Welcome to the Webinar – if you have not dialed into the audio portion, please do so now. In the U.S., dial 1-719-457-6209, passcode **733 424 0539**. If you are outside the U.S., the webinar invitation includes a list of toll-free numbers for different countries. Please note that your phone is automatically on mute when you join to avoid disturbing other participants.

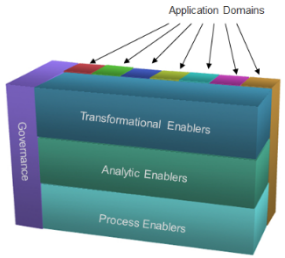
INCOSE Webinar Series

Wednesday 19th February 2020 – Webinar 134

21st Century Systems Engineering

Steven Dam

INCOSE is offering Webinars…



- To provide a forum for experts in the field of Systems Engineering to present information on the "State of the Art"
- To explain how INCOSE works, and how to make the most out of INCOSE membership

INCOSE Systems Engineering Professional PDU Credit

Please note that you can claim 1PDU credit towards your Systems Engineering Professional re-certification by attending this webinar. INCOSE webinars may also apply to the PDU requirements of other organizations, depending on the subject matter

To qualify, you must have attended through at least 75% of the webinar for webinars that last less than one hour, or through 45 minutes of the webinar for webinars that last for 1 hour or longer.

Here is the link to details about certification renewal, including information on PDUs.
http://www.incose.org/certification/CertProcess/CertRenew

Choreography

1. Andy Pickard (your host) will introduce the Webinar and the speaker
2. Steve will speak for about 40 to 45 minutes
3. During his talk, participants can write questions using the Webex Q&A window
4. After Steve completes his talk, he will spend 10 minutes answering questions that Andy selects from those submitted by the audience
5. Andy Pickard will provide information about upcoming Webinars and then end this session
6. This Webinar is being recorded and will be made available on the INCOSE website to members and employees of CAB organizations

SE that’s faster, better, and cheaper

21st Century Systems Engineering

Overview

- INCOSE's FuSE Initiative
- Why Is SE Today Too Slow?
- What's Needed for 21st Century SE?
- The Future of Systems Engineering is Here

INCOSE's FuSE Initiative

- Future of Systems Engineering (FuSE)
- The Purpose of FuSE is to:
  - "Position systems engineering to leverage new

Owner: Shortell

What will good look when we have used FUSE to deliver systems?

Methods, Processes, Techniques for self-learning systems (including process changes and handling V&V)
Improved simulations to handle dynamic objectives
Architecture techniques for AI heavy systems
Demostrate how AI positively improves a system while considering -ilities (i.e. safety/security) within acceptable bounds

What is stopping us from doing this now?

Data availability/usage (OCI and IP concerns)
Knowledge and research
Assurance, Trust, Understanding of the technologies
SE too slow to keep up with AI advancements

Source: Owner Shortell, FuSE (Future of Systems Engineering) Town Hall. INCOSE 2019 International Symposium, Torrance, CA, US, 28 January 2019

Why Is SE Today Too Slow?

How Have Our Languages Evolved Over the Last 60 Years?

- 1960s – used flow charting techniques derived from software (SREM created for software and systems engineering)
- 1970 –Data Flow Diagramming – heavily influenced by software development
- 1980s - IDEF, State Machine modeling and Computer-Aided Systems Engineering tools (e.g., RDD-100)
- 1990s – eFFBDs and Object-Oriented Analysis and Design/UML - derived from software techniques
- 2000s – SysML: a profile on UML
- 2010s – Still SysML, but LML emerged derived from systems engineering techniques; LML version 1.1 included an ontology so that systems engineering can be performed at the system entity level, instead of at the diagram level

Why do we always seem to be 10 years behind the software world?

