Next-Generation Systems Engineering: Expansion, Foundation, Unification
21 June 2011

Presented at INCOSE Symposium 2011
Denver, CO

Duane Hybertson
The MITRE Corporation
McLean, VA
dhyberts@mitre.org

Topics

- Expansion
  - Traditional SE is expanded to address more complex or “wicked” problems: pandemics, criminal justice, irregular warfare, …
- Foundation
  - The expanded SE is grounded on a science of systems engineering that comprises multiple existing sciences, including systems science, complex systems science, social and biological sciences
- Unification
  - The traditional and more complex elements of the expanded SE, as well as the supporting science foundation, are unified into a holistic and grounded next-generation multidiscipline
- A Proposed Model-Oriented Approach

Contrast of Traditional SE and Complex SE

TSE	CSE
• Mechanistic systems that have relatively predictable behavior	• Organisms whose behavior is to some degree unpredictable
• Components are machines or mechanistic elements	• Components are considered to be people or organic or autonomous elements
• Are relatively stable, and change only by external agent	• Are adaptive and learn, grow, evolve, change on their own
• Are designed and organized by an external designer	• Are self-organizing, and the whole system emerges over time

SE has persisted in excluding CSE:

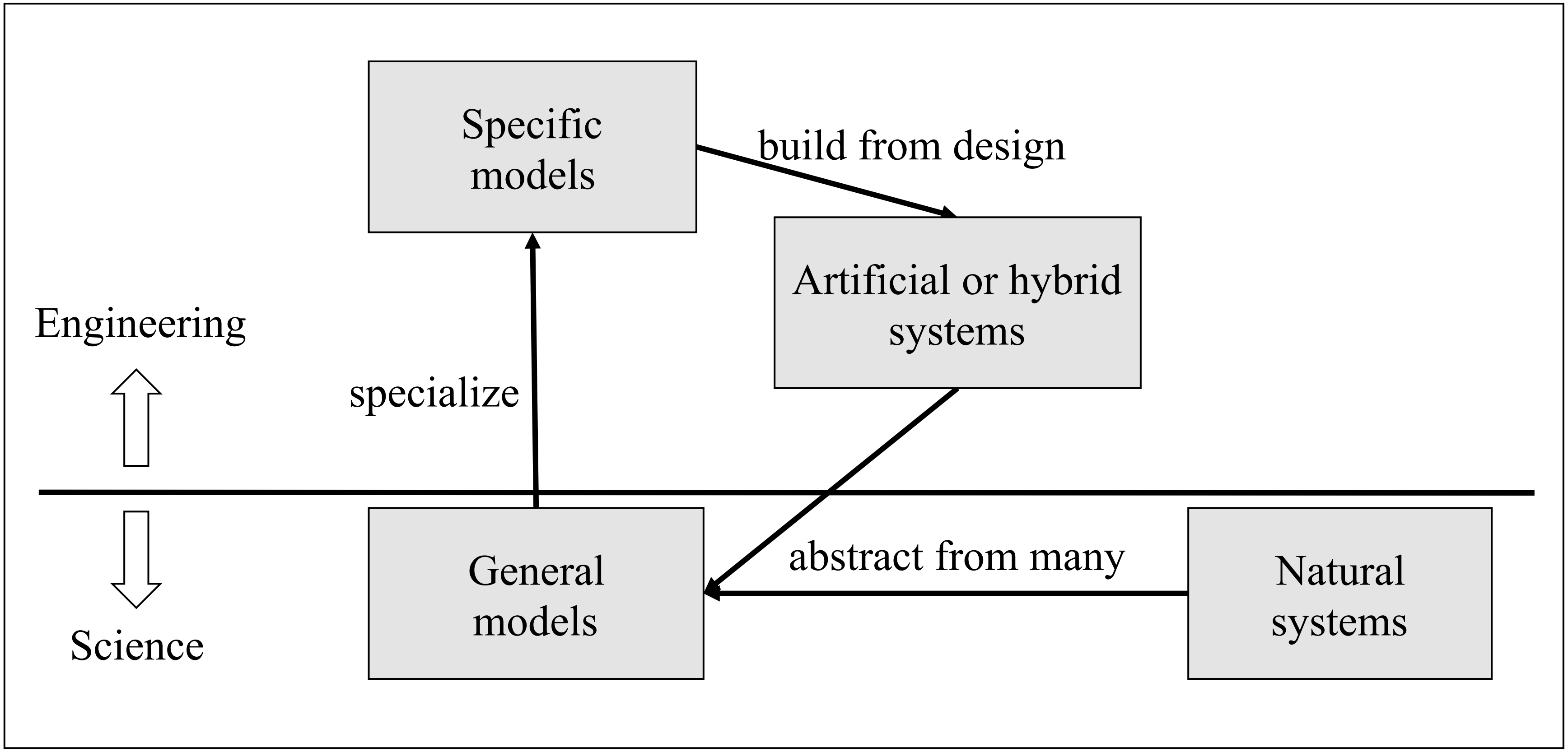
- Goode and Machol (1957): Acknowledged common points of interest with social, biological, and ecological systems, but excluded them from the scope of SE.
- Kossiakoff and Sweet (2003): SE excludes complex systems such as social structure, living organisms, and eco-systems.

But now SE has begun expanding

- Why should social, organizational, biological, ecological systems be excluded from SE?
  - We see repeated and continuing evidence of lack of systems thinking in these areas and the resulting negative and unexpected consequences. Wouldn't an SE approach help?
- From Braha et al., 2006. *Complex Engineered Systems: Science Meets Technology*.
  - Systems such as the Internet, power grids, markets, multinational enterprises, and transportation networks have different characteristics; they continually change, and grow, and involve significant uncertainties. TSE world view is not well equipped to handle this type of system.
- From MIT Engineering Systems Division strategic plan:
  - Working at the frontiers of … larger and more complex systems for energy, the environment, communications, health care, manufacturing, and logistics. Many challenges involving these big, "messy" systems stem from the interactions of people, organizations, and technology
  - Tackling engineering systems challenges requires perspectives from **engineering, management, and social sciences** to explore the fundamental structures underlying systems and to frame and model problems so that they can be rigorously addressed.
- INCOSE Revitalization Project (Mackey et al. 2003)—Challenge areas for SE
  - International terrorism, global warming, AIDS epidemic, international energy policy, clean water, delivery of healthcare…

