
***Improving Success with
Technology using an
Organizational Epistemology –
A Conceptual Decision
Framework for Early SE***

***Dr. Chris R. Powell
Univ. of Wisconsin – Eau Claire
Sr. Technical Director, HPTi, Reston, VA***

October 27, 2010

Contact Info

- **Dr. Chris R. Powell**
 - **High Performance Technologies, Inc. (HPTi)**
- **cpowell@hpti.com**
- **(715) 379-9696 (cell)**

- **Frame SE/SoSE Problem as Knowledge Management (KM) issue**
- **Recent Research**
 - **KM in Technology Decision-Making**
 - **Conceptual Decision Framework for Improving Technology Success**
- **So What?**
 - **Model-Based SE Approach**
 - **Complex Systems Realization Framework**

- **Frame SE/SoSE Problem as Knowledge Management issue**
- **Recent Research**
 - **KM in Technology Decision-Making**
 - **Conceptual Decision Framework for Improving Technology Success**
- **So What?**
 - **Model-Based SE Approach**
 - **Complex Systems Realization Framework**

Fundamental SE/SoSE Issue

- **OSD needs to reconcile tension between:**
 - **Increasing acquisition speed**
 - **Meeting more complex requirements**
- ***Contention:* This is significantly a knowledge structure and management issue**

Key Solution Factors

- Notion: Expedite time-consuming acquisition steps by providing and ensuring the following, starting early in the lifecycle:
 - Rigorous recording of assumptions, parameters, constraints, and other information through models, attributes, and metadata
 - Rigorous correspondence of artifacts across lifecycle steps/phases/etc. through common program taxonomy and ontology (e.g., model framework and metadata)
 - Quick, comprehensive testability of assumptions through simulation
 - Simultaneous, early, and ongoing consideration of engineering and program design issues to ensure risk prevention

Creation, discovery, structure and use of more precise, accurate knowledge

- **Frame SE/SoSE Problem as Knowledge Management issue**
- **Recent research**
 - **KM in Technology Decision-Making**
 - **Conceptual Decision Framework for Improving Technology Success**
- **So What?**
 - **Model-Based SE Approach**
 - **Complex Systems Realization Framework**

Research Problems

- **Problem: IT Investments fail a lot (45-90%)**
- **Problem: (IT) Investment evaluation and justification is:**
 - **Highly subjective (executive “gut”)**
 - ❖ Basically, satisficing and bounded rationality
 - **Imperfectly objective (reliance on overly constrained financial metrics)**
- **Problem: IT consideration requires extensive knowledge outside domain of business/mission process**
- **Problem: IT vendors lack appreciation for business/mission knowledge**
- **Need new framework for (IT) investment evaluation and justification:**
 - **Reconcile human nature**
 - **Oriented to Tacit Knowledge**
 - **Concept: “Set” knowledge to be satisficed and live within bounded rationality**

