34th Annual INCOSE international symposium

hybrid event

Dublin, Ireland
July 2 - 6, 2024

—Fewer surprise failures and more “happy little accidents”

Dealing with Emergence in Model Based Engineering

Abstract

Steven C. Holt, Technical Fellow, Boeing
Model Based Engineering of Product, Production, and Sustainment Systems

- Emergence is what happens when the whole is more than the sum of the parts. It is both something we count on and something we fear in product development. When we can confidently design an airplane capable of doing things that none of its constituent elements can do alone this is called Weak Emergence. Our skill at creating intentional Weak Emergence is the basis for much of our technical success and many Systems Engineering decomposition and integration approaches. But sometimes we are surprised by things that we didn't expect. That is unintentional emergence. It may be good or bad; an opportunity or a problem. The extreme case, Strong Emergence, cannot be predicted in advance by analysis. It is not simply engineering error or insufficient training. Even error free models created by experts will not be able to predict Strong Emergence.

- As we move deeper into a world of Model Based Engineering and more and more models are connected in a common ecosystem, we face increasing risk that we will be unable to anticipate emergent behavior. This presentation will focus on the risks we face with emergence, how our current approach to Model Based Engineering alone may be inadequate to address the risks, and recommendations for what we can do to gain the benefits of emergence and avoid the downside failures.

Systems Engineering, Models and Emergence

- INCOSE Vision 2035 foresees increasing use of models of ever-increasing complexity and precision, including models that feed other models.
- The goal of Systems Engineering is based on being able to model emergence.
  - "SYSTEMS ENGINEERING AIMS TO ENSURE THE PIECES WORK TOGETHER TO ACHIEVE THE OBJECTIVES OF THE WHOLE."

INCOSE Vision 2035, page 8

Outline: Dealing with Emergence in Model Based Engineering

Spoiler alert:
The proposed solution is relatively simple and based on successful approaches to resolve similar issues in the past.

- Models in Model Based Engineering
  - Definition of Models
  - The necessity for models to be abstractions of reality
  - Appropriate detail in models based on use and context
- Emergence and Emergent Behavior
  - Definition of Emergence
  - The necessity for intentional emergence in Product Development and the risks of unintentional emergence
- The problem of increasing model detail and increasing unpredictable results
  - Choosing the right type of model and the appropriate abstractions to match the context
- Direction of a solution and recommendations
  - Summary of the type of high-level capabilities required.
  - Examples of near-term actions to be encouraged

What is a Model?

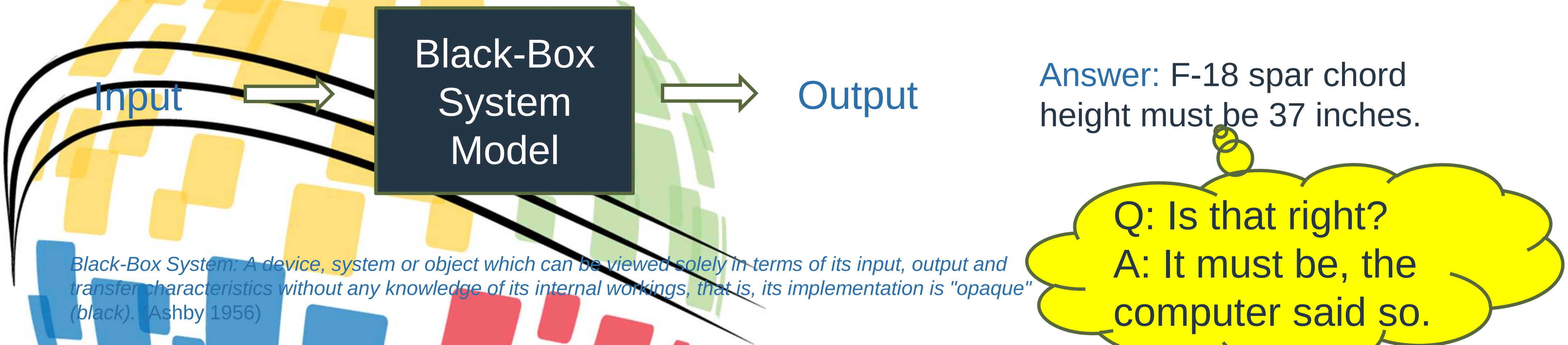
From the Systems Engineering Body of Knowledge Wiki. (Emphasis added)

