spreadsheet View vs. SysML DNA Signature View

Subset of full model

COSYSMO 1.0 — CONSTRUCTIVE SYSTEMS ENGINEERING COST MODEL

© Ricardo Valerdi, University of Southern California

ENTER SIZE PARAMETERS FOR SYSTEM OF INTEREST

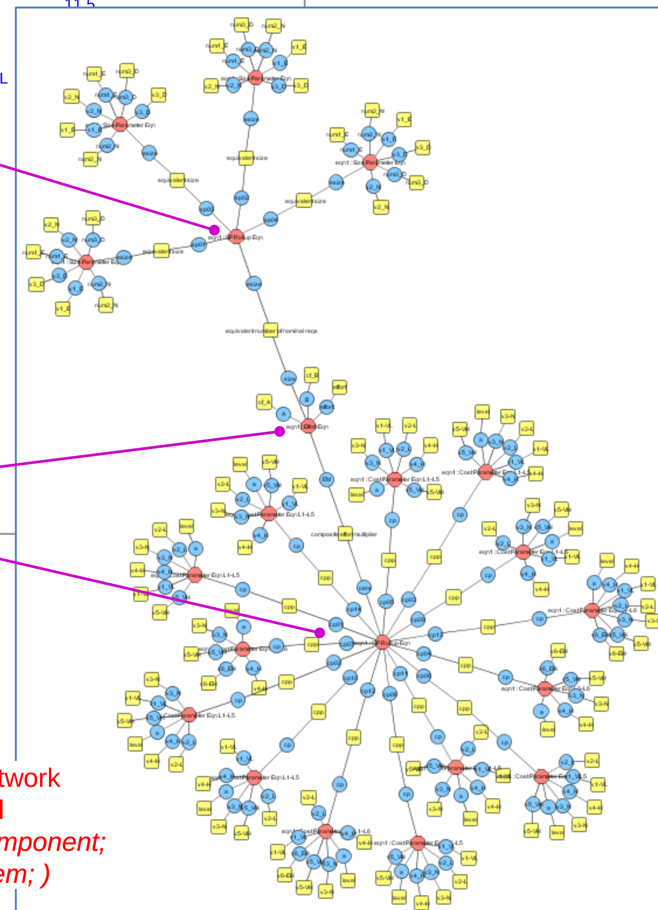
	Easy	Nominal	Difficult			E	N	D
# of System Requirements	10	20	7	60	equivalent size	0.5	1.0	5.0
# of System Interfaces	10	10	8	89		1.1	2.8	6.3
# of Algorithms	20	15	10	221		2.2	4.1	11.5
# of Operational Scenarios	5	5	4	223		6.2	14.4	
				593				

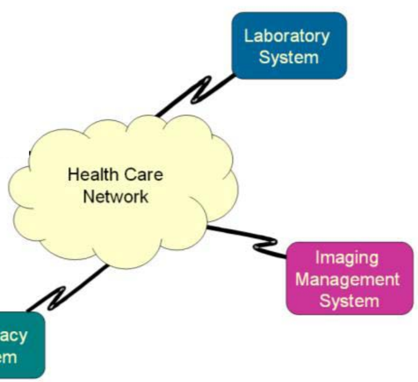
SELECT COST PARAMETERS FOR SYSTEM OF INTEREST

			VL	VL-L	L
Requirements Understanding	H	0.77460	1.85		1.59
Architecture Understanding	H	0.80623	1.62		1.44
Level of Service Requirements	H	1.31909	0.62		0.70
Migration Complexity	N	1.00000			
Technology Risk	N	1.00000	0.70		0.77
Documentation	H	1.13137	0.82		0.86
# and diversity of installations/platforms	N	1.00000			
# of recursive levels in the design	N	1.00000	0.80		0.85
Stakeholder team cohesion	VL	1.50000	1.50		1.36
Personnel/team capability	N	1.00000	1.48		1.34
Personnel experience/continuity	N	1.00000	1.46		1.33
Process capability	EH	0.68000	1.46		1.33
Multisite coordination	L	1.15326	1.33		1.24
Tool support	N	1.00000	1.34		1.25
		1.09632 composite effort multiplier			