Researching IT as a social problem rather than just a technical problem

Key Issue – Contrast

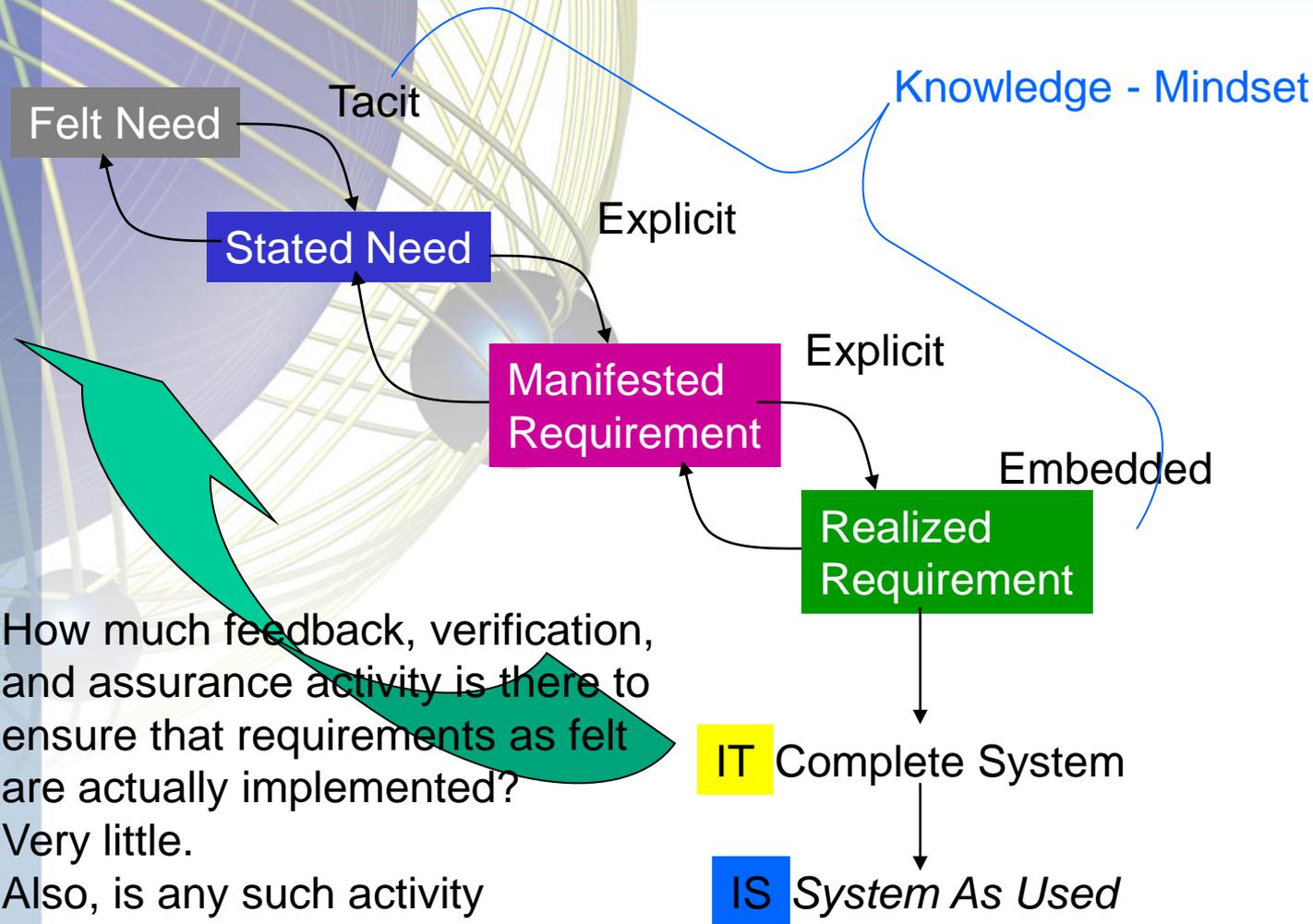
- **IT – the technology itself**
 - **Positivist**
 - **Natural science**
 - **Objective**
 - **Financial**
 - **Engineering/Computer Science**

- **IS – IT in an organizational context**
 - **Phenomenological**
 - **Social**
 - **Subjective**
 - **“Soft”**
 - **Management**
 - **Socially-constructed**
 - ***Emergent***

IT frequently (and systemically) fails because of IT/IS contrast

- **IT research should also focus on the socially constructed and emergent nature of IT as IS**
- **Structured dialog, (human felt needs AND technological aims), may improve the process of technology realization.**
- **Methodology**
 - **Analysis of IT/IS failure factors using case studies.**
 - **Pilot of a survey characterizes and tests elements of structured dialog through BPA (Business Process Analysis) tools**
- **Practical output: Conceptual decision framework**

Key Issue - Needs vs. Requirements



How much feedback, verification, and assurance activity is there to ensure that requirements as felt are actually implemented?

Very little.

Also, is any such activity structural? No.

Problems to Concept Model

Problems

Evaluation & Justification
Highly Subjective
Imperfectly Objective

Knowledge Domain
Mismatches

High (IT) Investment
Failure Rates
(45-90%)

Argument

IT investment is not working with right knowledge for evaluation and justification, and doesn't reflect human nature in decision-making.

Hypothesis

A structured dialog technique can improve the process of IT project evaluation and justification.

Research Concept Model:

- Understand Failure Factors
- Focus human teams on these factors
- Efficiently facilitate dialog to "set" knowledge
- Evaluate dialog outcome
- Evaluate program outcome