1. *A physical, mathematical, or otherwise logical representation of a system, entity, phenomenon, or process.* (DoD 1998)
2. *A representation of one or more concepts that may be realized in the physical world.* (Friedenthal, Moore, Steiner 2009)
3. *A simplified representation of a system at some particular point in time or space intended to promote understanding of the real system.* (Bellinger 2004)
4. *An abstraction of a system, aimed at understanding, communicating, explaining, or designing aspects of interest of that system* (Dori 2002)
5. *A selective representation of some system whose form and content are chosen based on a specific set of concerns. The model is related to the system by an explicit or implicit mapping.* (Object Management Group 2010)

"All models are wrong, some are useful." George Box
"The map is not the territory, the [model] is not the thing it describes."
Alfred Korzybski (paraphrased)

The model may not be the thing, but…

- Vision 2035 anticipates models that very closely (or, exactly) match reality.
  - Digital Twin, Digital Systems Models
- History shows that “black box” engineering results are often accepted without question.
  - Spreadsheets and closed form calculations
  - Finite Element Models, Computational Fluid Dynamics Models
  - AI responses (LLM, etc.)



Black-Box System: A device, system or object which can be viewed solely in terms of its input, output and transfer characteristics without any knowledge of its internal workings, that is, its implementation is "opaque" (black). (Ashby 1956)

Another model

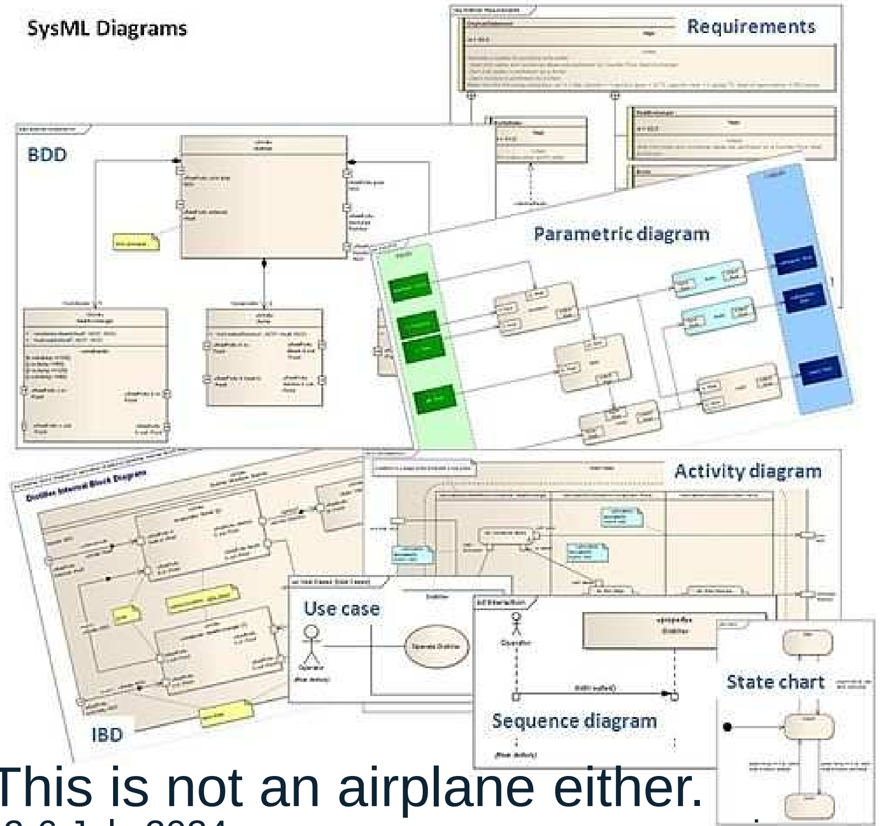
Rene Magritte: The Treachery of Images

https://collections.lacma.org/node/239578

Airplanes: The Treachery of Images

Another model in that theme

The Treachery of Models



This is not an airplane either.

What SysML models and what it doesn't.

SysML has 9 model types for modeling complex systems including 4 in the Behavioral Model category.

All 9 are deterministic models of what the architect <u>expects</u> to happen, prepared in advance. As such, they are a perception of what will happen. What actually happens may differ from the models.

