The Role of Complexities in Systems Engineering Cost Estimating Processes

Leone Z. Young, Stevens Institute of Technology
Dr. John V. Farr, Stevens Institute of Technology
Dr. Ricardo Valerdi, Massachusetts Institute of Technology

Agenda

1. **Purpose & Motivation**
2. **Objectives**
3. **Complexity**
   - Overview
   - Types
4. **The Proposed Approach**
   - The Constructive Systems Engineering Cost Model (COSYSMO) - Sizing Categories
   - Mapping and Measuring Complexities with Systems Artifacts
   - The Systems Modeling Language (SysML)
5. **Summary & Conclusion**
6. **Future Work**

Purpose & Motivation

- Systems Designs, Systems Complexities & Systems Cost Estimates
  - **Systems Engineers ≠ Systems Cost Estimators**
  - Systems Complexities ***Impact*** Estimate Ability on…
    - Systems Engineering
    - Project & Program Management
    - Hardware, Software, Integration
- Cost Estimating Community & Literatures
  - Lack of procedural guidance of systems complexities
  - Void in Literature

Objectives

- **Mappings and Measuring Systems Complexities with Artifacts**
  - *Requirements, Interfaces, Algorithms and Operation Scenarios (COSYSMO) as Mapping Categories*
- **Awareness** of Systems Complexities while Creating Systems Cost Estimates
  - Impact on Systems Cost Estimates
  - Systems Cost Estimators = Main Audience
    - **Better** Understanding of Systems Complexities
    - **Enhance** Systems Cost Estimating Techniques
- **Support** Current Research - Project Management Life Cycle Costing

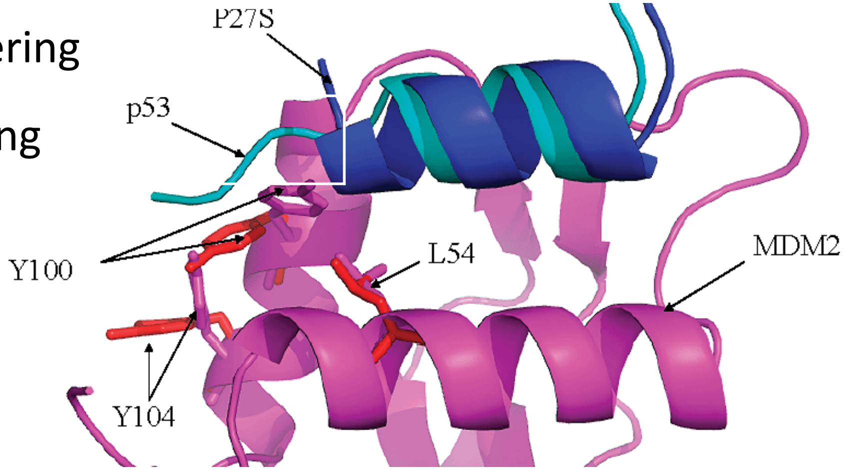
Complexity

"Another group that adopted the word "complexity" is ***wine connoisseurs***. When they say a **wine** is **complex**, they mean they think it **tastes good**, and it strikes a **good balance** between all the different ways a wine can be measured. **Rich** and **deep**, **oaky** with a hint of **fresh** bougainvillea blossoms after a rainstorm on a Tuesday…"

Colwell, B. "Complexity in Design", *IEEE Computer,* Vol. 38, No. 10, pp 10-12, Oct 2005

Complexity Types

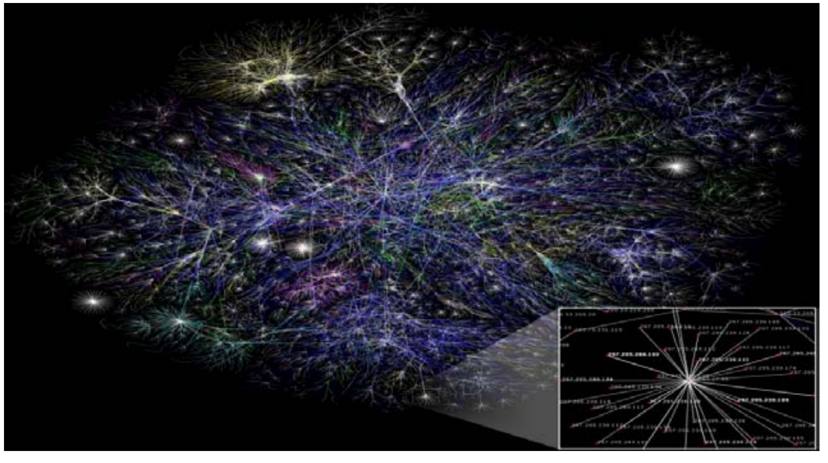
- Literature Reviews
  - Chemistry
  - Biology
  - Systems Biology
  - Geography
  - Computer Science
  - Software Engineering
  - Biogeography
  - Ecology
  - Project Management
  - Management Information Systems
  - Computer Engineering
  - Systems Engineering
- Significantly Investigated
  - Many Disciplines/Domains
  - Many Different Typologies