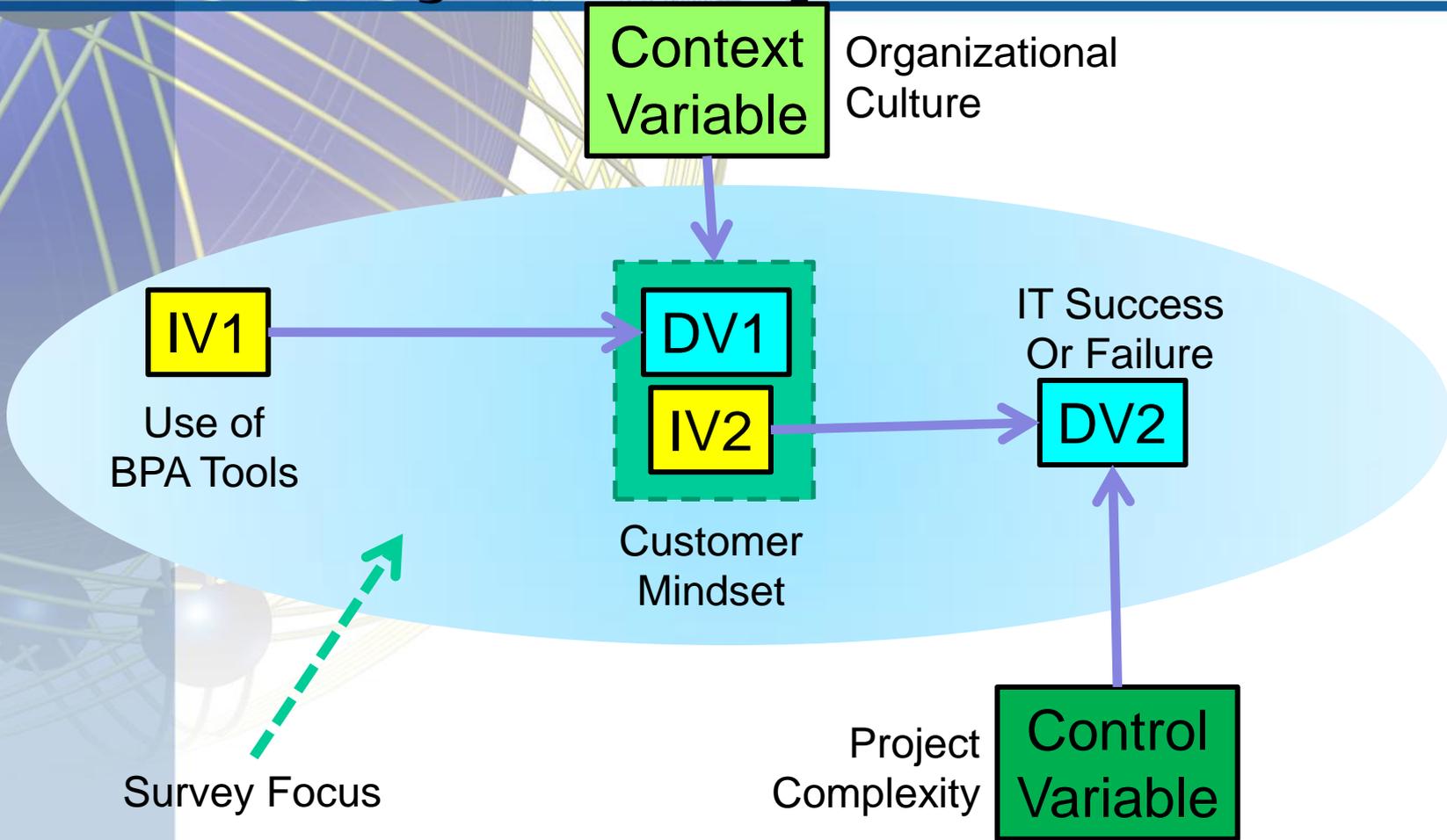
1. Case Studies (*hermeneutic circle*)
2. MetaInterpretation
3. Use and study tools to more systematically analyze IT requirements based on results of above (*positivist angle*)
4. Using survey, evaluate use of dialog to achieve shared meaning and concomitant impact on IT success/failure (*critical theory*) (Adorno, Habermas) (Giddens, Orlikowski, Foucault).
5. Finish *phenomenology* of IT wrt organizational epistemology through conceptual decision framework (Husserl, Heidegger, Ihde)

Probing: “Do what I mean, not (necessarily) what I tell you”

Case Study Findings: IT Failure – Core Root Causes & Relationship to Dialog

- **Inexperienced management – Direct**
- **Inexperienced technical staff – Direct**
- **Minimal quality control – Indirect**
- **Unstable requirements – Direct**
- **Less than 5% component reuse – Indirect**
- **Generalists only – Indirect**
- **Ineffective development technologies – No**
- **Manual estimating – Indirect**
- **Manual planning – Direct**
- **Informal progress tracking – Direct**
- **Inexperienced clients – Direct**
- **Inadequate tool suites – Indirect**

HPTi *Survey Conceptual Framework*



Results: Conceptual Decision Framework

- **Clarification of the purpose of the new IT/IS system**
- **Improving team dynamics to enable better collaboration**
- **The production and use of shared meaning**
- **Creation and use of common language**
- **Shared experience**
- **Repetition and continuation of dialog**
- **Making a specific shared decision**

- **Frame SE/SoSE Problem as Knowledge Management issue**
- **Recent research**
 - **KM in Technology Decision-Making**
 - **Conceptual Decision Framework for Improving Technology Success**
- **So What?**
 - **Model-Based SE Approach**
 - **Complex Systems Realization Framework**

