Understanding and Modeling Interaction Effects in Complex Engineering Enterprises

Christopher Glazner
PhD Candidate, MIT Engineering Systems Division
2007 LAI Plenary Conference
Agenda

- Overview of the problem
- Hypotheses
- Enterprise-level interactions
- Useful modeling approaches
- Next Steps
Motivation

• Many **interactions** across the enterprise can give rise to **behavioral complexity**

• My experience: the Joint Strike Fighter Program
  • An engineering enterprise tasked to develop, produce and sustain an highly complex fighter that will become the mainstay of US and allied armed forces for 30 years
  • Evolving threat environment: the system’s requirements will inevitably change over its lifetime
  • Complex, large-scale interconnected open socio-technical (CLIOS)system
    • Many stakeholders, partners with differing value propositions
    • Open environment: encompasses political, regulatory, social domains

• An enterprise-level view is needed to understand and design complex engineering systems such as this
Current Approaches to Understanding Enterprises

Enterprise architecture frameworks are descriptive, not explanatory

- Several “views” of an enterprise are identified, such as organization, process, information, and strategy
- Each view is individually described according to a framework, such as Zachman or FEAF
- These views are static and describe the current state
- It has been tacitly assumed that the sum of these “views” can answer enterprise-level questions

Enterprise modeling is fundamentally reductionist, limited scope

- The enterprise is decomposed into “domains” (aka views)
- Each domain is separately modeled to answer problems relevant to that domain
- Unfortunately, many problems take place across domain boundaries:
  - IT architecture doesn’t reflect organizational realities
  - Strategic concerns drive partnering arrangements that affect the design process due to security constraints

There is a need for an approach to understand the dynamic interactions that effect behavior across the enterprise
“Views”: The Enterprise Architect's Flashlight

- Not linear
- Not independent
- Numerous interactions
Key Hypotheses

• The interactions that occur across the boundaries of the enterprise architecture views can give rise to emergent behavior (operational dynamics).

• This operational behavior is best modeled using hybrid, multi-scale modeling techniques.

• These modeling efforts can aid enterprise leaders in anticipating operational behavior attributable to the enterprise architecture and help them modify the architecture to suit their needs.
The Idea

The Enterprise

Enterprise Architecture Model

Frameworks:
- DODAF
- Zachman
- FEAF
- Etc..

Inputs

Strategy Policy Organization
Process Knowledge Information Product/Service

Outputs

Strategy Knowledge Process Information Product/Service

© 2004 Massachusetts Institute of Technology  Presenter/date - 7
Interactions in an Enterprise Architecture Framework

Policy / External Factors

Strategy -> Process

Process -> Organization

Organization -> Knowledge

Knowledge -> Information Technology

Information Technology -> Products / Services

Products / Services -> Strategy

© 2004 Massachusetts Institute of Technology  Presenter/date - 8
The Hybrid Modeling Approach

Organization View

Process View

Strategy View

Sub-model 1

Sub-model 2

Sub-model 3

Long-Term Strategy

(Mid-Term Strategy)

(Short-Term Strategy)

(Individuals)

Agent-based Model

Discrete Event Model

System Dynamics Model
The Merits of a Hybrid Modeling Approach

- Behavior at the scale of the enterprise is often emergent, arising from the interactions across the views and across scales.

- The behaviors within one of the enterprise views at a certain scale often fits neatly within one context, but these behaviors interact with others that do not. There is no one modeling approach that can capture all of these behaviors and their interaction appropriately. (Mingers and Gill, 1997)

- The idea is to use a modeling approach that is best tuned to each context/EA view (bottom-up, top-down, sequential, event-based, contingent-based, etc) for each sub-model.

- By linking together the multiple contexts/sub-models with feedback, complex behavior can emerge within the model, allowing a user to explore ways to affect changes to these behaviors through changes in the architecture.
The interactions/interdependendencies between the enterprise architecture views (cross-view interactions) is central to this work. I have informally termed these interactions within the model “hooks.” They fall into two broad categories:

- Variables: key enterprise metrics, agent population sizes, characteristics, DE process attributes
- Events: DE model events, changes in contingency fit, game outcomes

A hook has attributes that have meaning across the many contexts of the sub-models.

Practically, these hooks are pathways that allow models in different contexts to communicate and influence each other.
Modeling Environment

- I am using the program AnyLogic ([www.xjtek.com](http://www.xjtek.com)) for my simulation
  - Benefits:
    - Environment natively supports system dynamics, agent-based modeling, discrete event modeling, and Bayesian Network modeling
    - Flexible, open JAVA based environment makes it possible to insert custom code and models
    - Supports exportable applet-based animations and optimization of models based on specified parameters
- I may also use OrgCon in addition to model strategic and organizational contingencies within the model.
Next Steps

- Finish development of a hybrid model based on LAI’s Lean Enterprise Value Simulation;
- Develop a hybrid model of enterprise interactions based on a case study of enterprise change taken from the aerospace industry;
- Explore methods/approaches to validation and verification of enterprise models.
• Focus of work: developing a modeling approach to capture complex enterprise behaviors driven by high level interactions across the enterprise.

• The approach taken involves hybrid modeling (multi-methodology) at multiple scales, specific to a point change in the enterprise.

• The key to understanding such behavior is through analyzing the interactions and interdependencies across the many facets of the enterprise.
Questions?
Current Topics for thought…

• How simple can these models be before loosing the desired insight?
• How are such complex simulations verified?
• When is hybrid modeling the right way to go?
• In what context are local and enterprise decisions being made?
• What insight do local decision makers have over enterprise issues? What information is and is not available? What incentives drive local behavior?
• How can enterprise leaders determine the right levers and incentives across the enterprise?
• Can causality relationships be inferred?
A Concrete Example: 
Lean Enterprise Value Simulation

• A simulation of a complex aerospace enterprise
• Philosophy draws heavily on LAI research and the recent book *Lean Enterprise Value*
• Content and cases based on LAI member experience
• Created and distributed by Hugh McManus (hmcmcanus@alum.mit.edu) and Eric Rebentisch (erebenti@mit.edu)
A Tool to Teach Enterprise Transformation Skills

- An experience in using lean methods to analyze complex systems
  - Begin working at local levels to gain familiarity with techniques and understand benefits and challenges
  - Expand to enterprise to explore the issues of coordination, communication, analysis, and execution
  - Better understand enterprise change dynamics
  - Explore alternative implementation strategies and tradeoff benefits

- Repeated cycles of lecture, exercises, and implementation reinforce transformation process lessons
• Major enterprise components
• Flows (physical, knowledge, money) between them
• Key enterprise data and sources of inefficiency
• Can show enterprise wide (multi-program) flows
Strengths and Weaknesses of the LEV Model

• Strengths
  • Demonstrates emergent behavior from the flow of information and resources across contextual boundaries
  • Provides a proof of concept from some aspects of my proposed modeling
  • Simple; completely specified

• Weaknesses
  • Currently does not use an EA Framework as the basis of the model (currently process based)
  • As a result, there is little need for hybrid modeling