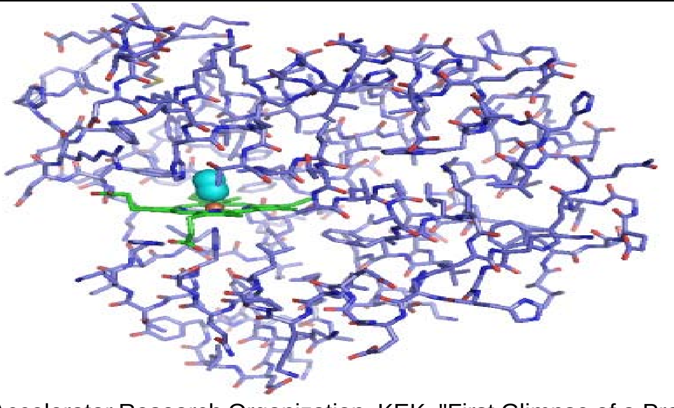


Truhlar, D. G., "Molecular Modeling of Complex Chemical Systems", Journal of the American Chemical Society (JACS), 130 (50), 16824–16827, December 10, 2008

Complexity Types (cont'd)

Discipline	Type of Complexity	Description
Project Management	Project Complexity	Contains two dimensions - organizational and technological, and is consisted of various interrelated parts and can be operationalized in terms of differentiation and interdependency or connectivity (Baccarini, 1996)
	"	Defined by project organization, scope and interconnection between project elements. 3 classification – assembly, system and array (Sauser et al., 2005; Shenhar & Dvir, 1996)
Computer Engineering	Inter-Component Complexity	Defined by the interactions of components at the system level where complexity can grow exponentially due to aggregation in nature (Rumpler, 2006)
	Interface Complexity	Considers an interface of a component in isolation, often cannot be measured in concrete numbers (Rumpler, 2006)
Software Engineering/ Computer Science	Structural Complexity	Defined by the design and structure of software, such as modularity, loose coupling, tight cohesion, control flows (e.g. McCabe – cyclomatic complexity), interfaces and maintainability (Laird & Brennan, 2006; Tran et al., 2002; Fenton, 1994; Lew et al., 1988)

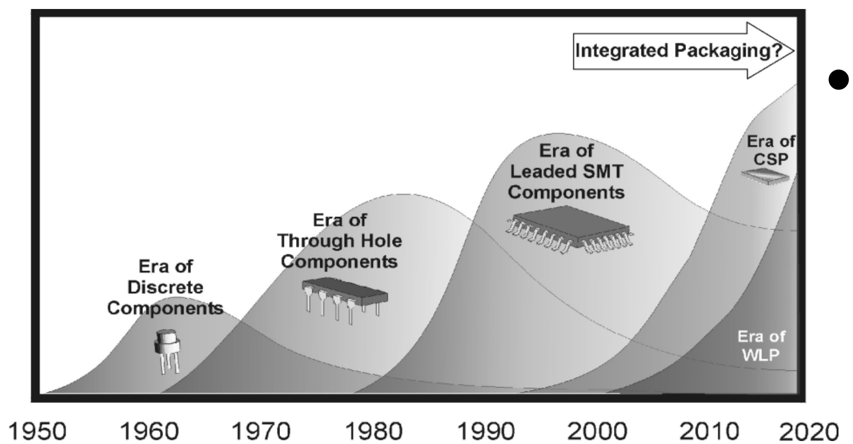




High Energy Accelerator Research Organization, KEK, "First Glimpse of a Protein Molecule Breathing Caught in Action", Feb 12, 2009, retrieved from http://www.kek.jp/intra-e/press/2009/ERATO.html