The SE field is beginning to reach out to CS concepts

Science-Based Engineering



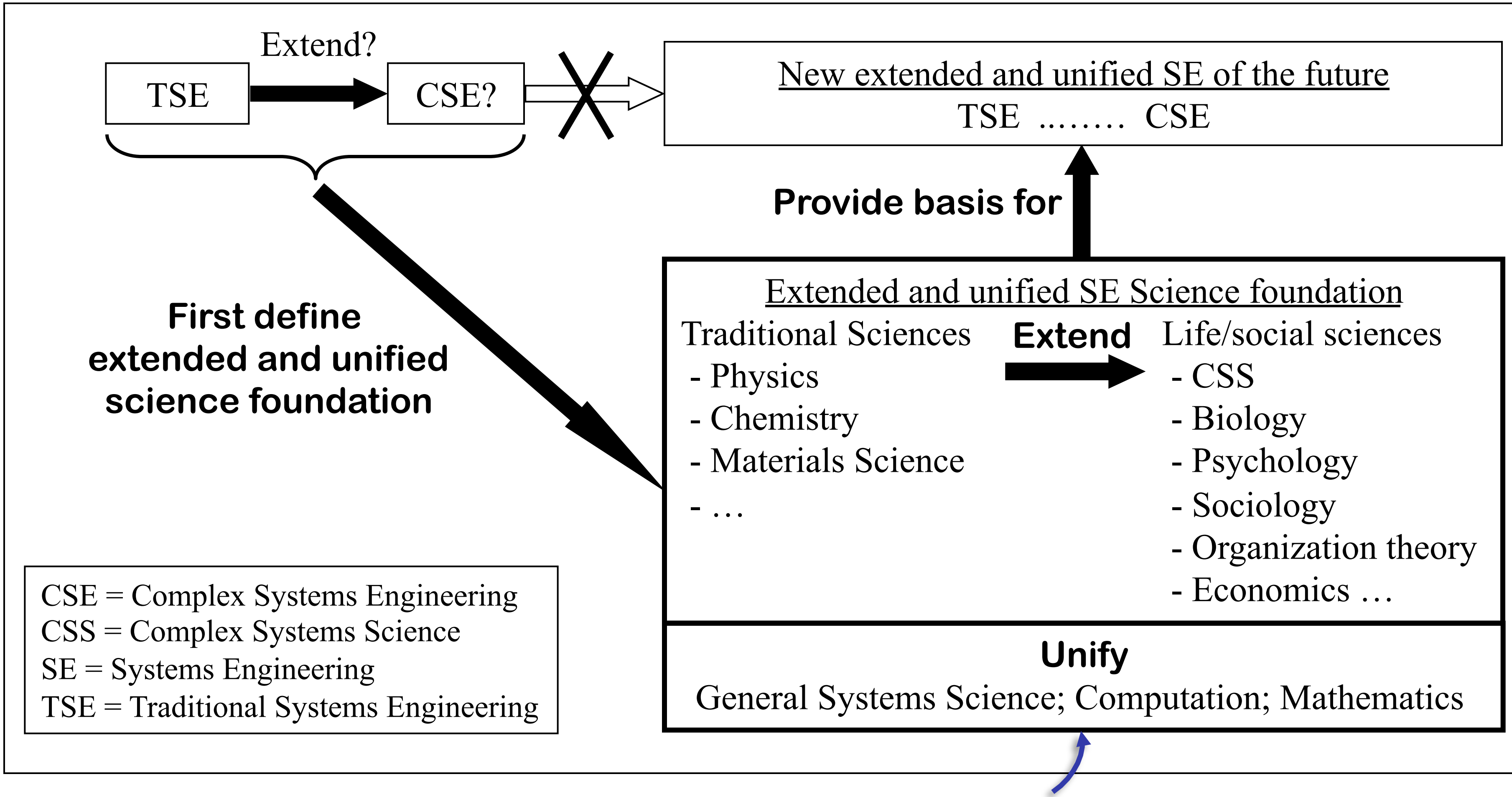
SES Foundation disciplines of SE (future)

INCOSE Complex Systems Working Group → Complex systems science

INCOSE Systems Science Working Group → Systems science

Sciences	Mathematics	Other Supporting Disciplines
Anthropology	Algebra	Artificial life
Bioinformatics	Category theory	Conceptual modeling
Biology	Complex functions theory	Decision theory
Catastrophe theory	Continuous math (calculus...)	Epistemology
Chaos theory	Discrete math (logics...)	Forecasting
Chemistry	Dynamical systems	Game theory
Cognitive science	Factor analysis	Information theory
Complex systems science	Geometry	Knowledge management
Cybernetics	Graph theory	Knowledge representation
Ecology	Nonlinear systems	Law
Economics	Numerical analysis	Mechatronics
Epidemiology	Probability & statistical theory	Mereology
Geology	Topology	Public policy
Linguistics		Ontology
Medical/health science	**Computation, Informatics**	Operations research, management science
Military science	Artificial Intelligence	Simulation
Network science	Computer science	Small worlds
Organization theory/science	Computing science	Urban planning
Physics-classical, theory of relativity, quantum theory	Formal languages and methods	
Political science	Formal semantics	
Psychology	Software science	
Sociology		
Systems science		