Would a more detailed/accurate model help?

- The assumption is that a more detailed model will produce better answers.
  - As Sherlock Holmes said in “A Study in Scarlet”:
    - “‘From a drop of water,’ said the writer, ‘a logician could infer the possibility of an Atlantic or a Niagara without having seen or heard of one or the other. So all life is a great chain, the nature of which is known whenever we are shown a single link of it.’”
- Today we’d say—Maybe, but more likely in fiction than reality

The consequences of an overly detailed model

Quote from “Sylvie and Bruno: Concluded” by Lewis Carrol, 1893.

“That’s another thing we’ve learned from your Nation,” said Mein Herr, “map-making. But we’ve carried it much further than you. **What do you consider the largest map that would be really useful?”**

“About six inches to the mile.”

“Only six inches!” exclaimed Mein Herr. “We very soon got to six yards to the mile. Then we tried a hundred yards to the mile. And then came the grandest idea of all! **We actually made a map of the country, on the scale of a mile to the mile!**”

“Have you used it much?” I enquired.

“It has never been spread out, yet,” said Mein Herr: “**the farmers objected: they said it would cover the whole country, and shut out the sunlight! So we now use the country itself, as its own map, and I assure you it does nearly as well. "**

Cartographic Abstraction

- Map makers ran into this centuries ago.
  - To be fit for use a map (or model) must decide what to leave out and what to simplify
- Solution: Cartographic Abstraction
  - Intended use: What the target user wants to do
  - Simplification: Make it easy to make sense of the map
  - Exaggeration: Make it easy to focus on key points

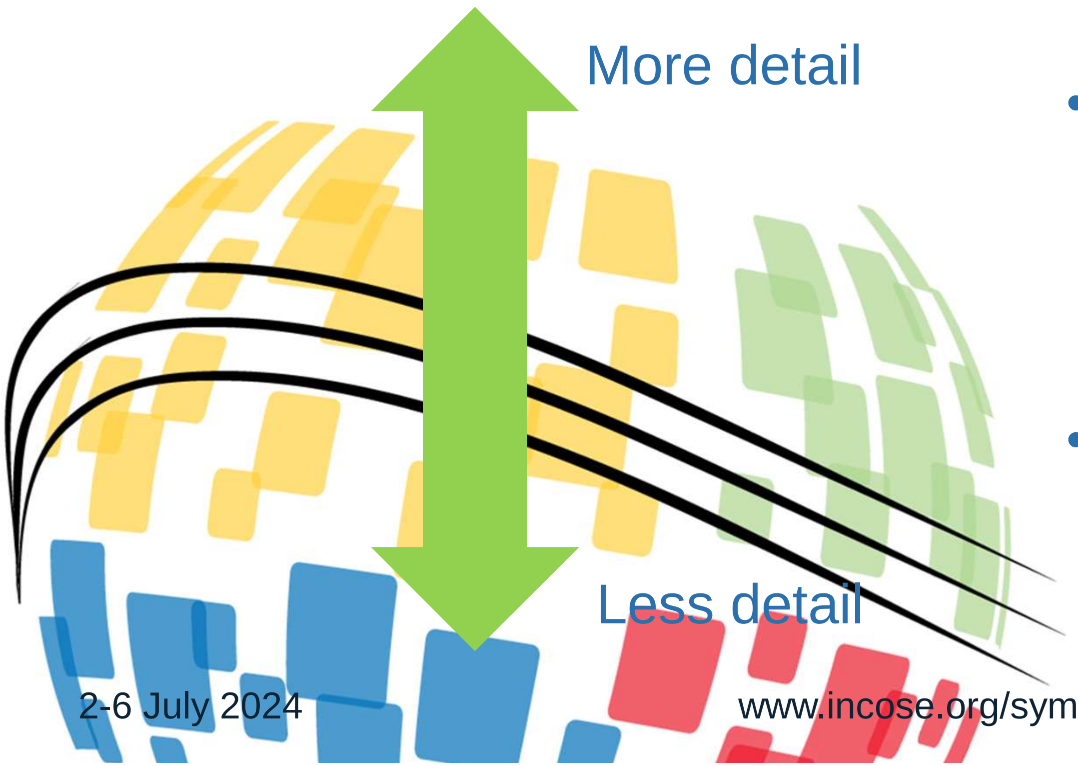
A Modeling Conflict: Abstraction

“Perhaps the most fundamental concept in systems modeling is abstraction, which concerns hiding unimportant details in order to focus on essential characteristics. Systems that are worth modeling have too many details for all of them to reasonably be modeled. Apart from the sheer size and structural complexity that a system may possess, a system may be behaviorally complex as well, with emergent properties, non-deterministic behavior, and other difficult-to-characterize properties. Consequently, **models must focus on a few vital characteristics in order to be computationally and intellectually tractable**.”