- **Frame SE/SoSE Problem as Knowledge Management issue**
- **Recent research**
 - **KM in Technology Decision-Making**
 - **Conceptual Decision Framework for Improving Technology Success**
- **So What?**
 - **Model-Based SE Approach**
 - **Complex Systems Realization Framework**

Key SE Issues

- Lifecycle – current SE approach focused on specific *program* level
 - Need approach that transcends lifecycle
- System – evolution from *platform* to SoS
 - Need approach that handles SoS attributes – adaptable, flexible, adjustable, dynamically defined, interoperable, *emergent*
- Architecture artifacts are typically text-based or pictorially one-dimensional
 - Need approach that handles system complexity

Systems are now out-pacing ability of current SE to keep up

Key Factors

- **New acquisition model**
- **Architecture-driven SE approach**
- **Visualization of dynamic, multiple system dimensions in context**
- **Modular, component-oriented design to enable system portability, extensibility, and address dynamic requirements**
- **Need to involve multiple COI/COP during system lifecycle**
- **Need to enable system adaptability and flexibility – to a series of unknown and new requirements**

A Model-Driven Systems Engineering approach can address these factors simultaneously

■ Model:

- Pattern, plan, representation, description
- Shows main object, system, or concept and/or its workings
- A model is an approximation, representation, or idealization of selected aspects of the structure, behavior, operation, or other characteristics of a real-world process, concept, or system (IEEE 610.12-1990), i.e. an *abstraction*.
- A model usually offers different *views* in order to serve different purposes. A view is a representation of a system from the perspective of related concerns or issues (IEEE 1471-2000).