pro forma parameter values

SYSTEMS ENGINEERING PERSON MONTHS	**241.8**	38.55	1.06



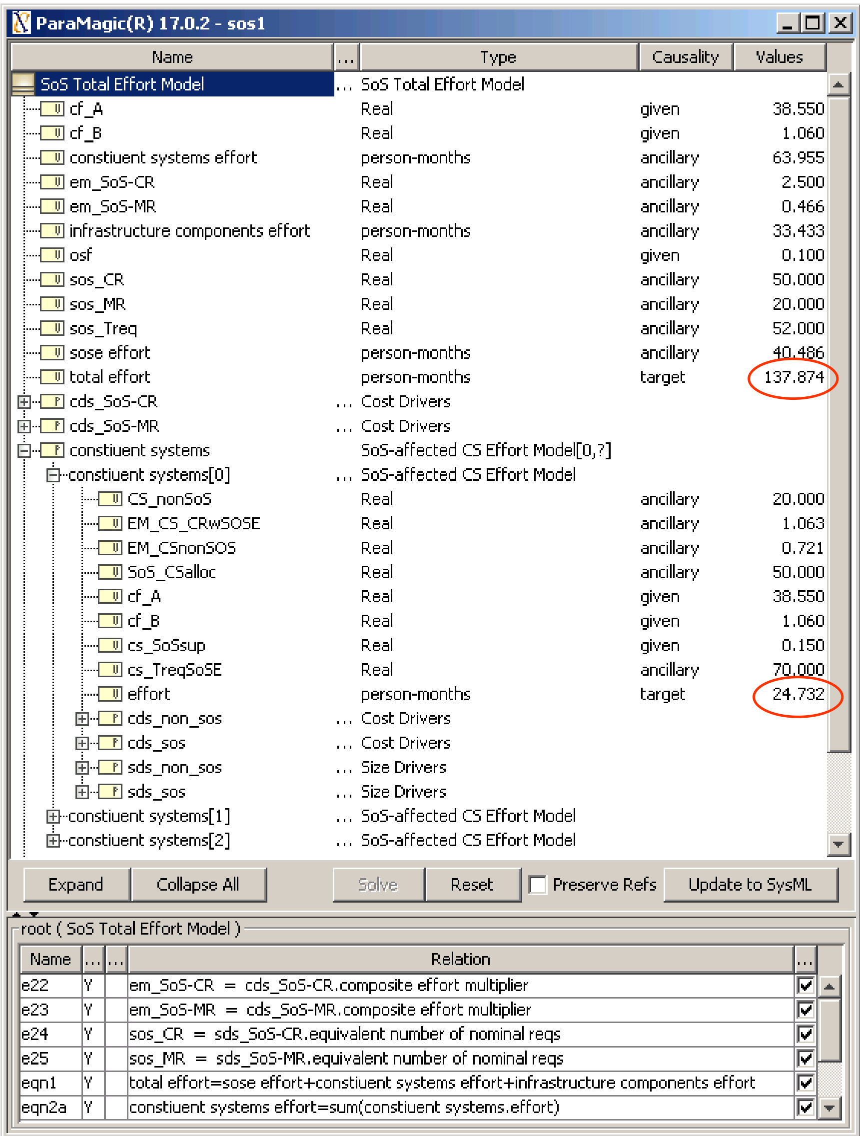


Healthcare IT Network
Effort Model
(an infrastructure component;
a primitive system;)

Healthcare SoS Case Study1: Model Execution

Tool for Solving SysML Instance Structures

(object-oriented spreadsheet-like tool)



ParaMagic(R) 17.0.2 - sos1

Name	...	Type	Causality	Values
SoS Total Effort Model	...	SoS Total Effort Model		
cf_A		Real	given	38.550
cf_B		Real	given	1.060
constiuent systems effort		person-months	ancillary	63.955
em_SoS-CR		Real	ancillary	2.500
em_SoS-MR		Real	ancillary	0.466
infrastructure components effort		person-months	ancillary	33.433
osf		Real	given	0.100
sos_CR		Real	ancillary	50.000
sos_MR		Real	ancillary	20.000
sos_Treq		Real	ancillary	52.000
sose effort		person-months	ancillary	40.486
total effort		person-months	target	137.874
cds_SoS-CR	...	Cost Drivers		
cds_SoS-MR	...	Cost Drivers		
constiuent systems		SoS-affected CS Effort Model[0,?]		
constiuent systems[0]	...	SoS-affected CS Effort Model		
CS_nonSoS		Real	ancillary	20.000
EM_CS_CRwSOSE		Real	ancillary	1.063
EM_CSnonSOS		Real	ancillary	0.721
SoS_CSalloc		Real	ancillary	50.000
cf_A		Real	given	38.550
cf_B		Real	given	1.060
cs_SoSsup		Real	given	0.150
cs_TreqSoSE		Real	ancillary	70.000
effort		person-months	target	24.732
cds_non_sos	...	Cost Drivers		
cds_sos	...	Cost Drivers		
sds_non_sos	...	Size Drivers		
sds_sos	...	Size Drivers		
constiuent systems[1]	...	SoS-affected CS Effort Model		
constiuent systems[2]	...	SoS-affected CS Effort Model		

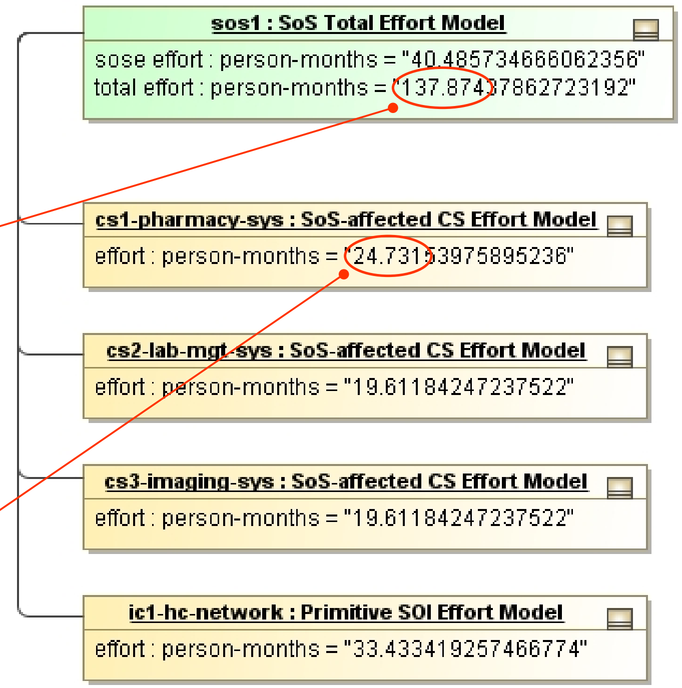
Expand | Collapse All | Solve | Reset | Preserve Refs | Update to SysML

root (SoS Total Effort Model)