SEBoK wiki: Systems Engineering Body of Knowledge

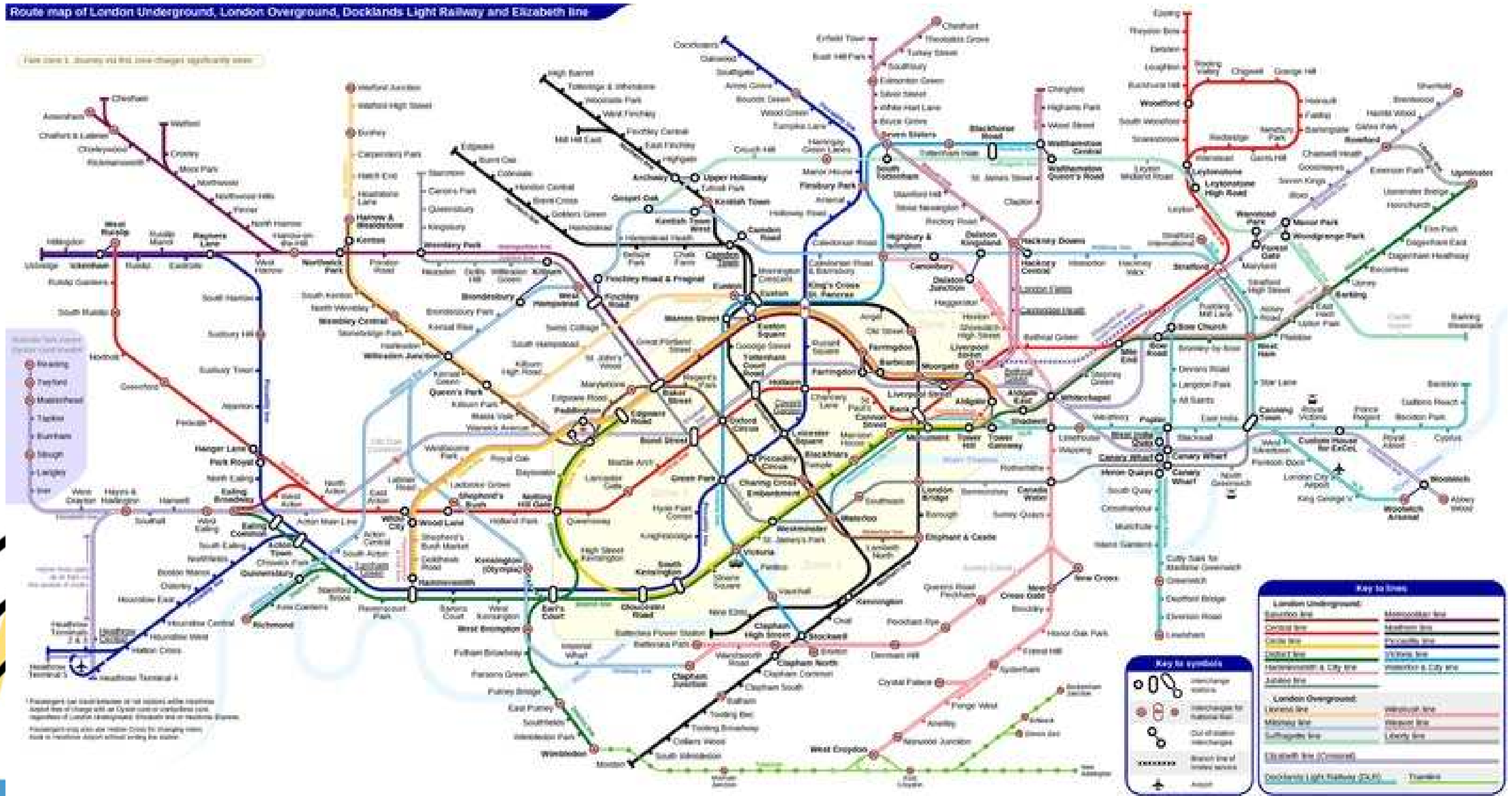
Conflict: More detail or less detail?