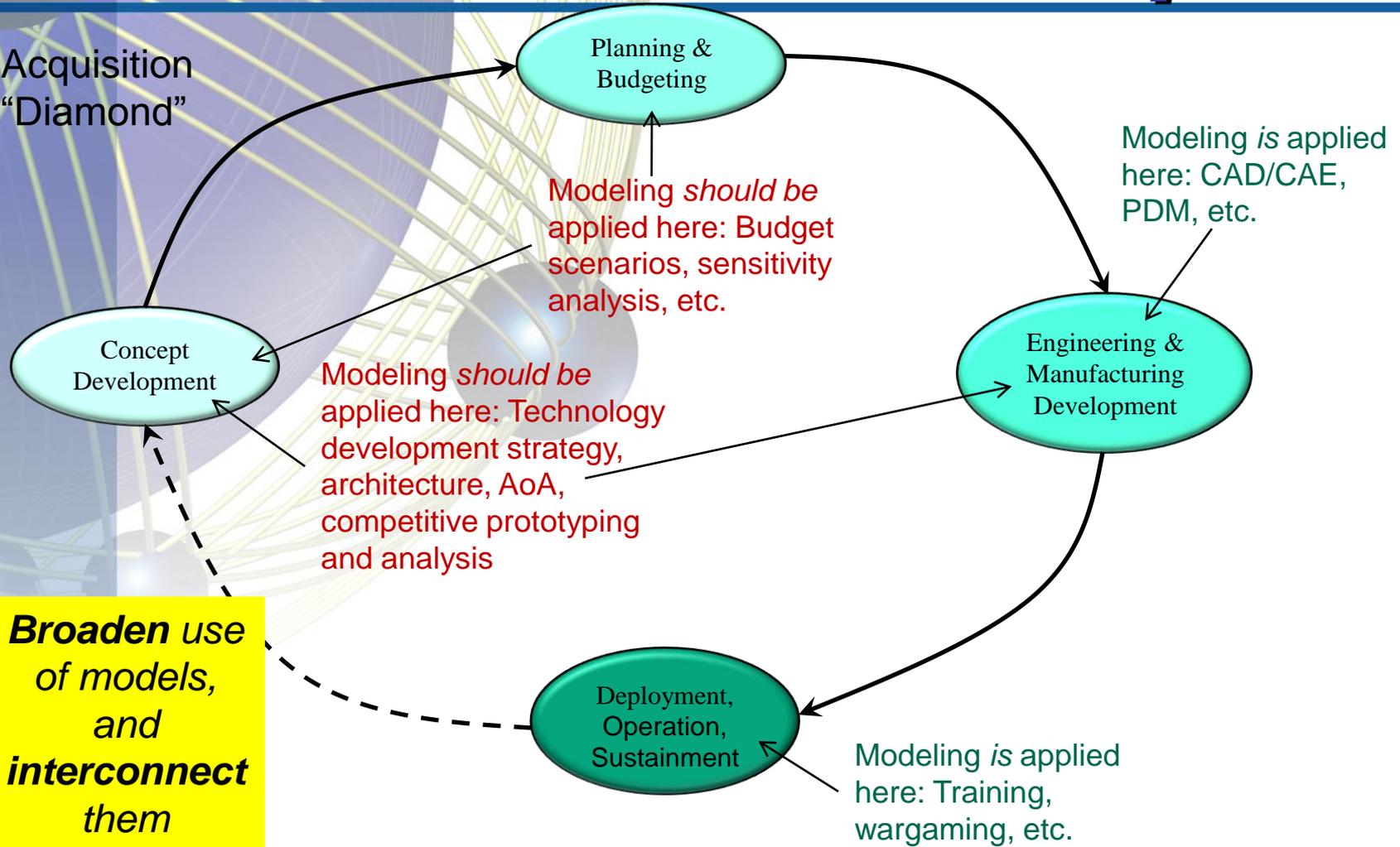
■ MBSE

- Formalized application of modeling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases (INCOSE-TP-2004-004-02, Version 2.03, September 2007)

Operationalizing the Definition

- **Model**
 - Isolation of dependent and independent variables from a system
 - Exploration of complexity and its implications
 - Representation of reality
- **Uses and Modes**
 - Facilitating discussion – focusing on several attributes rather than entire system
 - Systems of models – which should be connected
 - Metamethodology – a process (modeling) that enables the process of SE
 - Efficiency and Speed – reduces circular discussion and runarounds
 - Quality – reliability and repeatability of SE results

Acquisition
"Diamond"



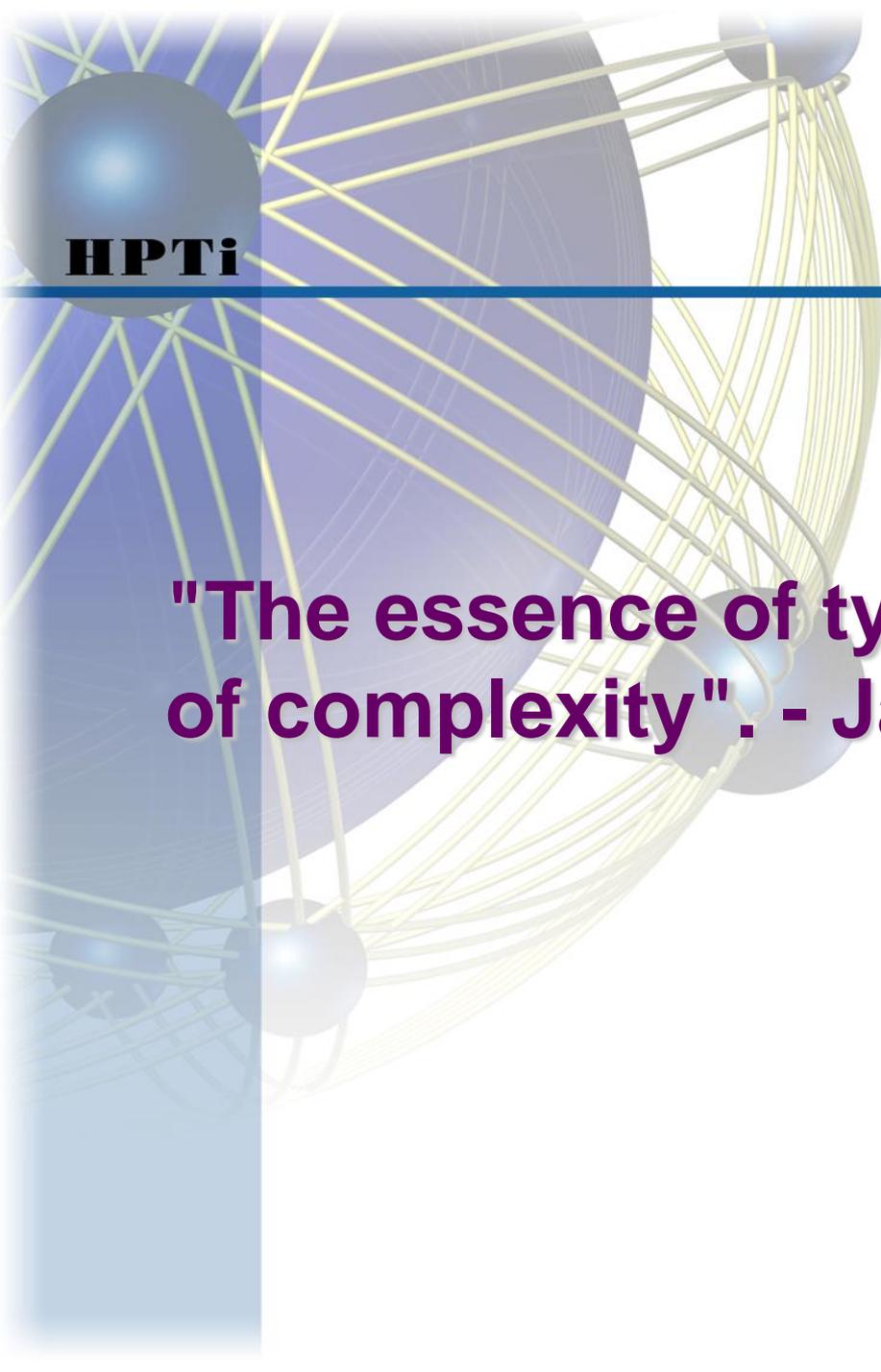
Broaden use of models, and interconnect them