How Can We Expand SE? Need Foundation and Unification



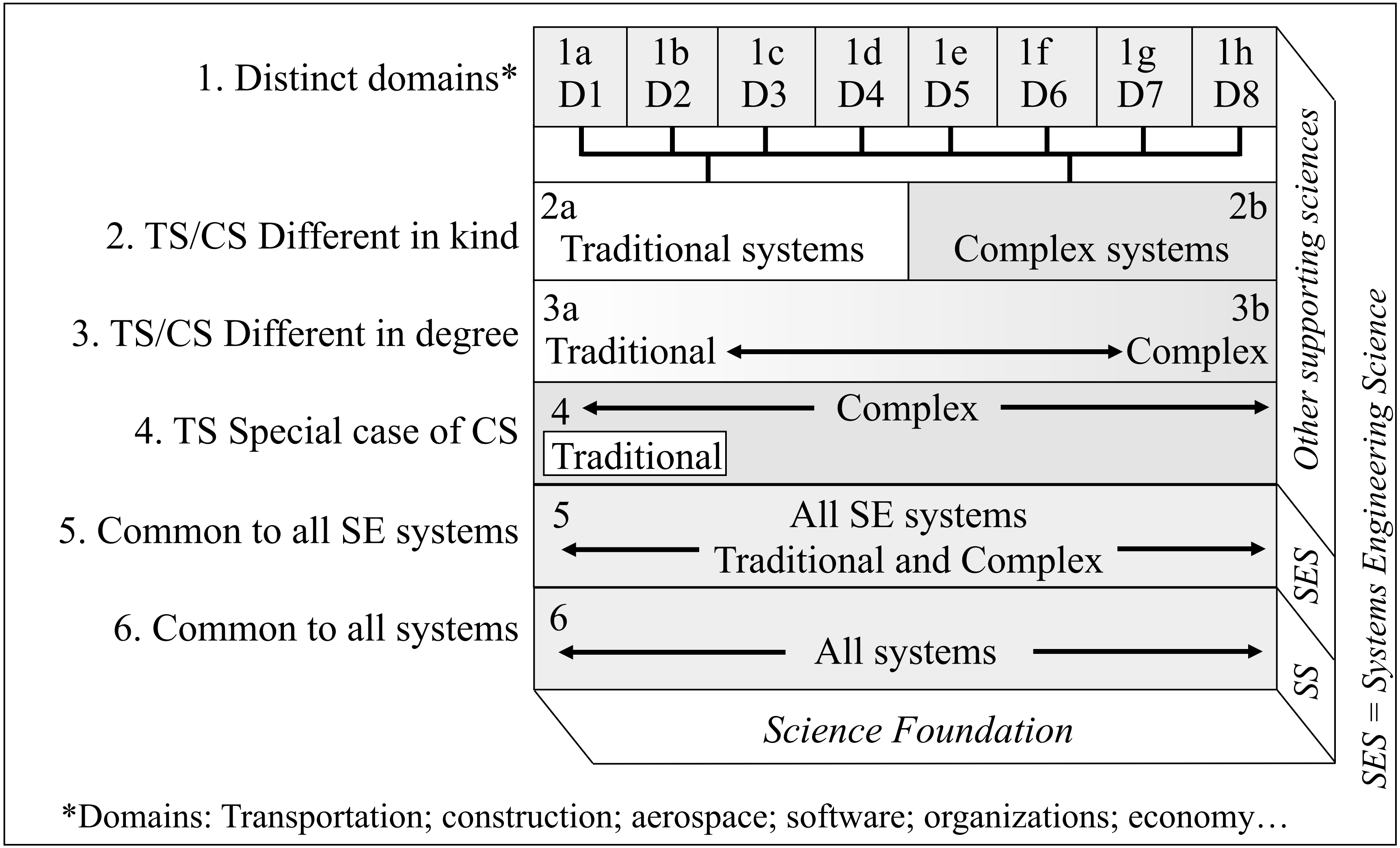
Needed solution: An expanded and unified SE science (SES)

SE needs to strengthen its foundation to expand and unify

A Model-Oriented Approach

- Unifying expanded system taxonomy and science foundation
- Model orientation
- Model space: structure; body of knowledge
- Model space context: Collective actualization
- Benefits of model-oriented approach

Systems Taxonomy on a Science Foundation



Relation between TS and CS is not monolithic; it is multi-faceted

Model Orientation

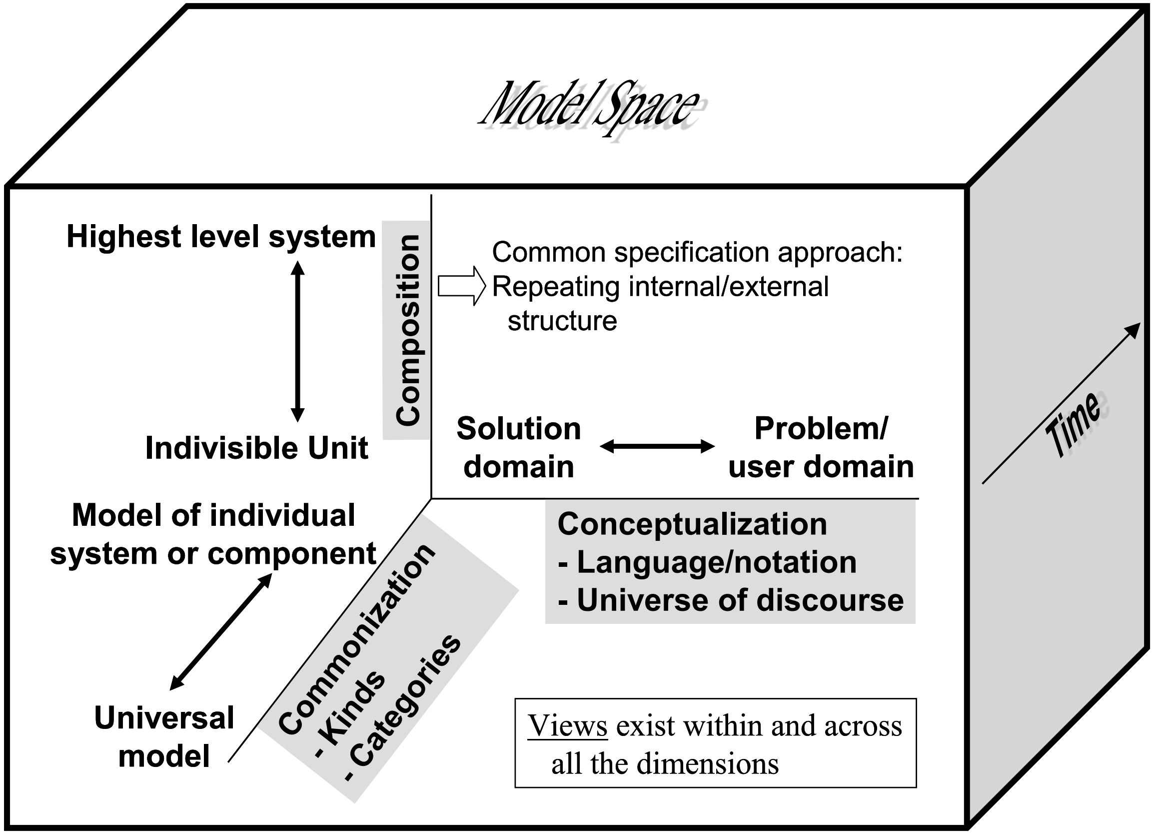
- Orientation: All types of system descriptions are usefully portrayed as models
  - Science: Theories are models of classes of existing systems (natural or artificial)
  - Engineering: Artifacts (concept of operations, requirements spec, architecture description, design spec…) are models of potential (to-be) or existing (as-built) systems or classes of systems
  - Mathematics: Equations and other relations are models of classes of systems
- All these models can be defined in terms of rules, constraints, and other relations

Model Space Structure

Dimensions: 4 + 1

- Composition: Whole/part holarchy
- Commonization: Distinctions; similarities/differences
- Conceptualization: Description, models, languages, ontology
- Time: Change, adaptation
- Views: Cross-cutting models from given perspective or concerns