Name	...	...	Relation	...
e22	Y		em_SoS-CR = cds_SoS-CR.composite effort multiplier	☑
e23	Y		em_SoS-MR = cds_SoS-MR.composite effort multiplier	☑
e24	Y		sos_CR = sds_SoS-CR.equivalent number of nominal reqs	☑
e25	Y		sos_MR = sds_SoS-MR.equivalent number of nominal reqs	☑
eqn1	Y		total effort=sose effort+constiuent systems effort+infrastructure components effort	☑
eqn2a	Y		constiuent systems effort=sum(constiuent systems.effort)	☑

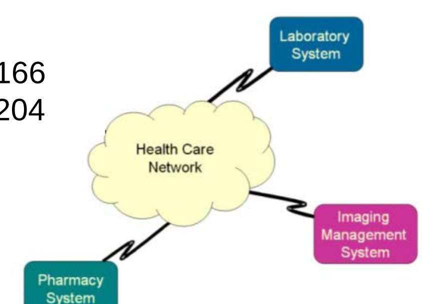
Top-Level SysML Instances

(bdd view - after solving in ParaMagic)



No. of variables: 1166

No. of equations: 204



Healthcare SoS Case Study1: Implementation Results

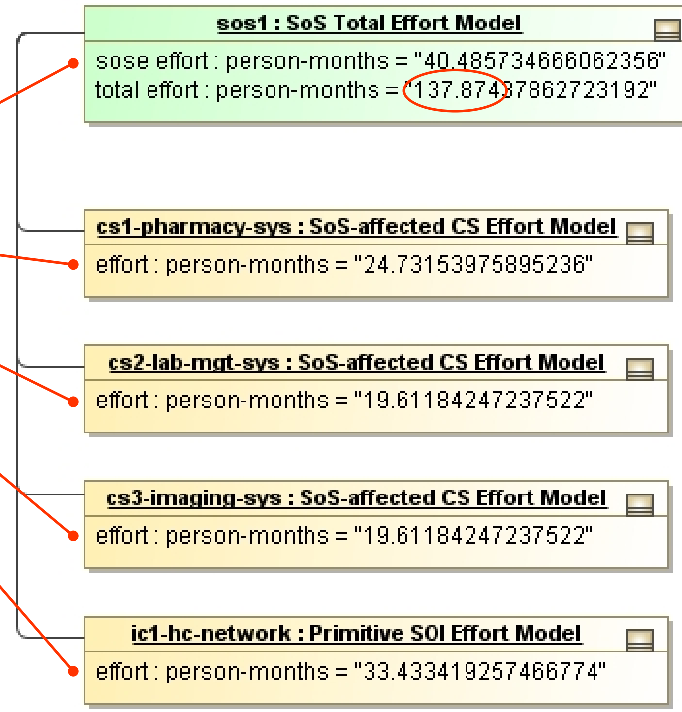
Good verification compared to original results

Original Results Summary [Lane 2009]

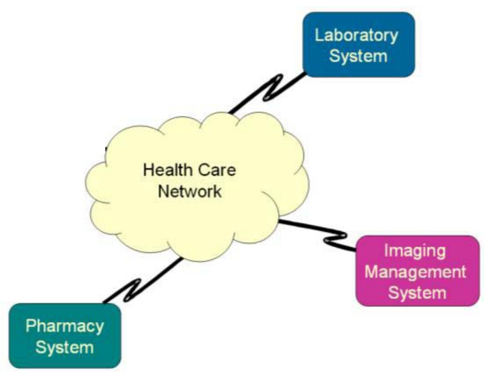
(subject to known corrections & round-off)

Aspect	Formula	Calculated Effort
SoSE effort (Equation 5)	Effort = $38.55*[((SoS_{CR} / SoS_{Treq})*(SoS_{Treq})^{1.06} * EM_{SOS\text{-}CR}) + ((SoS_{MR} / SoS_{Treq})*(SoS_{Treq})^{1.06} * EM_{SOS\text{-}MR}*OSF)] /152$ $= 38.55*[((50 / 52) * (52)^{1.06} * 2.50) + (20/52)*(52)^{1.06} * 0.47 * 10\%)] / 152$	40.41
Pharmacy System effort (Equation 4)	Effort = $38.55*[(1.0+CS_{SoSsup}) * ((SoS_{CSalloc}/CS_{TreqSoSE})*(CS_{TreqSoSE})^{1.06} * EM_{CS\text{-}CRwSOSE}) + (CS_{nonSoS}/CS_{TreqSoSE}) * (CS_{TreqSoSE})^{1.06} * EM_{CSnonSOS}] /152$ $= 38.55 *[(1.15) * ((50/70)*(70)^{1.06} * 1.06 + (20/70) * (70)^{1.06}*0.72] / 152$	24.65
Laboratory System effort (Equation 4)	Effort = $38.55*[(1.0+CS_{SoSsup}) * ((SoS_{CSalloc}/CS_{TreqSoSE})*(CS_{TreqSoSE})^{1.06} * EM_{CS\text{-}CRwSOSE}) + (CS_{nonSoS}/CS_{TreqSoSE}) * (CS_{TreqSoSE})^{1.06} * EM_{CSnonSOS}] /152$ $= 38.55 *[(1.15) * ((50/50)*(50)^{1.06} * 1.06 + 0] / 152$	19.55
Imaging System effort (Equation 4)	Effort = $38.55*[(1.0+CS_{SoSsup}) * ((SoS_{CSalloc}/CS_{TreqSoSE})*(CS_{TreqSoSE})^{1.06} * EM_{CS\text{-}CRwSOSE}) + (CS_{nonSoS}/CS_{TreqSoSE}) * (CS_{TreqSoSE})^{1.06} * EM_{CSnonSOS}] /152$ $= 38.55 *[(1.15) * ((50/50)*(50)^{1.06} * 1.06 + 0] / 152$	19.55
New infrastructure component effort (Equation 1)	Effort = $38.55*EM*(size)^{1.06}/152$ $= 38.55 * 1.0 * (100)^{1.06} / 152$	33.43
	Total Effort:	137.59