Practical Impact – Solution Factors

- **Produce congruence across program**
- **Produce scenarios/vignettes for higher quality analysis across program**
- **Produce and use a common language across program**
- **Effective communications, within and outside of team**
- **Effective decision-making**

Implement 7 factors from conceptual decision framework

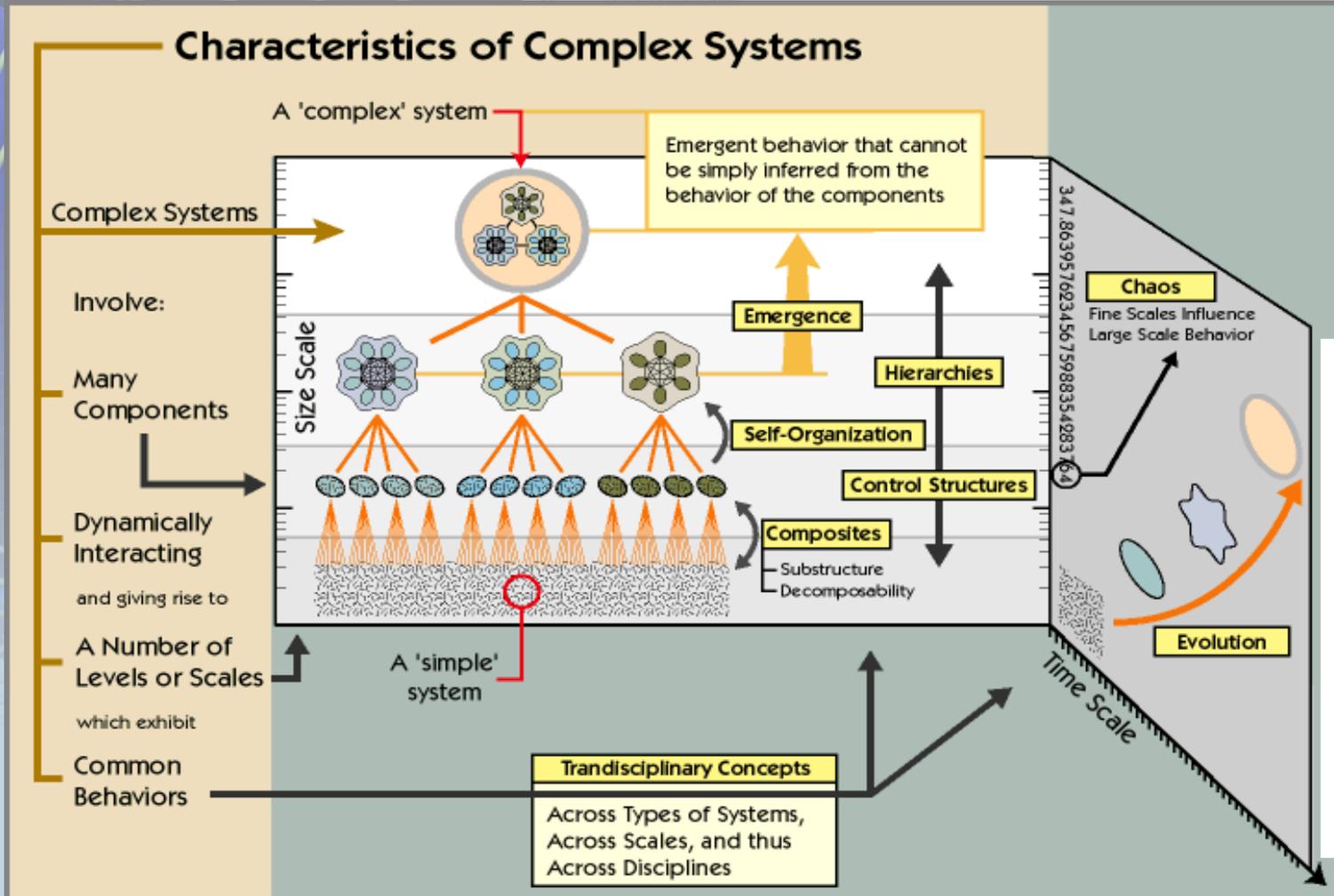
- **Frame SE/SoSE Problem as Knowledge Management issue**
- **Recent research**
 - **KM in Technology Decision-Making**
 - **Conceptual Decision Framework for Improving Technology Success**
- **So What?**
 - **Model-Based SE Approach**
 - **Complex Systems Realization Framework**