Complexity Types (cont'd)

Discipline	Type of Complexity	Description
Systems Engineering	Behavioral Complexity	Defined by the difficulty of prediction on system outputs or behavior, also known as dynamic complexity (Mostashari & Sussman, 2009)
	Social-Political Complexity	Describes systemic complexity factors such as human cognitive limitations, economics, environmental sustainability, etc (Sheard & Mostashari, 2009)
	Technical Complexity (Systems Integration based)	The magnitude of technical integration requirements on systems capabilities and functions, interfaces performance, strategies, methodologies at system and subsystem levels (Jain et al., 2008)



Fjelstad. J., DiStefano, T., and Faraci, A., "Wafer Level Packaging of Compliant, Chip Size ICs", *International Journal of Microelectronics, Vol. 17, No. 2, 23-27, 2000*

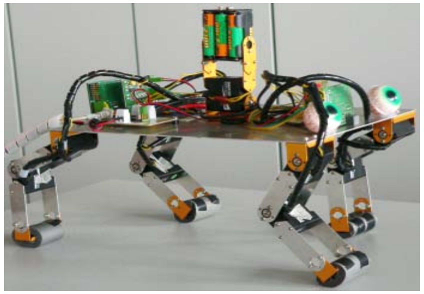
- **Paper documented**
  - ***32* Complexity Types**
  - ***12* Disciplines and Domains**

Quantify Systems Complexities with Systems Artifacts

- Management Information Systems & IT
  - Bhatacharya et al., 2007
    - Artifact-Centered Operational Modeling
  - Dori, D., 2002
    - Object-Process Methodology
  - Nigam and Caswell, 2003
    - Business Artifacts

Honda, ATR and Shimadzu Jointly Develop Brain-Machine Interface Technology Enabling Control of a Robot by Human Thought Alone, Honda ASIMO, retrieved from http://world.honda.com/news/2009/c090331Brain-Machine-Interface-Technology/

- Robotic Engineering
  - Takeda et al., 2002
    - Artifact Intelligence



Universitat Osnabruck, Robotics Research, Walking Test Robot (WALTER), retrieved from http://www.inf.uos.de/techinf/robotics.html

- ***Systems Engineering – Cost Estimating?***

The Constructive Systems Engineering Cost Model (COSYSMO)

$$PM_{NS} = A \cdot \left(\sum_k (w_{e,k}\Phi_{e,k} + w_{n,k}\Phi_{n,k} + w_{d,k}\Phi_{d,k}) \right)^E \cdot \prod_{j=1}^{14} EM_j$$

Where:
PM_{NS} = effort in Person Months (Nominal Schedule)
A = calibration constant derived from historical project data
E = represents diseconomies of scale
k = {REQ, IF, ALG, SCN}
w_k = weight for "easy", "nominal", or "difficult" size driver
Φ_k = quantity of "k" size driver
EM = effort multiplier for the *j*th cost driver.

Systems Sizing Categories

- # of Systems Requirements
- # of Major Interfaces
- # of Critical Algorithms
- # of Operation Scenarios

Mapping and Measuring Systems Complexities with Artifacts

Size Drivers of COSYSMO	Types of Complexity (Exemplary)	Identified From/Relevant (Discipline)	Systems Artifacts (Exemplary)	Measuring Basis By Quantity (Exemplary)
Requirements	Hierarchical/Structural Complexity	Biological Science	Systems Requirement Documents	# of requirements
	Configuration Complexity (Systems Integration based)	SE	Systems Baseline	# of changes, systems specifications
	Technical Complexity (Systems Integration based)	SE	Systems Specification Documents, Test Plan and Form	# of systems capabilities, functions, specifications
System Interfaces	Functional Complexity	Biological Science	Functional Diagram	# of systems functions
	Aggregate Complexity	Geography	Relationship Diagram	# of connections, relationships
	Project Complexity (Shenhar and Dvir, 1996)	PM	Integration Plan	# of integrating activities and tasks
	Inter-Component Complexity	Computer Science	Design Diagrams	# of classes, objects, activities, sequences
	Static/Structural/Combinatorial Complexity	SE	Systems Decomposition Diagram	# of subsystems, components

Mapping and Measuring Systems Complexities with Artifacts (cont'd)