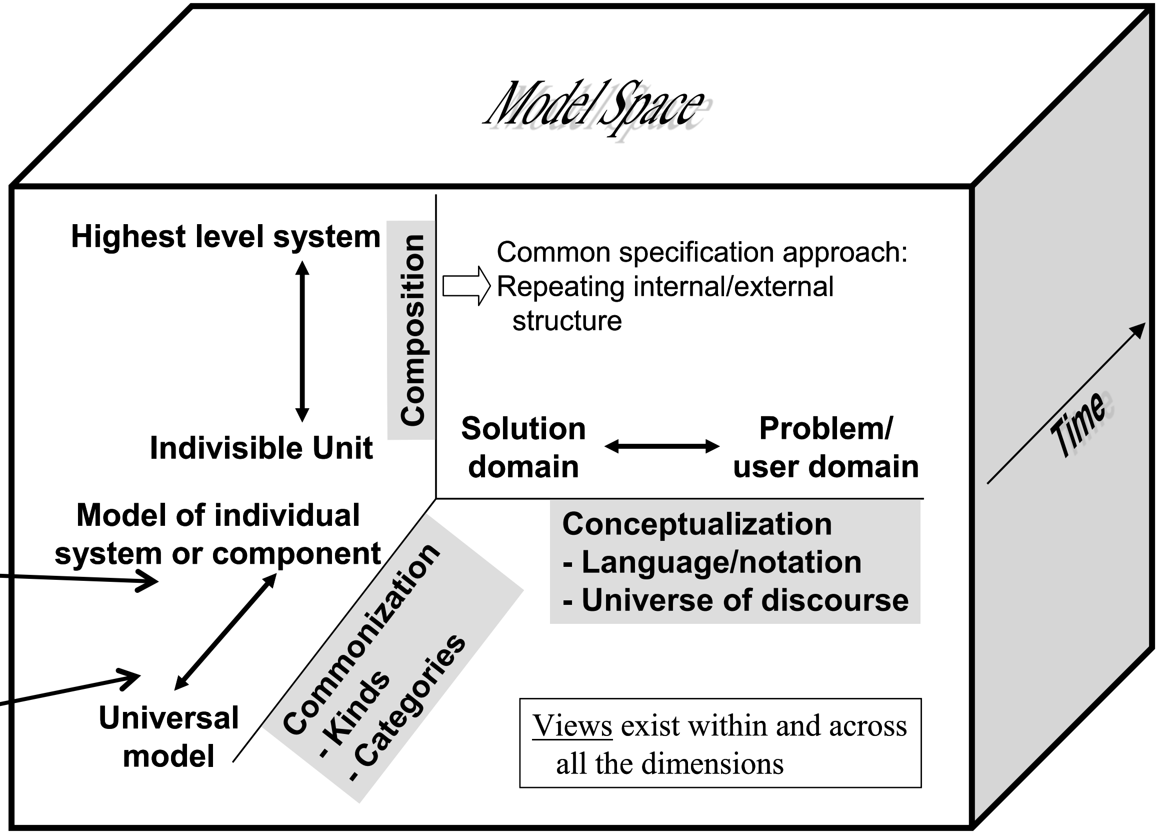
Model space contains all models associated with SE: engineering, science, math



Model Space Support of SE Body of Knowledge

Specific models: Project information about specific systems

General models: SE community body of knowledge of SE domains and SE supporting sciences



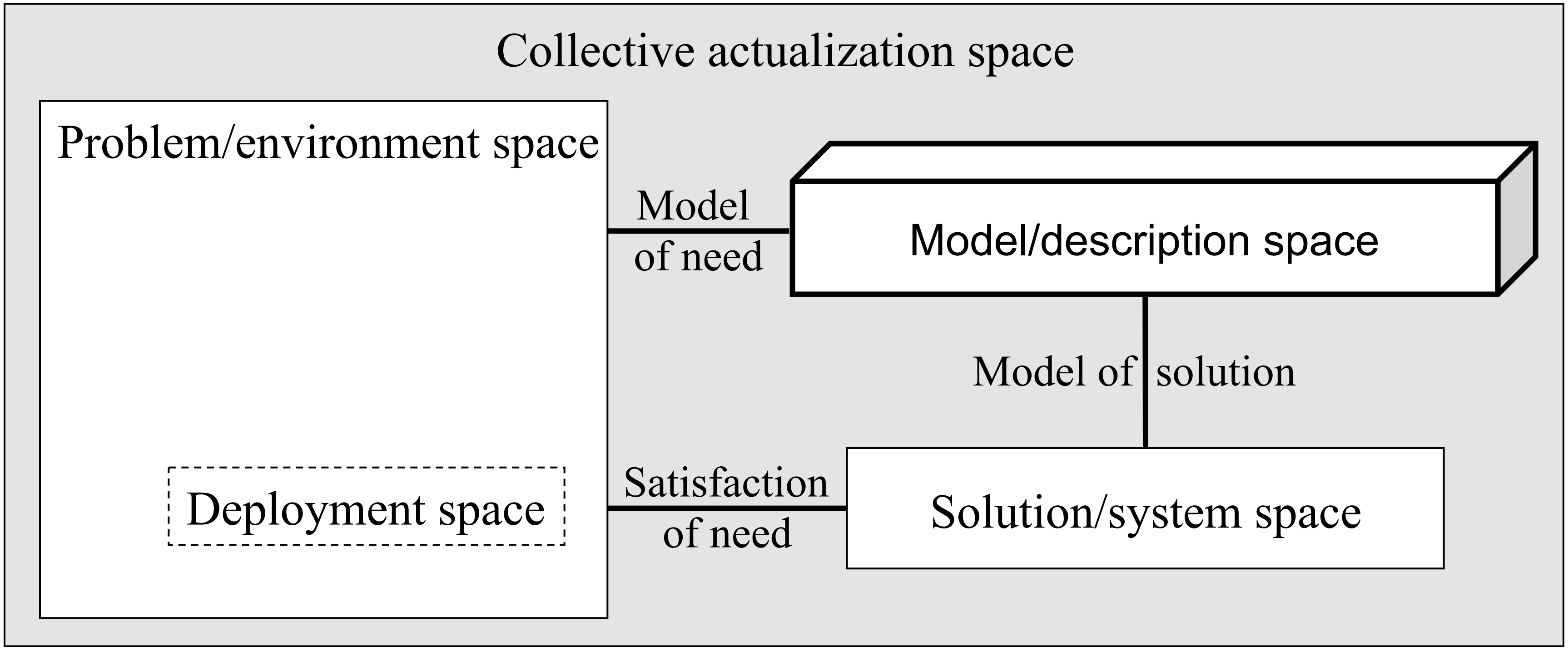
TSE process: Engineering, Management
The expanded SE process: Collective Actualization

Collective actualization process includes engineering, managing, and self-organizing