- It’s not a matter or one or the other. It depends on context.
- To be successful, a model must match its intended use.



- A model for design or analysis or prediction will (usually) benefit from added detail.
- A model for communication or exploration of ideas will (usually) benefit from simplicity and less detail

A masterful example



The London Tube Map: the World Standard

- Harry Beck, 1931
  - Electrical circuit drafter
- Users: transit riders.
- <u>Not geographically correct </u>and of limited value to operators, maintenance crews, etc.

Systems Engineering Models

- SysML and related models are great for detailed interface definitions. They are "less than perfect" as communication models
- Some models predict hurricane tracks but have no explanation of how the tracks form.
- Some models explain how plate tectonics work but have no predictive capability as to when earthquakes will happen.

- Key point: We need multiple models and multiple model types for multiple uses. Model diversity is valuable.
  – Necessary Condition: Being able to transfer data between models should be standardized and effortless

How can models be wrong?
(A non-exhaustive list)

Creating the model

- Wrong/inappropriate algorithms
- Modeler missed connections or dependencies
- Modeler made an error
- Modeler coded in their assumptions and/or cognitive biases
- Errors in learning data set used to create or check the model
- Reality does not/can not match architecture model

Running the model

- Inappropriate model type for the problem
- Missing input
- Incorrect input
- Wrong seed prompt
- Misinterpreted output
- Accepted output without question
- Accepting the model as The Way.

Understanding the results

- Emergence and emergent behavior that is unpredictable

If we have reason to question the output, we may find these by closer observation.

What is this? How would we know it happened?

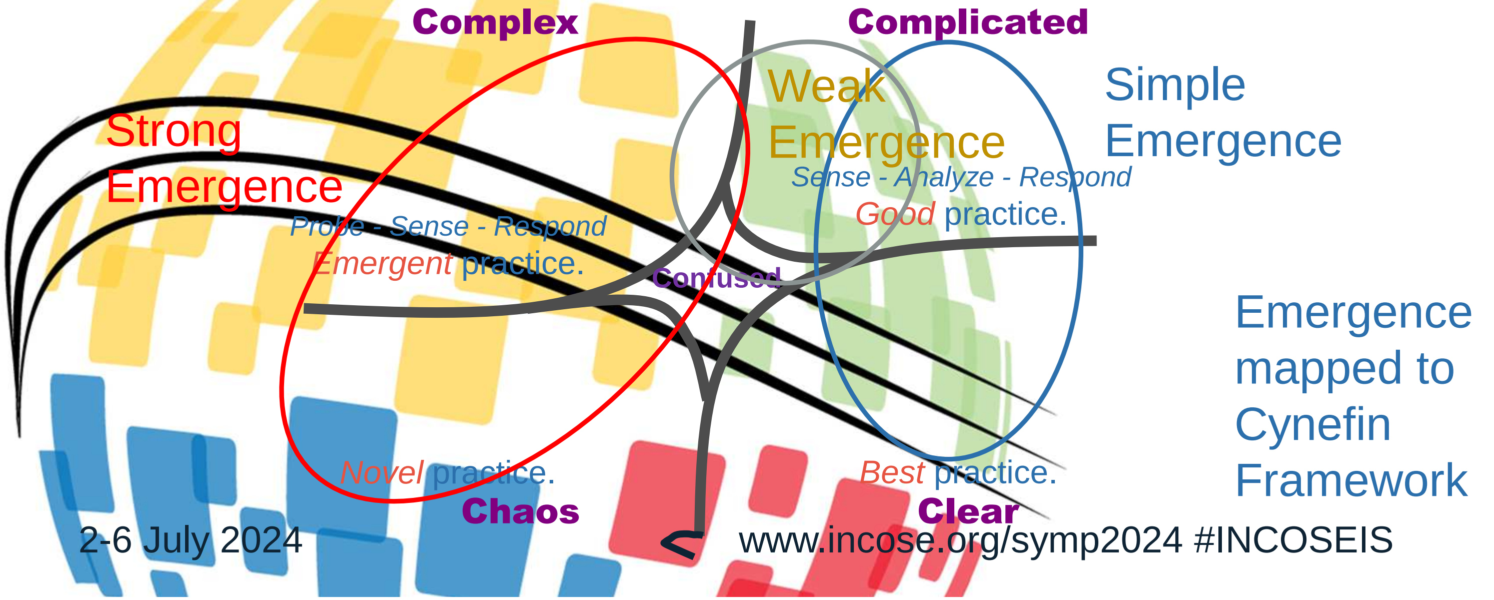
What is Emergence?