Size Drivers of COSYSMO	Types of Complexity (Exemplary)	Identified From/Relevant (Discipline)	Systems Artifacts (Exemplary)	Measuring Basis By Quantity (Exemplary)
Algorithms	Algorithmic/Deterministic Complexity	Geography/ Biogeography	Line of Codes (LOC)	# of lines of codes
	Computational Complexity	Software Engineering	Data Flow Chart	# of data volume and capability
	Implementation Complexity	Computer Science	Function Point Analysis (FPA)	# of functions
	Conceptual Complexity	Software Engineering	Code Recursions	# of loops
Operational Scenarios	Operational Complexity (Systems Integration based)	SE	Use Cases	# of actors, objects, flows/processes, components
	Configuration Complexity (Systems Integration based)	SE	Concept of Operations	# of operational processes, capabilities
	Nested Complexity	SE	Context Diagram	# of external connections, actors, interactions

The Constructive Systems Engineering Cost Model (COSYSMO)

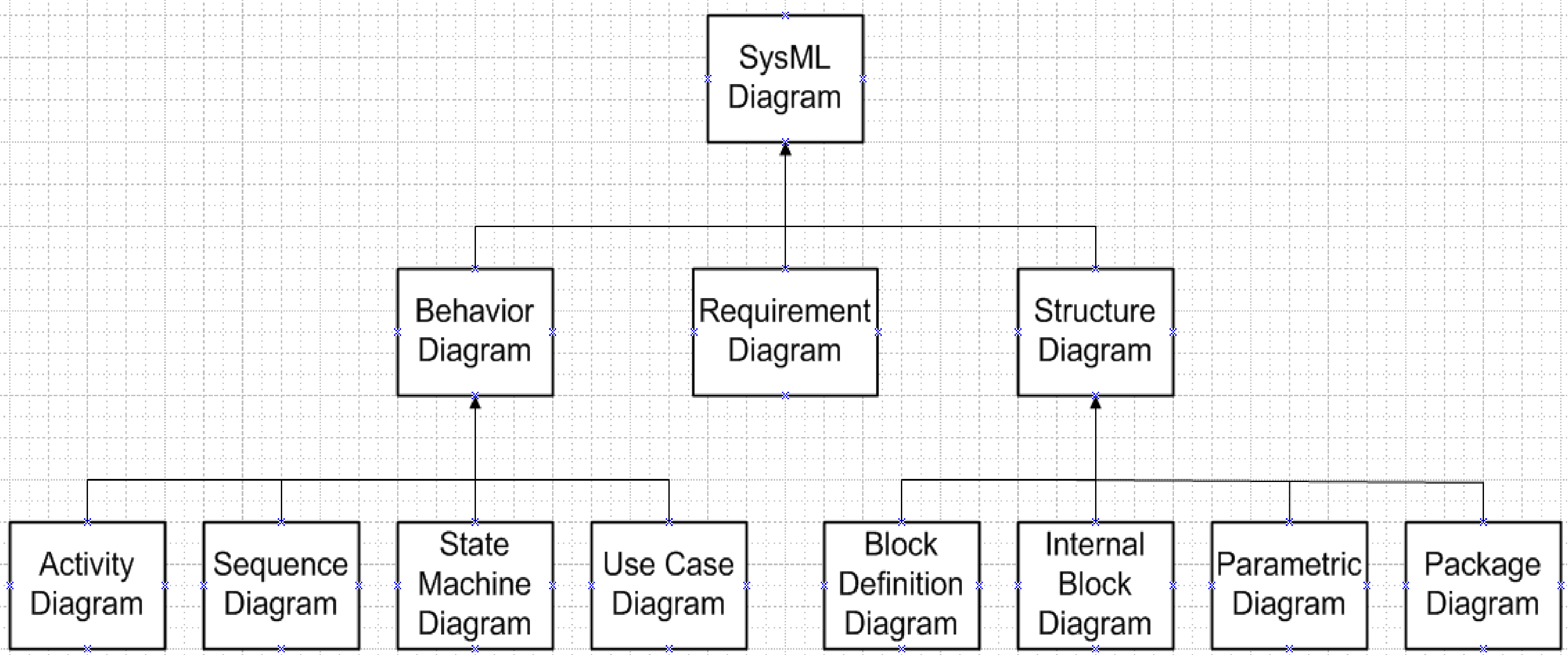
$$PM_{NS} = A \cdot \left(\sum_{k} (w_{e,k}\Phi_{e,k} + w_{n,k}\Phi_{n,k} + w_{d,k}\Phi_{d,k}) \right)^{E} \cdot \prod_{j=1}^{14} EM_j$$