	Engineering	Autonomy
Enterprise	EEM enterprise engineering/management	EA enterprise autonomy
Produced System	SE system engineering	SA system autonomy

Encompasses mix of engineering/management (‘other’) and autonomy (‘self’) for both enterprise and produced system; covers TS-CS spectrum

Context: Collective Actualization Space



Problem space: Problem domain environments in which engineered systems are needed – and eventually used in the deployment space

Model space: Models of problems and solutions, ranging from general models representing knowledge of multiple systems to specific models of a problem or solution

Solution space: Solution engineered with the help of models and that satisfies one or more needs in problem environment; in the preponderance of cases, the solution is a system

Collective actualization space: Engineering agents, self-organizing and managing agents, interaction, modeling/engineering/management processes and tools

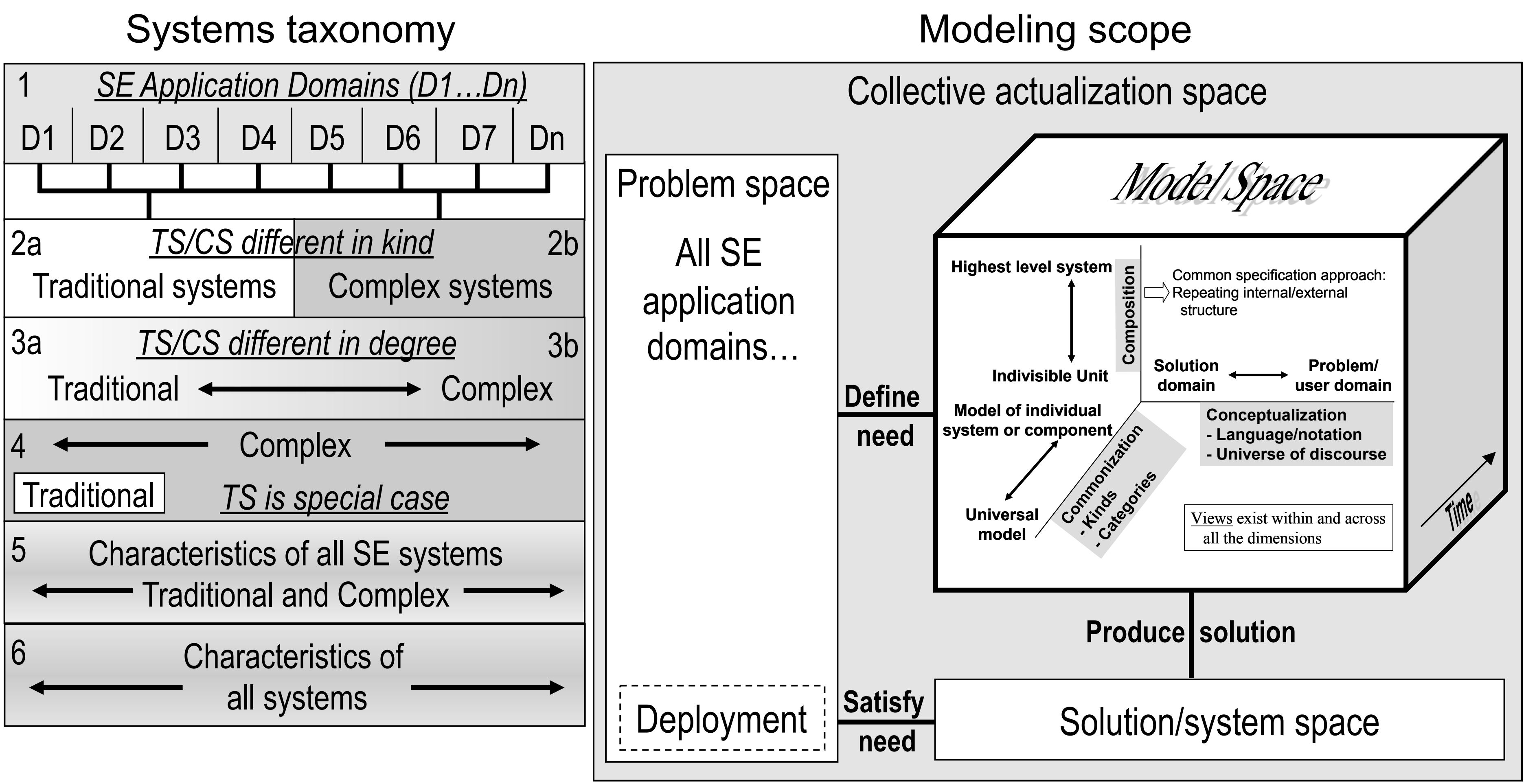
Model Orientation Benefits

- Models simplify and clarify our work; they help us focus on characteristics of interest and ignore the rest
- Because of the power of abstraction, the modeling approach scales up and down
- Model orientation unifies TS and CS
  - TS CONOPS ⇔ influence of environment on a CS: both are models of the interaction of a system with its environment.
  - TS requirements specification ⇔ focus on a CS as a whole: both are models of the externally visible behavior and properties of a system.
  - TS architecture or design ⇔ focus on a CS as a set of interacting parts: both are models of the internal system components and their interaction.
- General models capture knowledge and enable the SES foundation: reference architectures, patterns, scientific theories in the various contributing disciplines
- Software and SE both moving toward model orientation (MDA, UML, SysML, BPMN, MBSE, MDD, MDE), but are not quite there yet

Model orientation is a major enabler of SE expansion, foundation, unification

MDA=model driven architecture; UML=unified modeling language; SysML=systems modeling language; BPMN=business process modeling notation; MBSE=model-based systems engineering; MDD=model-driven development; MDE=model-driven engineering

Model-Oriented Systems Engineering Science (MOSES): Composite View



- Systems taxonomy partitions SE systems and shows traditional and complex systems relations
- Collective actualization space includes all SE and management processes, artifacts, contexts
- Problem space includes all application domains SE supports and in which systems are deployed
- Model Space contains all engineering, science, math, and management artifacts (as models)
- Solution/system space includes all solutions/systems produced by actualization processes

BACKUP

Synopsis: SE Challenges

Challenges and opportunities

Weaknesses of traditional SE

Attempts to incorporate people, organizations, social issues

Address full scope of TS, CS

Scale up to multi-organizational, multi-national enterprises, SoS

Scale down to nanotechnology, biosystems

Proposed solution

Expansion: Add CSE, self-similar scaling; include architecting, EE/EA/ESE under the SE umbrella

Unification: Unify TSE + CSE, common taxonomy

Foundation: Fuller incorporation of SS, CSS, and other supporting disciplines

Orientation: Model orientation, model space, SE body of knowledge

Scope of proposal: Future SE science that provides basis for development of future SE methods