HPTi

"The essence of tyranny is the denial of complexity". - Jacob Burkhardt

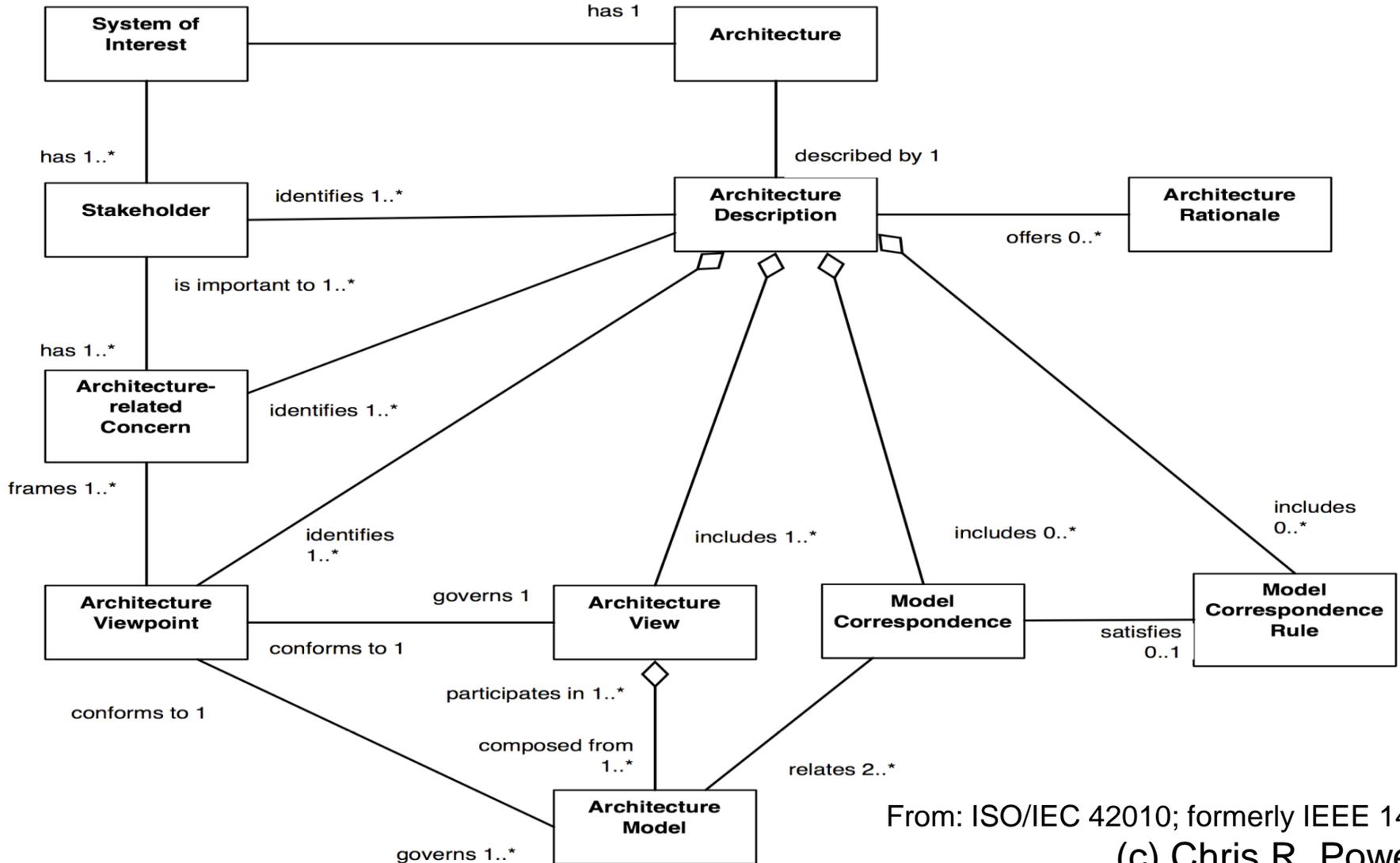
What is a Complex System?