Where:
PM_{NS} = effort in Person Months (Nominal Schedule)
A = calibration constant derived from historical project data
E = represents diseconomies of scale
k = {REQ, IF, ALG, SCN}
w_k = weight for “easy”, “nominal”, or “difficult” size driver
Φ_k = quantity of “k” size driver
EM = effort multiplier for the *j*th cost driver.

Quantified Systems Complexities

- # of Systems Requirements
- # of Major Interfaces
- # of Critical Algorithms
- # of Operation Scenarios

The Systems Modeling Language (SysML) Approach



Friedenthal, S., Moore, A., and Steiner, R., "A Practical Guide to SysML: The Systems Modeling Language," Burlington, MA, Morgan Kaufmann/Elsevier, 2008

Summary & Conclusion

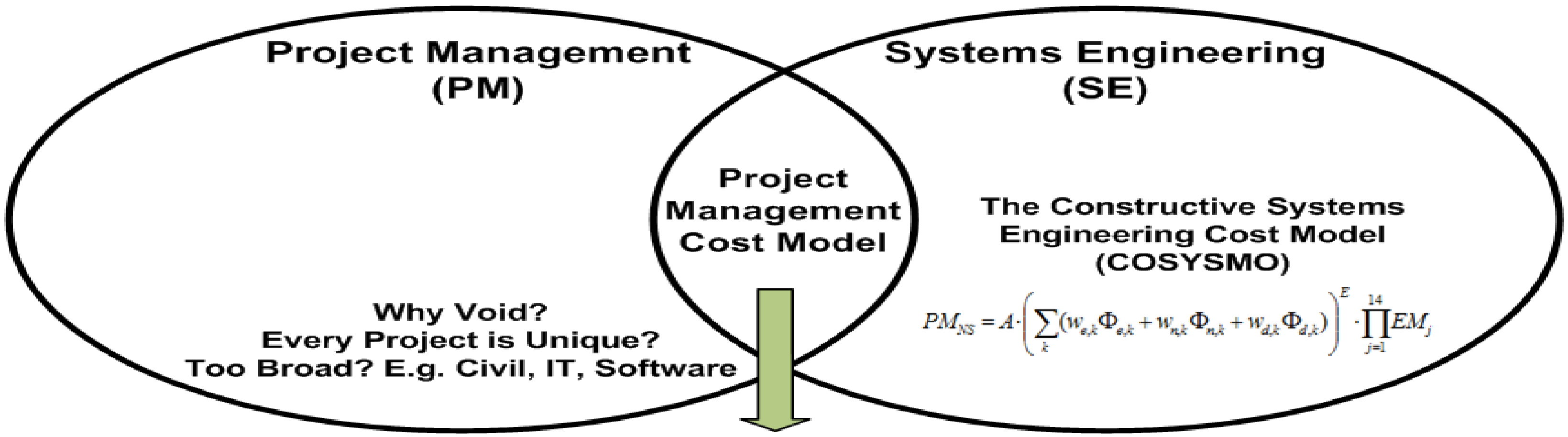
- Systems Complexities = Multidimensional
- Systems Cost Estimators must recognize
  - Systems complexities ***do impact*** on systems cost estimates
  - Systems complexities can be ***inherent in nature*** and may not be captured all in one particular aspect of SE
  - Unreliable estimates ***lead*** to unfavorable project outcomes

Future Work

- Systems Integration Cost Estimating Effort
- Systems Project/Program Management Cost Estimating Effort
  - Complexities
    - Types
    - Roles
    - Depths
    - Durations
    - Magnitude

Project Management Life Cycle Costing Framework & COSYSMO

Project Management Life Cycle Costing Framework & COSYSMO



Similar Approach as COSYSMO -> Intend to be a complementary extension

$$PM = A * (Size)^E * (EM)$$

May Be Based On:

1. Systems Complexities
2. Project Complexities
3. Project Management Effort Multipliers
4. Project Durations
5. Compliance of ANSE/EIA 632
6. Capability Maturity Model (CMM)

Where:

PM = Person Months

A = calibration factor

Size = measure(s) of functional size of a system that has an additive effect on project management effort

E = scale factor(s) having an exponential or nonlinear effect on project management effort

EM = effort multipliers that influence project management effort

Questions?