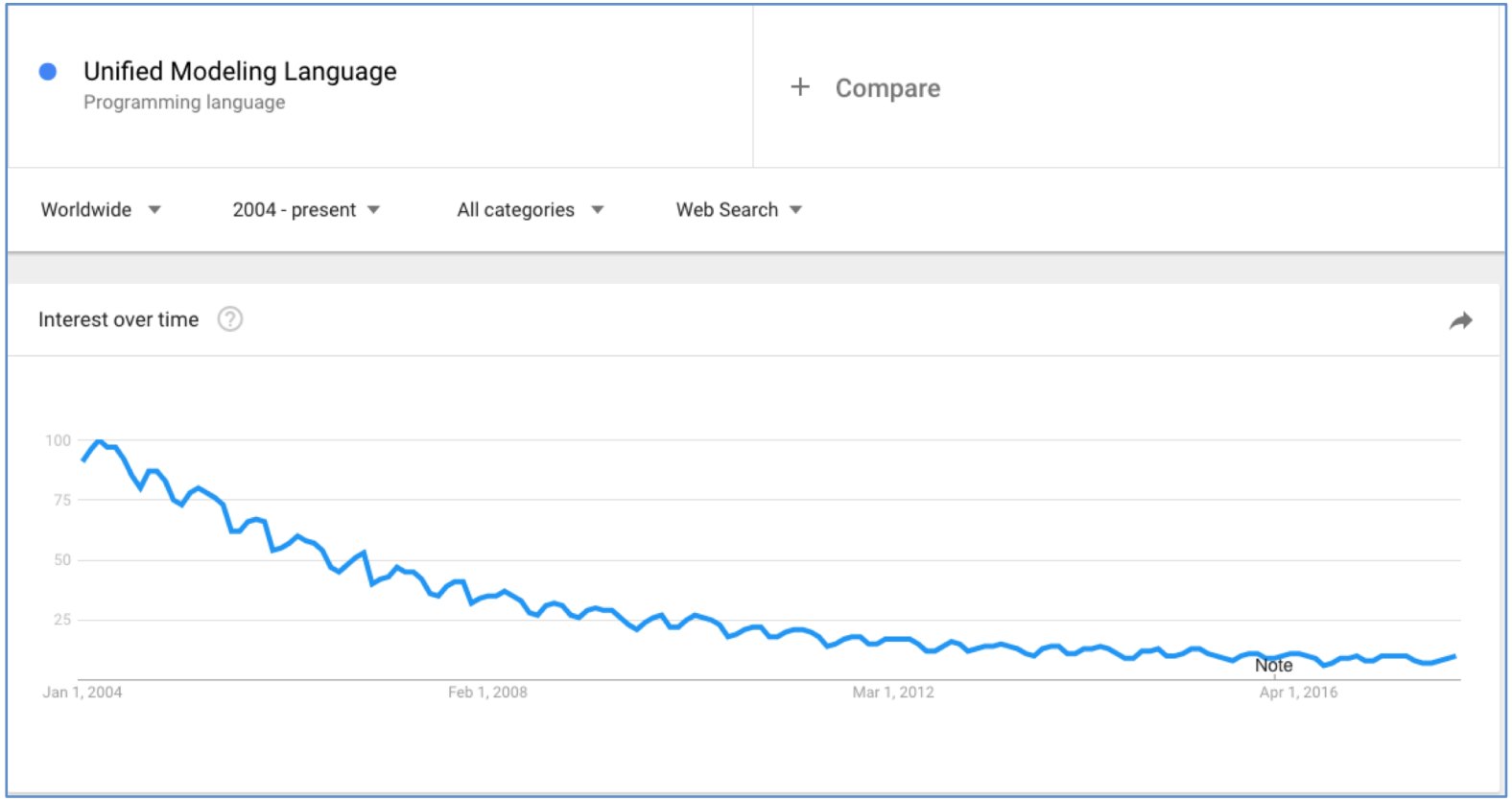
What Do We Need that We Are Missing?

- Need methods to capture and visualize tremendous amounts of information
- Massive storage and retrieval of information
- Need not only all the technical readouts, but also the programmatic information
- Capability to move data around easily, between applications
- A language that enables decomposition and abstraction
  - A systems engineering language, not a software engineering language
  - A language that is simple so that systems engineering can easily use it

But I know you are saying SysML does all this right?

Why Isn't SysML Enough?