SysML-Based Results Summary

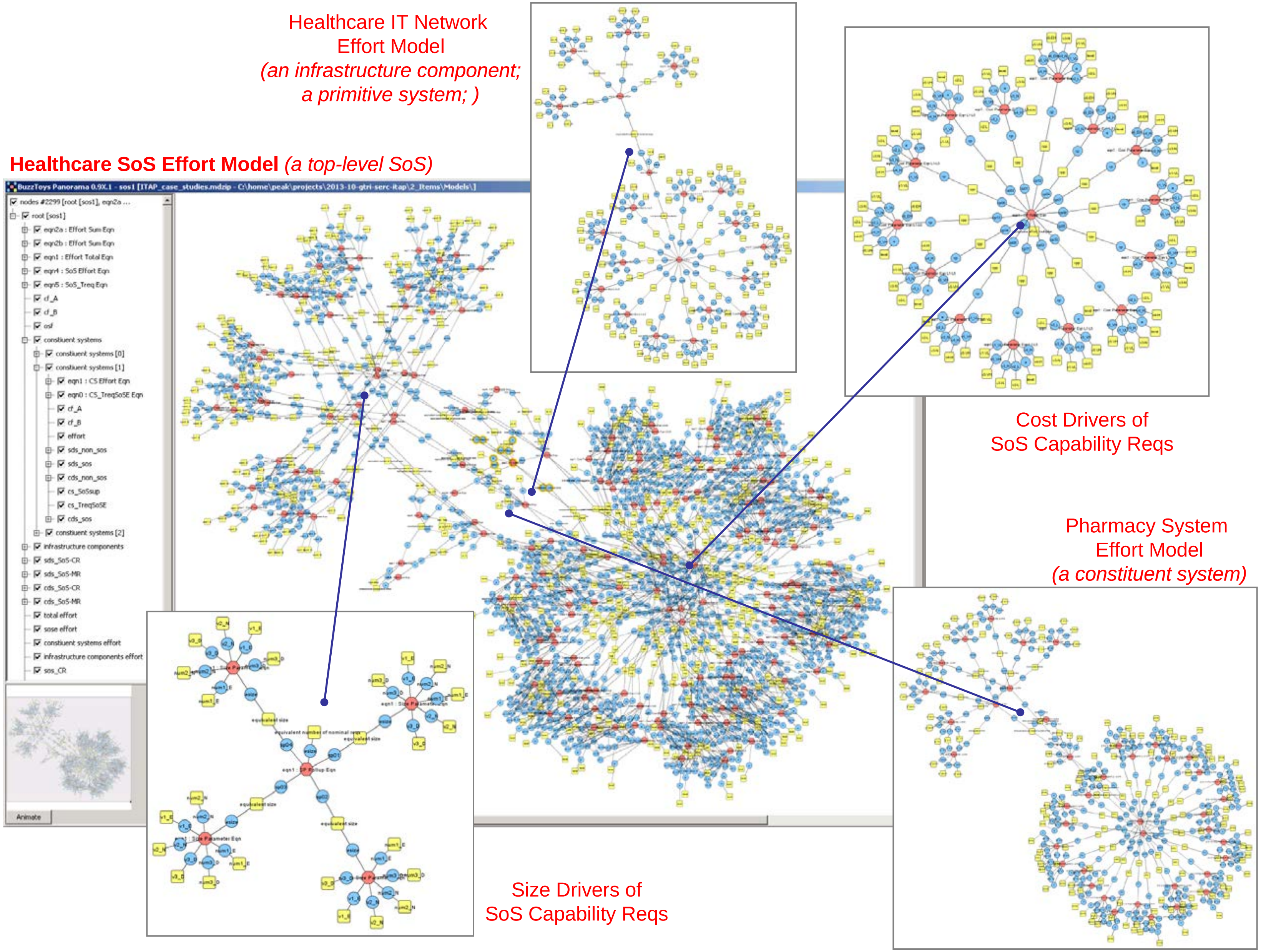


No. of variables: 1166
No. of equations: 204



SYSTEMS ENGINEERING Research Center

Healthcare SoS Case Study1: Full Model
SysML DNA Signature View



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)