Bar-Yam, Yaneer, "Engineering complex systems: multiscale analysis and evolutionary engineering," in Braha, Dan, Ali A. Minai, and Yaneer Bar-Yam. *Complex Engineered Systems*. Cambridge, Massachusetts: Springer, 2006 [Bar-Yam 2006]

- **Emergence**: Emergence is related to the dependence of the whole on parts, the interdependence of parts, and specialization of parts. While studying the parts in isolation does not work, the nature of complex systems can be probed by investigating how changes in one part affect the others, and the behavior of the whole.
- **Pattern formation**: simple mathematical models capture pattern formation such as local activation / long range inhibition.
- **Multiple (meta-) stable states**: Small displacements (perturbations) lead to recovery, and larger ones can lead to radical changes of properties. Dynamics do not average simply.
- **Multi-scale descriptions** are needed to understand complex systems. Fine scales influence large scale behavior.
- It is difficult but not impossible to answer the question "How complex is it?"
- **Behavior (response) complexity**: To describe the behavior of a system we try to describe the response function: actions as a function of the environment. However, unless simplifying assumptions are made, this requires an amount of information that grows exponentially with the complexity of the environment.
- **Contrasts**. Complex systems often exhibit contrasting characteristics, including simplicity and complexity, order and disorder, random and predictable behavior, repeating patterns and change
- We cannot predict what a complex system will evolve into.

Architecture & Modeling



Need for Architecture

HPTi

Mission Drivers

Potential Mission
Process Changes

Potential Mission
Performance Changes

Potential Mission
Data Changes

Potential Changes in
Collaboration or
Interoperability

Potential New or
Changed Mission
Capabilities

Operational
Impacts

Personnel
Impacts

Training
Impacts

Support
Impacts

Infrastructure
Impacts

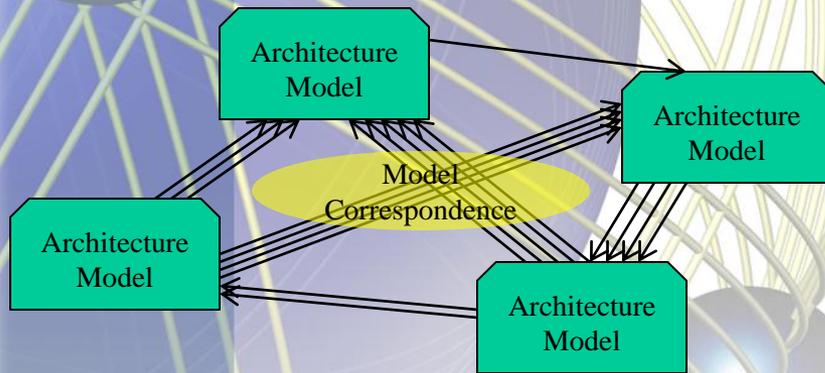
Related System
Impacts

Changes to
“what” gets done
Changes to “who”
is involved
Changes to “where”
things get done
Changes to “when”
things get done
Changes to “how”
things get done
Changes to “which”
information is used

*Architecture qualifies and quantifies impacts and implications of system changes incorporated and desired in response to changing mission drivers – social **and** technical*

Potential problem – increasing system complexity can make characterizing these difficult or impossible up front. Architecture and modeling ensure adequate capability to respond to “unknown unknowns”.

Architecture Modeling



- Rigorous Identification & Evaluation of Assumptions and Constraints
- Pursue Early & Explicit Coupling of System Attributes, Aspects, and Factors
- Identification of Metadata and Context
- Establish Mutual Team Understanding and Shared Decision Making
- Identify Impacts and Implications
- Establish Adaptability and Flexibility to Handle Unknown and Emergent Properties

- *Usage & Decision Modeling*
 - Concept Refinement Modeling
 - Operational Vignette Modeling
- *Technical Modeling*
 - Interface Specification & Modeling
 - Technology Modeling
 - Prototype Modeling
 - Emergent Properties Modeling
 - Testability Modeling
- *Program Modeling*
 - Requirements Modeling
 - Financial Scenarios & Sensitivity Analysis
 - EVM Criteria Modeling & Analysis
 - IMS/IMP Modeling
 - Risk Modeling & Prevention

These modeling processes characterize and explore a system in human-friendly and knowledge-surfacing terms which enable system success.

Re-envisioning SE/SoSE as KM

- **Maximize *efficient* involvement of OSD in programs to prevent risk**
 - **Leverage expertise and build more reachback**
 - **Tech transfer to Services**
 - **Continuous Engagement, Development Planning**
- **SE Research Agenda needed**
 - **Management & Technical concepts**
- **Manage programs as socio-technical systems**
- **Focus on knowledge discovery, creation, structure, and use**
- **Focus on human capital**
- **Change acquisition incentives to value contractor knowledge leverage – **pattern catalog****



HPTi

QUESTIONS?