- Emergence is “the principle that entities exhibit properties which are meaningful only when attributed to the whole, not to its parts.” (Peter Checkland, “Systems Thinking, Systems Practice” 1981)
- Emergence can be either **intentional** or **unintentional**.
- Three forms of Emergence (Dr. Scott Page)
  - **Simple Emergence**: Performance of combinations of parts in “ordered” systems that can be intentionally designed for. We design an aircraft and can predict much of its performance by analysis.
  - **Weak Emergence**: Performance of combinations of parts in “complex” systems that is both desirable and expected but cannot be fully predicted in advance. For a new airplane, we don’t know full flight characteristics until tested.
  - **Strong Emergence**: Properties and behavior that is not expected, not predicted and **cannot have been predicted**. It is evident only after the product has been built and tested.
- System of Systems are particularly at risk of Strong Emergence, which is frequently associated with failure

Ref: SEBoK on Emergence https://sebokwiki.org/wiki/Emergence

Is Emergence good or bad?

- When we design a product to meet requirements, we are using intentional Simple/Weak Emergence based on our experience, expertise, proven models, and engineering judgment.
- Strong Emergence is unpredictable and unintentional and usually only evident in hindsight.
  - Strong emergence nearly always comes across as a failure, often a catastrophic one.

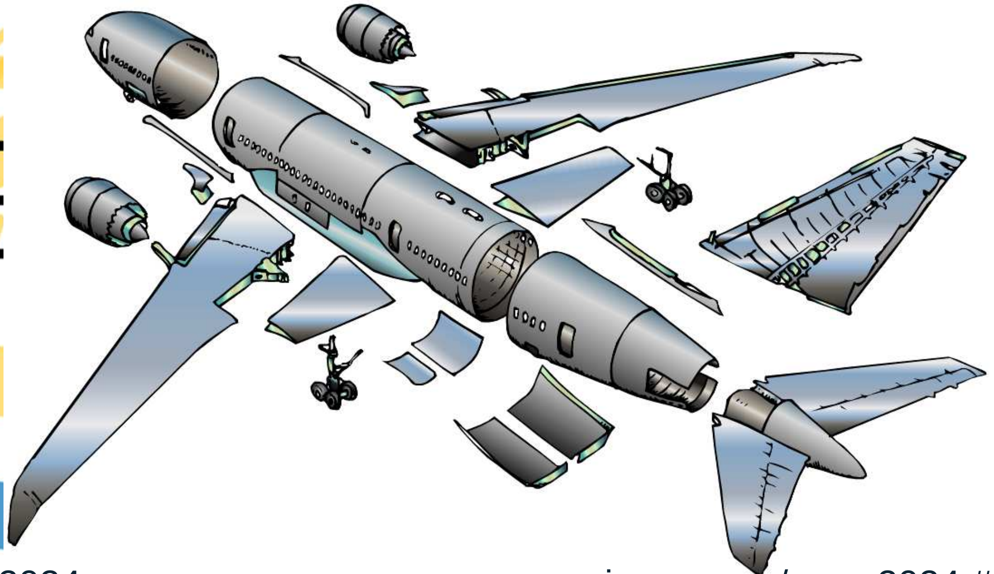


Whether Emergent Behavior is good or bad can depend on choosing the right model and the context

Emergence is a function of integration…or lack of it

Emergence only exists because of integration.

- An airplane has performance capabilities that none of its parts have individually.
- For successful intentional emergence, integration during design is critically important
- For successful response to unintentional emergence, integration in test and/or models is critically important for catching the emergent behavior as early as possible.



Unintentional emergence can be a good thing. An opportunity. If we recognize and can act on it.