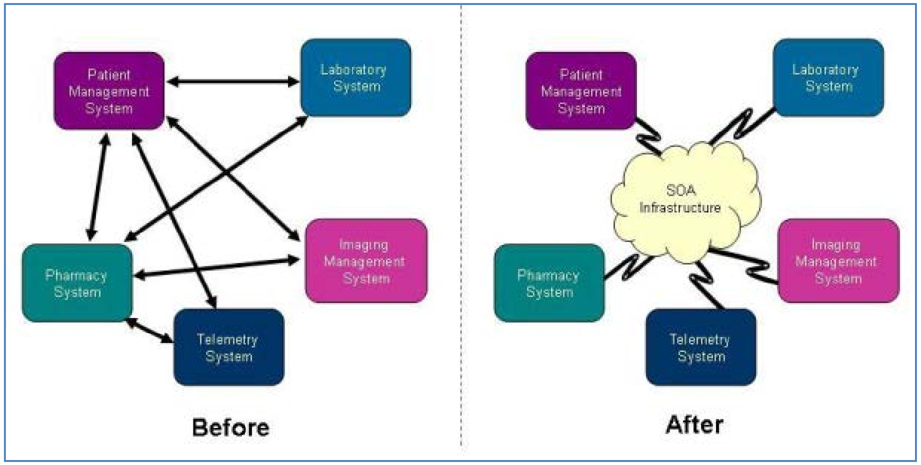


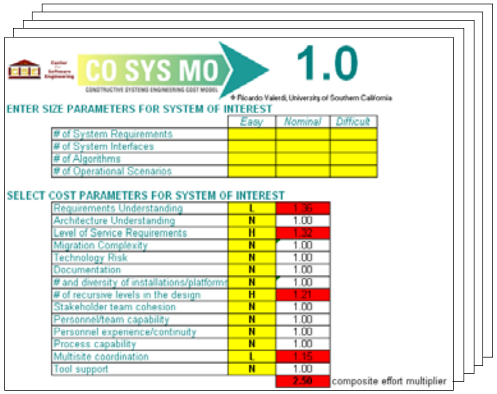
Table 6-2. Health Care SoS SOA Capability Size Overview

SoS Product (CS)	SoS SOA Requirements ($SoS_{CSalloc}$)	Concurrent Product Upgrade Size (CS_{nonSoS})	Total Requirements for CS ($CS_{TreqSoSE}$)
SOA Infrastructure	• Infrastructure characteristics: 20 • Shared data standards: 10	None (new constituent system)	30
Patient Management System	• Infrastructure characteristics: 10 • Shared data standards: 100	30 requirements	140
Pharmacy System	• Infrastructure characteristics: 10 • Shared data standards: 75	60 requirements	145
Telemetry System	• Infrastructure characteristics: 10 • Shared data standards: 40	25 requirements	75
Imaging Management System	• Infrastructure characteristics: 10 • Shared data standards: 35	50 requirements	95
Laboratory System	• Infrastructure characteristics: 10 • Shared data standards: 45	10 requirements	65

Table 6-3. SoSE Effort Equation Parameters[1]

Parameter	SoSE	SOA Infrastructure (New SoS Element)	Patient Management System	Pharmacy System	Telemetry System	Imaging Management System	Laboratory System
CS_{nonSoS}		0	30	60	25	50	10
$CS_{TreqSoSE}$		30	140	145	75	95	65
EM_{SOS-CR}	2.50						
EM_{SOS-MR}	0.47						
$EM_{CS-CRwSOSE}$		1.62	1.06	1.06	1.06	1.06	1.06
$EM_{CSnonSOS}$		n/a	0.72	0.67	0.83	0.72	1.02
OSF (minimal oversight of product vendors)	5%						
SoS_{CR}	130						
$SoS_{CSalloc}$		30	110	85	50	45	55
SoS_{MR} (sum of $CS_{non\text{-}SoS}$* OSF)	8.75						
CS_{SoSsup} (can range from 0-30%)		30%	10%	10%	10%	10%	10%
SoS_{Treq}	138.75						
A_i *(Default Value: 38.55)*	38.55	38.55	39.47	39.25	36.82	38.55	38.95
B_i *(Default Value: 1.06)*	1.06	1.06	1.06	1.06	1.06	1.06	1.06

Recursively uses COSYSMO, and adds SoS aspects.



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

Equation 6-6: *SoSE Effort*

$= 38.55*[((130 / 138.75)* (138.75)^{1.06} * 2.50) + ((8.75 / 138.75)*(138.7$

$= 112$ *person months*

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

$= 38.55*[(1.0+0.30) * ((30/30)* (30)^{1.06}* 1.62)] /152$

$= 19.65$ *person months*

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

$= 39.47*[(1.0+0.10) * ((110/140)* (140)^{1.06}* 1.06) + (30/140) * (140)^{1.06}$

$= 52.44$ *person months*

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

$= 39.25*[(1.0+0.10_p) * ((85/145)* (145)^{1.06}* 1.06) + (60/145) * (145)^{1.06}$

$= 46.62$ *person months*

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

$= 36.82*[(1.0+0.10) * ((50/75)* (75)^{1.06}* 1.06) + (25/75) * (75)^{1.06} * 0.83$

$= 24.57$ *person months*

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

$= 38.55*[(1.0+0.10) * ((45/95)* (95)^{1.06}* 1.06) + (50/95) * (95)^{1.06} * 0.72] /152$

$= 29.74$ *person months*

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

$= 38.95*[(1.0+0.10) * ((55/65)* (65)^{1.06}* 1.06) + (10/65) * (65)^{1.06} * 1.02] /152$

$= 24.48$ *person months*

Healthcare SoS Case Study2: 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*

$= 38.55*[((130 / 138.75)* (138.75)^{1.06} * 2.50) + ((8.75 / 138.75)*(138.75)^{1.06} * 0.47)] /152$

= *112 person months*

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

$= 38.55*[(1.0+0.30) * ((30_c/30)* (30)^{1.06}* 1.62)] /152$

= *19.65 person months*

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

$= 39.47*[(1.0+0.10) * ((110/140)* (140)^{1.06}* 1.06) + (30/140) * (140)^{1.06} * 0.72] /152$

= *52.44 person months*

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

$= 39.25*[(1.0+0.10_p) * ((85/145)* (145)^{1.06}* 1.06) + (60/145) * (145)^{1.06} * 0.67] /152$