CS = complex systems
CSE = complex SE
CSS = CS science
EA = enterprise arch.
EE = enterprise eng.
ESE = enterprise SE
SoS = system of systems
SS = systems science
TS = traditional systems
TSE = traditional SE

State of affairs of TSE

- Traditional SE (TSE) has many significant accomplishments in a number of application domains over the past half-century
- … but a significant number of TSE projects fail: exceed schedule or budget, or do not satisfy requirements, or do not work.
- From 2008 National Research Council report: Amount of time for the DoD to procure a major system is two or three times as long as it was 40 or 50 years ago.
  - A sample of major programs from 1945 through 1970, including Manhattan project and Apollo program, ranged from 2 ½ to 8 years
  - A sample over the past three decades ranged from 11 to 20 years.
- The view that the world stands still while an SE project spends one or two decades to build a system is not realistic
  - Systems are obsolete or no longer needed by the time they are built

Conclusion: TSE needs to be made more effective and agile

The debate over "complexity"

- Complex system enthusiasts: "We are bringing complex systems into the domain of SE."
- Experienced systems engineers: "Huh! We have been engineering complex systems for decades!"
- My position: Both camps have a point. SE has been engineering complex systems for decades. But I do think there is a change; the classes of systems being considered by SE ("CSE") goes beyond traditional SE. Recall the consistent message of these two books on SE a half century apart:
  - Goode and Machol (1957): Acknowledged common points of interest with social, biological, and ecological systems, but excluded them from the scope of SE.
  - Kossiakoff and Sweet (2003): SE excludes complex systems such as social structure, living organisms, and eco-systems.

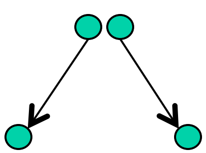
➔ SE/CSE is now <u>including</u> more of these types of complex systems, and applying more <u>models</u> of these types of complex systems

Relation of General SS with Complex Systems/CSS

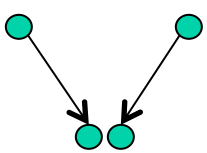
- There is a historical tension in the general SS and Complex Systems Science (CSS) community. The literature – both historical and current – shows intertwining and even conflation between the two. This is reflected to some degree in the efforts to define and relate SSWG and CxSWG in INCOSE
- My position: the two are distinct; both are needed
  - While both SS and CSS are multidisciplinary, the scope of CSS is a subset of SS
  - CSS is concerned with complex or organic systems across multiple disciplines
  - SS is concerned with all systems, which includes — for the purposes of SE — both complex/organic and traditional/mechanistic
  - Natural or Complex systems engineering (CSE), supported by CSS, forms the dominant area of the ***extension*** from TSE to the new SE and supporting SES
  - SS forms the dominant ***foundation*** for the full extended range of SE and SES

The tensions and paradoxes of CS

- A dominant but infrequently noted characteristic of CS is the balance and resolution of tension, contradiction, paradox of elements and features within the same system
  - A good model of this characteristic is yin-yang. Common tensions are stability/change; top-down/bottom-up; unity/variety; homeostatic/adaptive
- Example: Humans exhibit behavior that is both chaotic (sensitive to initial conditions) and purposeful (insensitive to initial conditions). Everyday examples:



  - Sensitive is the environment influence on behavior (go with the flow): After graduate school a person chooses between similar companies A and B, which can lead to very different career trajectories.



  - Insensitive is the purposeful influence on behavior (stay on course): A person has a strong drive to achieve a certain career goal, but may start out in either of two very different situations A and B and still reach the goal
  - The design process is a combination of both: chaotic and purposeful
- Example: SoS: Just another system or different kind of system? ➔ both

CS are not a monolith; they reconcile a full spectrum of opposites

Examples of SE Expansion (cont'd)

- Another example of SE expansion toward CSE: INCOSE Revitalization Project identified these challenge areas as amenable to the application of SE [documented in INCOSE paper "The Role of Systems Engineering in Combating Terrorism" (Mackey et al. 2003)]:
  - Reduction and eradication of international terrorism
  - Reduction of global warming
  - Eradication of the AIDS epidemic
  - Creation of an international energy policy
  - Provision of clean water supplies
  - Reduction of air and water pollution
  - Delivery of healthcare to disaster areas
  - Expansion of international agricultural production
  - Prevention of drug trafficking and abuse
  - Provision of affordable housing

Examples of SE Expansion (cont'd)

Gartner has identified seven properties that differentiate emergent architecture from the traditional approach to EA:

1. **Non-deterministic -** In the past, enterprise architects applied centralised decision-making to design outcomes. Using emergent architecture, they instead must decentralise decision-making to enable innovation.
2. **Autonomous actors -** Enterprise architects can no longer control all aspects of architecture as they once did. They must now recognise the broader business ecosystem and devolve control to constituents.
3. **Rule-bound actors -** Where in the past enterprise architects provided detailed design specifications for all aspects of the EA, they must now define a minimal set of rules and enable choice.
4. **Goal-oriented actors -** Previously, the only goals that mattered were the corporate goals but this has now shifted to each constituent acting in their own best interests.
5. **Local Influences:** Actors are influenced by local interactions and limited information. Feedback within their sphere of communication alters the behaviour of individuals. No individual actor has data about all of an emergent system. EA must increasingly coordinate.
6. **Dynamic or Adaptive Systems:** The system (the individual actors as well as the environment) changes over time. EA must design emergent systems sense and respond to changes in their environment.
7. **Resource-Constrained Environment:** An environment of abundance does not enable emergence; rather, the scarcity of resources drives emergence.

Application of Model-Oriented SES to SE

- Can we engineer complex autonomous systems?
- Connection model: Provides common model for designing interactive parts; for network design and analysis; …
- Holarchy model (composition): Provides an elegant way to scale up to the largest system of systems or enterprise, and scale down to nanosystems
  - Can use the same approach at each level
- Commonization model: Provides common approach to capture similarities (e.g., via holarchy model) and differences (e.g., via specialization or instantiation)
- Applying commonization model to holarchy/composition yields simplification of specification types into two: <u>Internal</u> spec and <u>external</u> spec, intertwining up and down the composition stack

Is engineering complex/natural systems an oxymoron?