Negative Strong Emergence: F-15 Vertical Stabilizer fatigue

F-18 photo used due to smoke trail

Dryden Flight Research Center EC89-0096-206 Photographed 1989
F-18 HARV smoke and tuft flow visualization. Angle of Attack = 20 deg. NASA photo.

- The F-15 was developed using experience on high-speed fighter design going back decades. Best Practices and expert designers.
- Fatigue cracks showed up in the vertical stabilizers soon after it went into service.
- F-15 had a higher Thrust to Weight Ratio than previous planes and could operate at higher Angle of Attack.
- High AoA resulted in vortices hitting Vertical Stabilizer and much higher loads than anticipated.

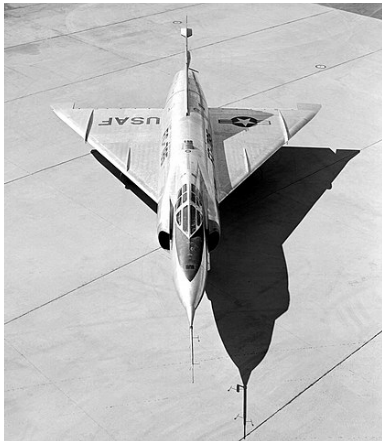
Positive Strong Emergence: The Lürssen Effect

Lt. J.E. Russell, Royal Navy official photographer, Public domain, via Wikimedia Commons

- World War 2 German torpedo boat made by Lürssen Shipyards
- Faster, longer range and more maneuverable than US and UK equivalents—due to emergence
- Prototype had rounded bottom hull to cope with waves. Testing showed that at speed the rudder lost effectivity. Added two outrigger rudders.
- Discovered by accident that the outrigger rudders significantly increased top speed.
  - Capable of 48 knot sprints and 43.5 knot cruise.
  - Performance was so superior that many served in other navies until 1965.

Can an MBSE model identify emergence? It depends.

- A model might catch things that people miss, but only for Weak Emergence.
  - The F-102 Delta Dagger/Deuce: Designed to be supersonic using current best approach but without an area rule fuselage couldn't hit Mach 1. (The redesigned F-106 used area ruled fuselage.)
  - F-15: Designed to well established process but assumed operation would be similar
- Investigate weak signals and anomalies. Avoid dismissing them as outliers.



WARNING: In hindsight nearly all Strong Emergence will be interpreted as due to insufficient detail or skill or oversight. This is rarely the case. Strong Emergence cannot be determined in advance by analysis alone.

A modeling conflict

- The more detailed our models are (higher fidelity) the greater their ability to model at the system level, but also the greater the chance that no one understands all the details and connections and dependencies.
- That means an increasing chance of unpredictable Strong Emergence.
- The simpler our models are the easier they are to understand and communicate, but they may be less accurate at the analysis and synthesis of product development.

How can we have both the detail we need to design and the simplicity we need to understand?

Direction of a Solution: Identify emergence early

- Have and use multiple, diverse models
  - Different variables, different methods, different fidelity, including physical mockups
  - Hone people's skills at "back of the envelope" gut checks
    - TLAR: That Looks About Right
    - "That's weird."
    - Heuristics
- Seek out conflicts between models
  - Differences are weak signals
  - Treat differences like near misses in Safety--investigate

We can learn from slide rules

https://commons.wikimedia.org/wiki/File:IBM_150_Extra_Engineers,_1951.jpg

- Slide rules are physical models for calculations.
- They don't tell you where the decimal point goes.
- Engineers used rough, order of magnitude, calculations to locate the decimal point.
- This built Engineering Judgment based on experience and a grounding in fundamental principles.

Time passed and calculators and computers appeared.

- Experienced engineers still used their rough calcs and slide rules to check their results…and those of less experienced engineers.

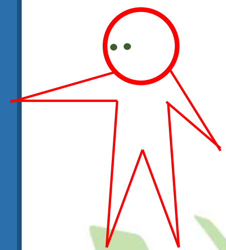
Variations of that theme repeated with new technologies

- Finite Element Models
- Computational Fluid Dynamic Models
- Many other types of calculations.

Some ways to improve Engineering Judgment

Drawing, sketching, writing by hand, making a mockup, speaking aloud, and using counter factuals all activate different neural pathways than reading or using a computer.