= *46.62 person months*

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

$= 36.82*[(1.0+0.10) * ((50/75)* (75)^{1.06}* 1.06) + (25/75) * (75)^{1.06} * 0.83] /152$

= *24.57 person months*

Equation 6-11: *Imaging Mana*

$= 38.55*[(1.0+0.10) * ((45/95)$

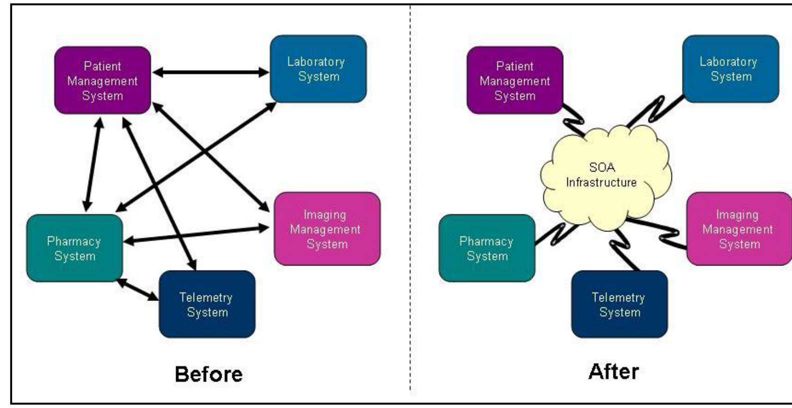
= *29.74 person months*

Equation 6-12: *Laboratory C*

$= 38.95*[(1.0+0.10) * ((55/65)$

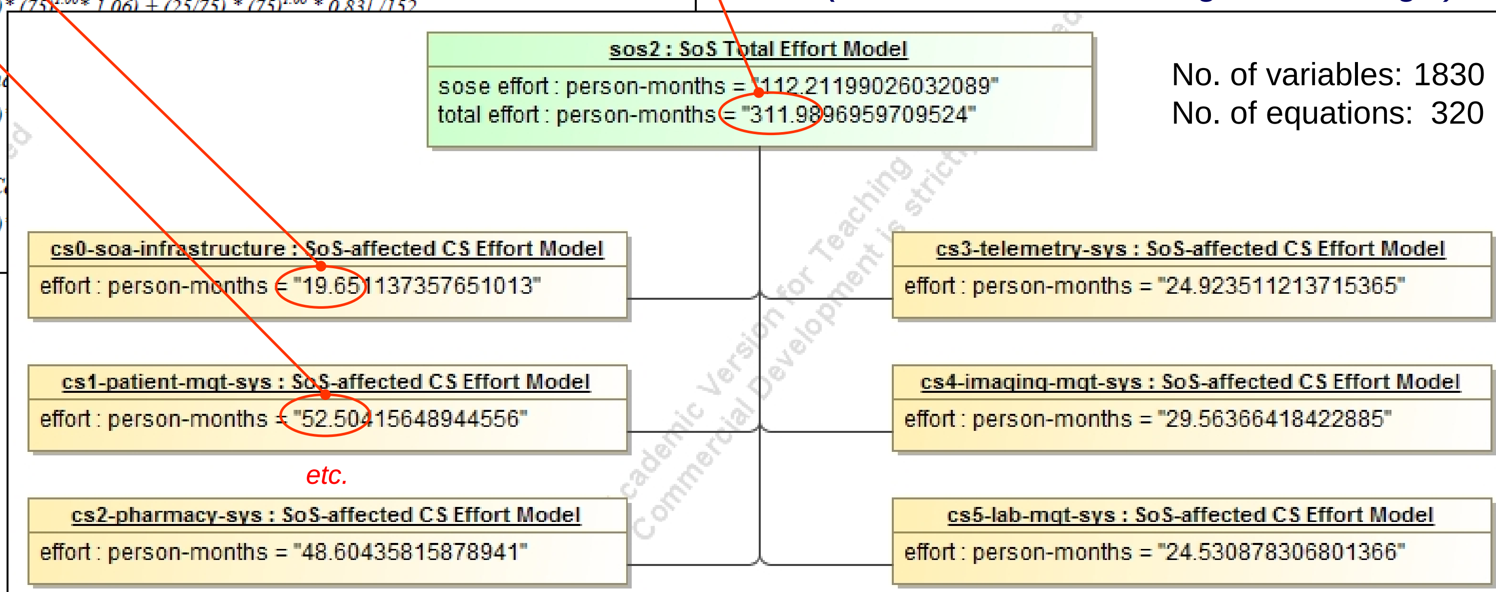
= *24.48 person months*

Original Schematic



Top-Level SysML Instances

(bdd view - after solving in ParaMagic)



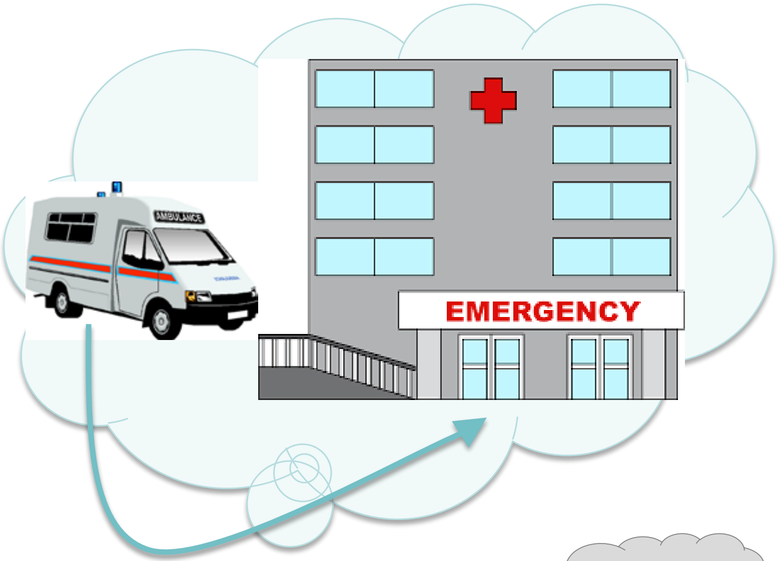
Good comparison (subject to errors in round-off)