- Engineer: design, lay out, decide, organize, construct…
- Complex/natural system: autonomous, self organizing, continually changing…
- ➔These seem incompatible

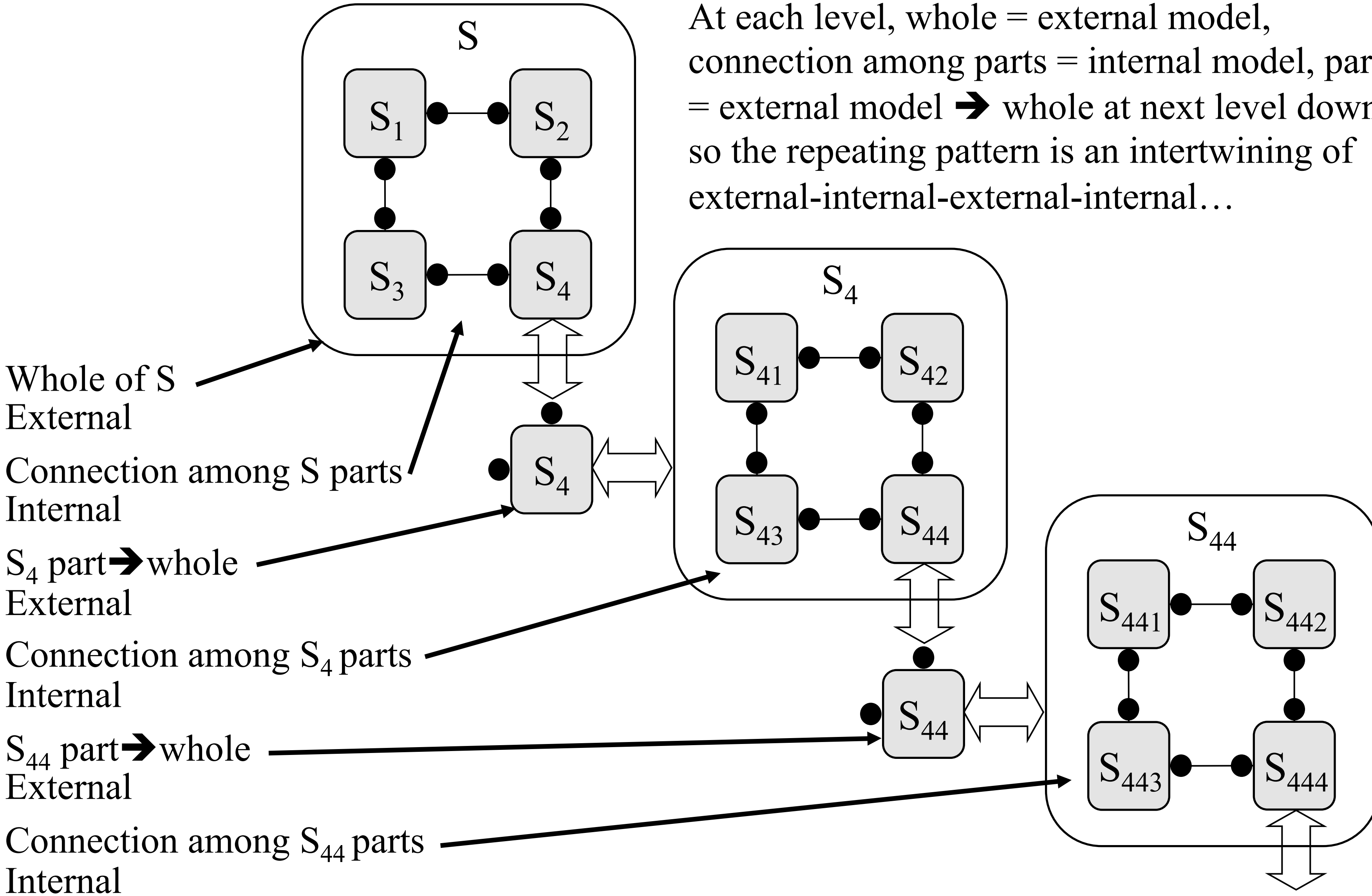
- Harmonized view: The systems engineer has less control than in a traditional system but still has influence
  - Performs a combination of engineering planning and design along with facilitating and shaping, arranging the environment
  - Engineering a complex system seems more difficult; but in many respects it can be easier: The systems engineer does not have to decide everything. Let the system figure things out for itself.
  - Engineering a CS is therefore a joint effort between the systems engineer and the system itself

Modest Proposal: Integrate INCOSE

- Integrate/synergize/align in this common framework several (or all?) INCOSE efforts
  - SSWG: MOSES science foundation
  - CxSWG: MOSES SE expansion
  - SE Body of Knowledge (SEBoK): MOSES model space
  - Model-Based SE (MBSE): MOSES model orientation
  - also other efforts such as certification, handbook, Anti-Terrorism International Working Group…
- Integrating these into a common framework would bring out the implicit synergies among the efforts and increase INCOSE influence in the SE community

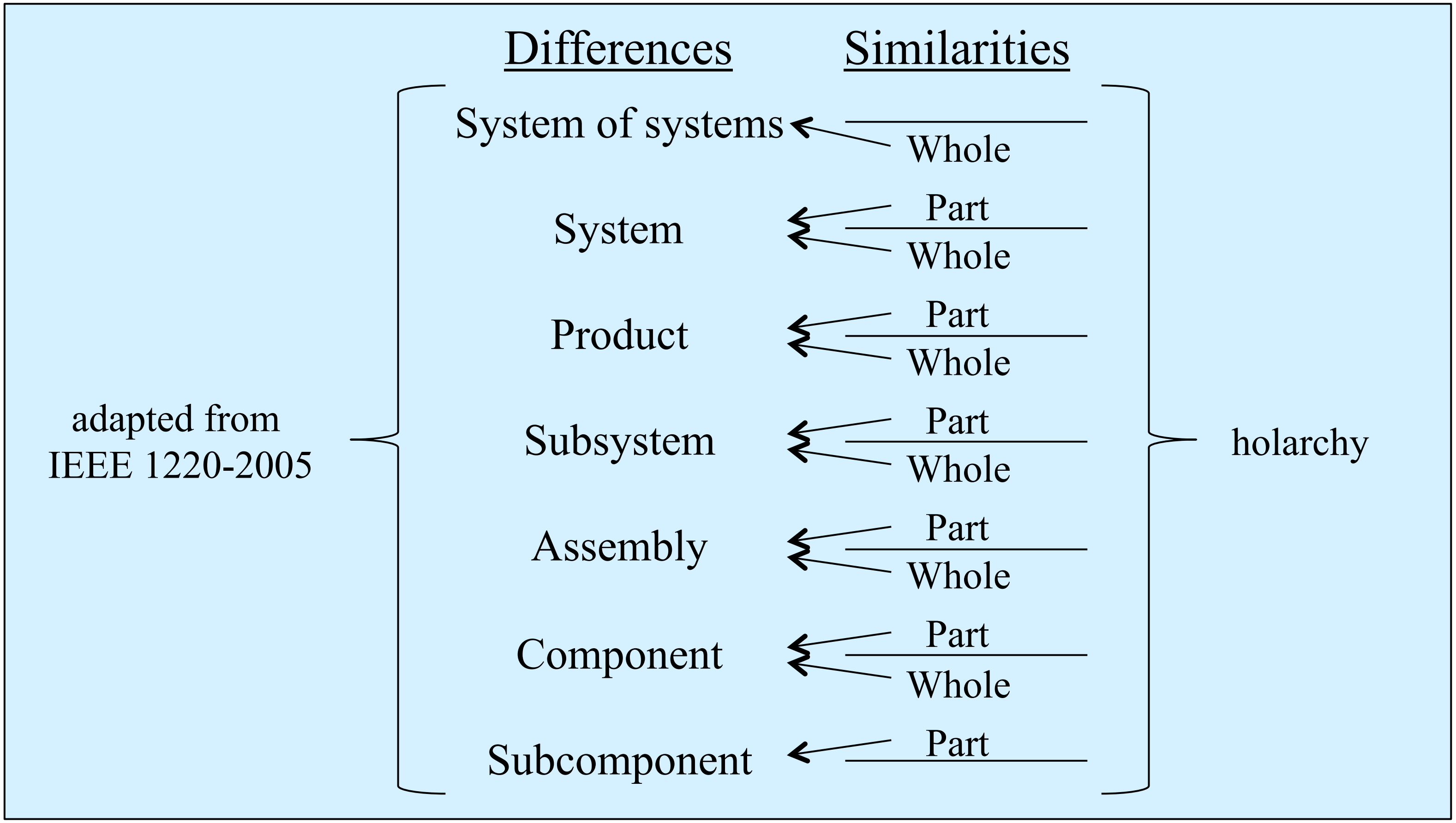
Two spec types cover the full composition range