These are alternative models with diverse methods.



- **Make assumptions visible**
  - Write down the assumptions behind the model---by hand
  - Draw the control volume, system boundaries, Free Body Diagram—by hand
  - Sketch out how you think one variable will impact another—by hand
  - Sketch out ideas—by hand
  - Make a physical mockup—by hand (foam core, blocks, clay, etc.)
- **Review what you're doing**
  - Explain it to someone else
  - Read it out loud, even if just to yourself
  - Document what you are NOT including
- **Look for conflicts and weak signals**
  - Mentally simulate what you think will happen
  - Use counter-factuals: scenarios that are unlikely, but still…
  - Use Applied Critical Thinking tools like Pre-Mortem
- **Evaluate the results**
  - **"What would it take for me to change my mind?"**
  - Always predict outcome in advance and compare the actual results to prediction, Study and learn from differences.
  - "It's the predictions you make before you run the test that count."

Being successful and avoiding failure

- There are two ways to fail during the implementation of new technology:
  - Not making the changes necessary to gain the benefit of the new technology
  - Abandoning existing policies, procedures and metrics that should be retained to enhance the new technology
- As promising and valuable as Model Based Engineering and AIs are, they are not sufficient alone.
  - Models based on reference environments and training sets such as a Digital Twin or LLMs cannot predict Strong Emergence because they can't model what they don't know (uncertainty).
- To successfully navigate an uncertain future, we must retain and enhance our Engineering Judgment skills
  - Heuristics and Rules of Thumb
  - First Principles
  - If you understand the underlying theory, you can modify the details to fit the context.
- We should encourage the retention of sufficient "old school" methods as alternate means to compare to the results of new technology methods and tools.
- That will increase our chances of identifying emergent behavior early enough to take advantage of fleeting opportunities and to avoid impending risks.

The Bottom Line

- The benefits of models for decreasing engineering workload and increasing quality are immense.
- Complex models increase the risk of unpredictable Emergent Behavior.
- Develop Engineering Judgment skills to help capture the gains and avoid the losses.

References and Influences

- Alexander, Christopher. "A City is Not a Tree." Architectural Forum, 1965.
- Alexander, Christopher. "The Timeless Way of Building" Oxford University Press, 1979.
- Ashby, W. Ross. "Introduction to Cybernetics" Wiley, 1956.
- Carroll, Lewis. "Sylvie and Bruno: Concluded" 1893. Public Domain
- Checkland, Peter. "Systems Thinking, Systems Practice" John Wiley and Sons, 1981
- Corning, Peter A. "The Re-Emergence of "Emergence": A Venerable Concept in Search of a Theory", *Complexity*, 7 (6): 18–30, 2002.
- Crawley, Ed; Cameron, Bruce; Selva, Dan. "Systems Architecture: Strategy and Product Development for Complex Systems" 2016.
- Garland, Ken. "Mr Beck's Underground Map" Capital Transport Publishing, 1994.
- Grant, Adam. "Think Again: The Power of Knowing What You Don't Know" Viking, 2021.
- Griffin, Michael. "How Do We Fix System Engineering?" 61st Intl Aeronautical Congress, IAD-10.D1.5.4, 2010.
- Holman, Peggy. "Engaging Emergence: Turning Upheaval into Opportunity" Barrett-Koehler, 2010.
- Kelly, Tom. "The Art of Innovation." Broadway Business, 2001.
- Page, Scott. "The Model Thinker: What you need to know to make data work for you" Basic Books, 2018.
- Page, Scott. "How 'Many Model' Thinkers Make Better Decisions" Harvard Business Review, 11/2018.
- Pendleton-Jullian, Ann; Brown, John Seely. "Design Unbound V 1: Designing for Emergence in a White Water World" MIT Press, 2018
- Saint, Andres. "Architect and Engineer: A Study in Sibling Rivalry" Yale University Press, 2008.
- SEBoK Editorial Board. 2022. *The Guide to the Systems Engineering Body of Knowledge (SEBoK)*, v. 2.7, R.J. Cloutier (Editor in Chief). Hoboken, NJ

34th Annual INCOSE
international symposium

hybrid event

Dublin, Ireland
July 2 - 6, 2024

www.incose.org/symp2024
#INCOSEIS