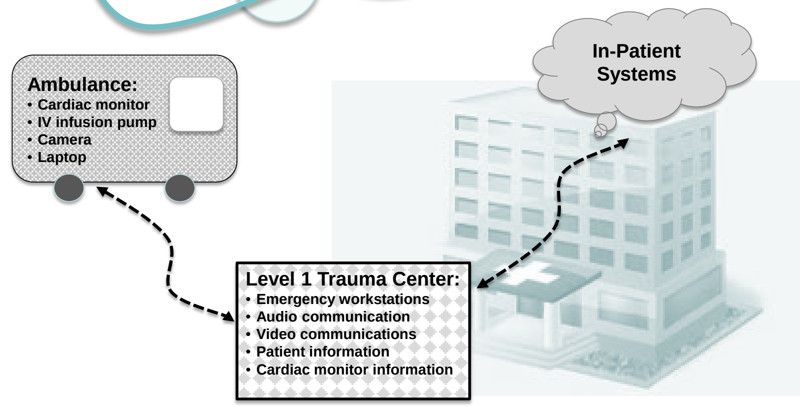
SYSTEMS ENGINEERING Research Center

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

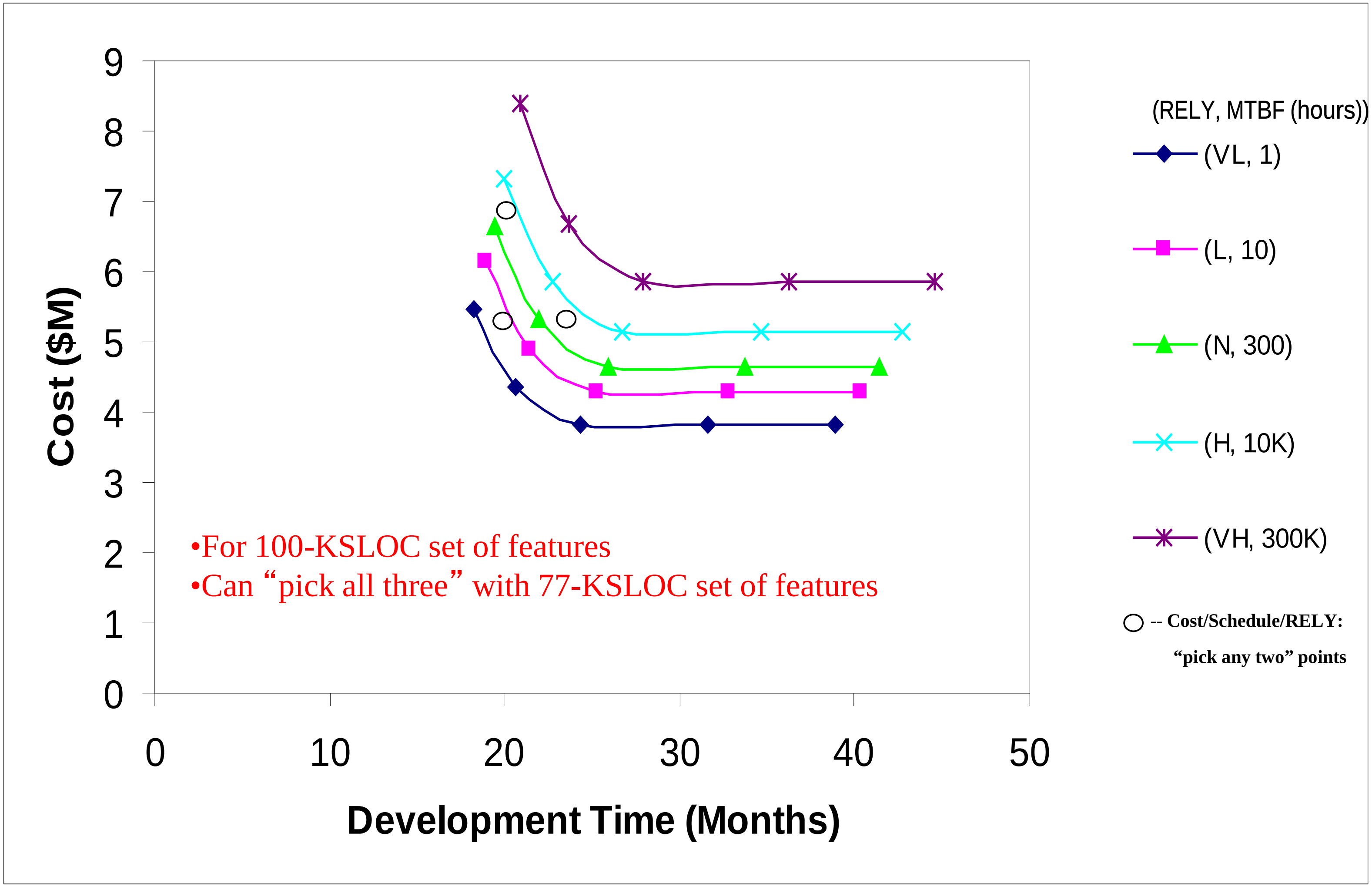
Summary & Impact

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?