At each level, whole = external model, connection among parts = internal model, part = external model ➔ whole at next level down; so the repeating pattern is an intertwining of external-internal-external-internal…



Commonization and Holarchy

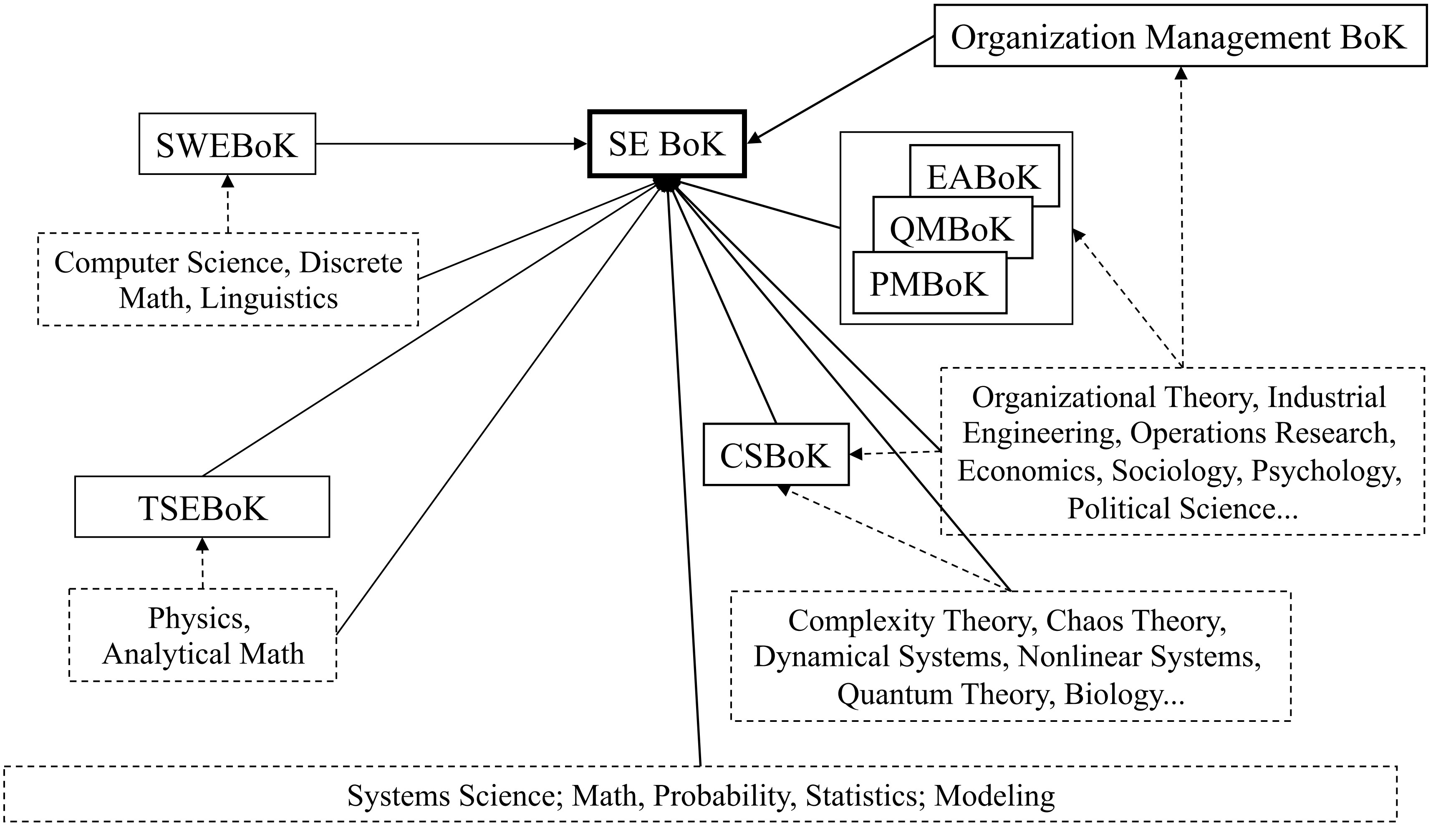
- IEEE 1220 focuses on differences at each composition level
- Holarchy model focuses on similarities across all levels
- SE needs combination of both: provided by the commonization model



Concept of Model

- A model is an explicit approximation, representation, or idealization of selected aspects of the structure, behavior, operation, properties, or other characteristics that can be associated with one or more systems.
  - Adapted from IEEE Standard Glossary of Modeling and Simulation Terminology. IEEE Std 610.3-1989.

SES Body of Knowledge Contributing Elements



Key: BoK=Body of Knowledge; CS=Complex Systems; EA=Enterprise Architecture; PM=Project Management; QM=Quality Management; SWE=Software Engineering; TSE=Traditional Systems Engineering