Suggestions?

Reserve for Back Up

Summary of Life Cycle Cost Estimating Methods

Method	Description	Advantages	Disadvantages
Actual Costs/ Extrapolation Method	Use costs experienced during prototyping, hardware engineering development models and early production items to project future costs for the same system	• Could provide detailed estimate • Reliance of actual development data	• Development data may not reflect cost correctly • Higher uncertainty • Often mistakenly use contract prices to substitute for actual cost • Various levels of detail involvement • Require existing actual production data
Analogy/ Comparative/Case-based Reasoning Method	Compare available data from similar completed projects and adjust estimates for the proposed project	• Reliance of historical data • Less complex than other methods • Save time	• Subjective/bias may be involved • Limited to mature technologies • Reliance of single data point • Hard to identify appropriate analog • Software and hardware often do not scale linearly • Not always possible to find programs of similar scope and complexity
Cost Accounting	Formulate based on the expenditures of reliability, maintainability, and decomposed component cost characteristics	• Reliance of detailed data collection	• Accounting ethics (i.e. cook the books) • Post-production phase strongly preferred • Requires of large and complex data collections • Labor intensive
Detailed Engineering Builds/Bottom-Up	Estimate directly at the decomposed component level leading to a total combined estimate	• Most detailed at the component level through work breakdown structures • Systemic oriented • Highly accurate • High visibility of cost drivers	• Resource-intensive (time and labor) • May overlook system integration costs • Reliance of stable systems architectures and technical knowledge • Highly prone to double-counting • Lacks ability to capture economies of scale
Expert Judgment/ Delphi Method	Produced by human experts' knowledge and experience via iterative processes and feedbacks	• Available when there are insufficient data, parametric cost relationships, or unstable system architectures	• Subjective/bias • Detail cost influence/driver may not be identified • Program complexities can make estimates less reliable • Human experience and knowledge required
Parametric/ Cost Estimating Relationship	Use mathematical expressions and historical data to create cost relationships models via regression analysis	• Statistical predictors provide information on expected value and confidence of prediction • Less reliance of systems architectures • Less subjective	• Heavy reliance of historical data • Attributes within data may be too complex to understand • Possibly resource intensive (time and labor) • Difficult to collect data and generate correct cost relationships during cost model development • Limited by data and independent variables
Top-Down	Based on the overall project characteristics and derived by decomposing into lower level components and life cycle phases.	• Fast and easy deployment • Minimal project detail required • Systemic oriented	• Less accurate than others • Tend to overlook lower level component details or major cost drivers • Limited detail available for justification

conference on systems engineering research 2010

Summary of Complexities

Discipline	Type of Complexity	Description
Chemistry/ Biology/ Systems Biology/Ecology	Hierarchical/Structural/ Configuration Complexity	Number of levels of components are nested within a boundary or environment (Ross & Arkin, 2009; Kolasa, 2005; Kitano, 2002; Edelman & Gally, 2001)
	Complicatedness/Functional Complexity	Number of different elementary components through specific interactions (Ross & Arkin, 2009; Kolasa, 2005; Kitano, 2002; Edelman & Gally, 2001)
Geography /Biogeography	Subjective Complexity	Based on human perception and is a consequence related to observers' own thinking (Reitsma, 2003)
	Statistical Complexity	Statistical measures of structure or pattern, circumventing the problem of statistical complexity where randomness equals maximal complexity (Reitsma, 2003)
	Algorithmic/Deterministic Complexity	Based on information theory, use mathematical computation to measure the algorithmic content of data (Reitsma, 2003; Manson, 2001)
	Aggregate Complexity	Access the holism and synergy resulting from the interaction of systems of linked components. Contains interrelated attributes: relationships, internal structure, environment, learning, emergence and evolutional (Manson, 2001)
Project Management	Project Complexity	Contains two dimensions - organizational and technological, and is consisted of various interrelated parts and can be operationalized in terms of differentiation and interdependency or connectivity (Baccarini, 1996)
	"	Defined by project organization, scope and interconnection between project elements. 3 classification – assembly, system and array (Sauser et al., 2005; Shenhar & Dvir, 1996)
	Product Complexity	The physical deliverable of the project and is the number of subsystems of a product and their interrelationships (Williams, 1999)
Management Information Systems/ Information Technology	Structural Organizational Complexity	Defined by the nature and strength of the relationships among project elements in the organizational boundary (Xia & Lee, 2004, 2005)
	Structural IT Complexity	Defined by the relationships among the diversity of IT elements (Xia & Lee, 2005, 2004)
	Dynamic Organizational Complexity	Defined by capturing the pattern and rate of change in organizational environment, such as business processes, organizational structures (Xia & Lee, 2005, 2004)
	Dynamic IT Complexity	Defined by measuring the pattern and rate of changes in IT environment, such as IT infrastructure, architecture, tools (Xia & Lee, 2004, 2005)