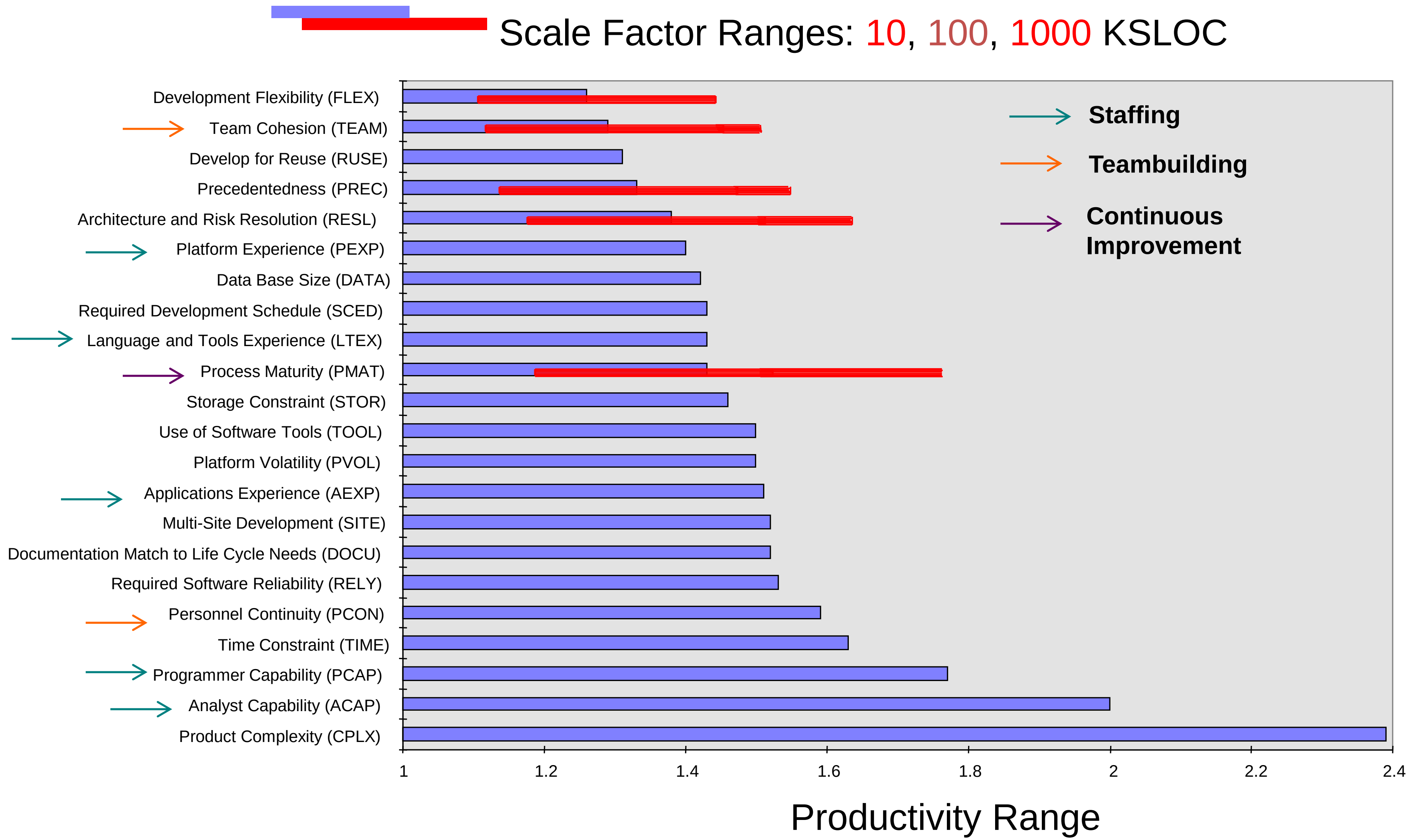


Affordability and Tradespace Framework

Cost Improvements and Tradeoffs

- **Get the Best from People**
  - Staffing, Incentivizing, Teambuilding
  - Facilities, Support Services
  - Kaizen (continuous improvement)
- **Make Tasks More Efficient**
  - Tools and Automation
  - Work and Oversight Streamlining
  - Collaboration Technology
- **Eliminate Tasks**
  - Lean and Agile Methods
  - Task Automation
  - Model-Based Product Generation
- **Eliminate Scrap, Rework**
  - Early Risk and Defect Elimination
  - Evidence-Based Decision Gates
  - Modularity Around Sources of Change
  - Incremental, Evolutionary Development
  - Value-Based, Agile Process Maturity
  - Risk-Based Prototyping
- **Simplify Products (KISS)**
  - Value-Based Capability Prioritization
  - Satisficing vs. Optimizing Performance
- **Reuse Components**
  - Domain Engineering and Architecture
  - Composable Components,Services, COTS
  - Legacy System Repurposing
- **Reduce Operations, Support Costs**
  - Automate Operations Elements
  - Design for Maintainability, Evolvability
  - Streamline Supply Chain
  - Anticipate, Prepare for Change
- **Value- and Architecture-Based Tradeoffs and Balancing**

Costing Insights: COCOMO II Productivity Ranges



COSYSMO Sys Engr Cost Drivers