Summary of Complexities (cont'd)

Computer Engineering	Inter-Component Complexity	Defined by the interactions of components at the system level where complexity can grow exponentially due to aggregation in nature (Rumpler, 2006)
	Interface Complexity	Considers an interface of a component in isolation, often cannot be measured in concrete numbers (Rumpler, 2006)
	Implementation Complexity	Describes the complexity of creating and implementing components beyond its interfaces complexity, such as program code (Rumpler, 2006)
	System-level Complexity	Inherent from inter-component complexity and is based on the interactions of the integrated system components in which may cause emergent behavior (Rumpler, 2006)
Software Engineering/ Computer Science	Structural Complexity	Defined by the design and structure of software, such as modularity, loose coupling, tight cohesion, control flows (e.g. McCabe – cyclomatic complexity), interfaces and maintainability (Laird & Brennan, 2006; Tran et al., 2002; Fenton, 1994; Lew et al., 1988)
	Conceptual Complexity	Defined as psychological related measures on comprehension that is difficult to quantify. i.e. logic understanding on code recursion (Laird & Brennan, 2006; Tran et al., 2002; Fenton, 1994; Lew et al., 1988)

Summary of Complexities (cont'd)

Discipline	Type of Complexity	Description
Systems Engineering	Computational Complexity	Defined by the magnitude of algorithms required within processes and procedures to perform computation. Useful to evaluate and compare implementation and designs for efficiency (Laird &Brennan, 2006; Tran et al., 2002; Fenton, 1994; Lew et al., 1988)
	Structural/Combinatorial Complexity	Defined by a large number of highly interconnected parts or subsystems (Mostashari & Sussman, 2009)
	Behavioral Complexity	Defined by the difficulty of prediction on system outputs or behavior, also known as dynamic complexity (Mostashari & Sussman, 2009)
	Nested Complexity	Several complex physical/technical systems embedded within a higher complex, governing social/organizational system (Mostashari & Sussman, 2009)
	Evaluative Complexity	Defined by the magnitude of multiple stakeholder environment where different stakeholders value different aspects of system performance in different way, in which decision-making is difficult (Mostashari & Sussman, 2009)
	Static Complexity	Defined by the size, number of connectivity and architecture of the system (Sheard & Mostashari, 2009)
	Dynamic Complexity	Describes nonlinear, dynamic emergence system behaviors (Sheard & Mostashari, 2009)
	Social-Political Complexity	Describes systemic complexity factors such as human cognitive limitations, economics, environmental sustainability, etc (Sheard & Mostashari, 2009)
	Technical Complexity (Systems Integration based)	The magnitude of technical integration requirements on systems capabilities and functions, interfaces performance, strategies, methodologies at system and subsystem levels (Jain et al., 2008)
	Programmatic Complexity (Systems Integration based)	The variance between the planned and available resources needed to support integration processes over system life cycle, such as budget, cost, schedule, etc. (Jain et al., 2008)
	Configuration Complexity (Systems Integration based)	The magnitude of inconsistencies and system developmental control, such as changes, volatility, documentation, specification, integration baselines (Jain et al., 2008)
	Operational Complexity (Systems Integration based)	The magnitude and level of operational support and system availability required for systems integration effort (Jain et al., 2008)
	Organizational Complexity (Systems Integration based)	Defined by the nature and existence of organizational strategy, compliance, processes and product line. i.e. service level agreements, regulations and guidelines for systems integration process and effort (Jain et al., 2008)