- Systems Modeling Language was developed to extend the software focused Unified Modeling Language (UML) to systems
- Interest in UML peaked in 2004
- Software developers have moved on to Agile, which requires *functional requirements*
- Both SysML and UML require experts to create and interpret
- Systems Engineering requires communications with *all* stakeholders



From Google Trends retrieved 11/17/2017

If you have to be an expert in SysML's lexicon and diagram specifications, who are you communicating with?

Why Isn't SysML Enough?

- But it's worse than just not being easy to understand
- SysML is lacking many of the programmatic pieces of information: risk, issues, decisions, schedule, cost, … as explicit diagrams or entities
- The lack of an ontology and the need for more diagrams has been noted and is in the process of being developed, but its going to be another 3 years (at least)
- But what if there was already a language that provided an ontology for SysML and filled in the missing pieces?

But don't we need more than just a language for 21st Century SE?

What's Needed for 21st Century SE?

What's Needed for 21st Century SE?

- Techniques
- Processes
- Tools

You can use any methodology with these 3!

This methodology isn't the only way to have a successful MBSE capability, but all three elements must be incorporated in any methodology you use

My Methodology

- Technique - LML
- Processes – Middle Out Process
- Tools – A modern real MBSE tool
  - Most systems engineers still use Microsoft Office or model drawing tools. These are severely lacking in the 21st century features we need

Let's start with the language – LML

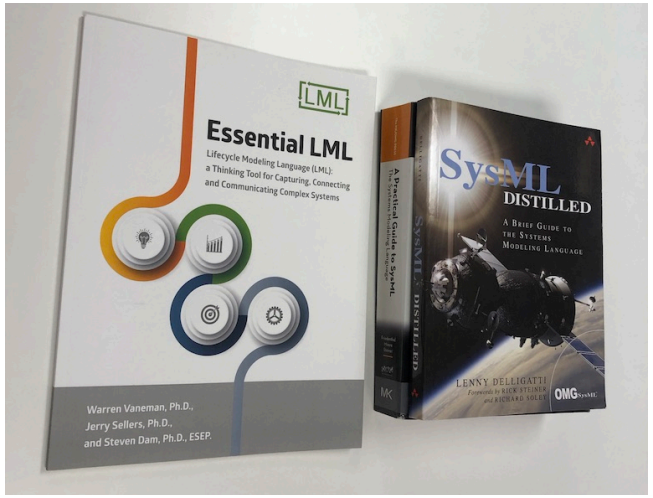
LML – An Open Standard

See it at www.lifecyclemodeling.org

LML's Simplified Taxonomy

- Action
- Artifact
- Asset
  - Resource
- Characteristic
  - Measure
- Connection
  - Conduit
  - Logical
- Cost
- Decision
- Input/Output
- Location
  - Physical, Orbital, Virtual
- Risk
- Statement
  - Requirement
- Time

Supports capturing information throughout the lifecycle



Essential LML: Lifecycle Modeling Language – A Thinking Tool for Capturing, Connecting, and Communicating Complex Systems

LML is a non-proprietary standard!

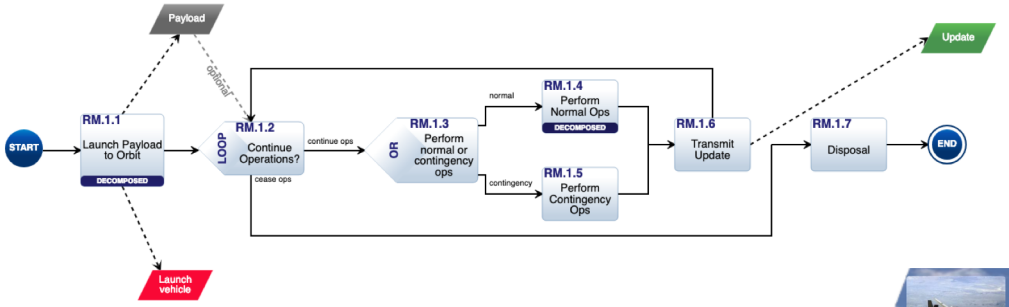
All Related to Each Other

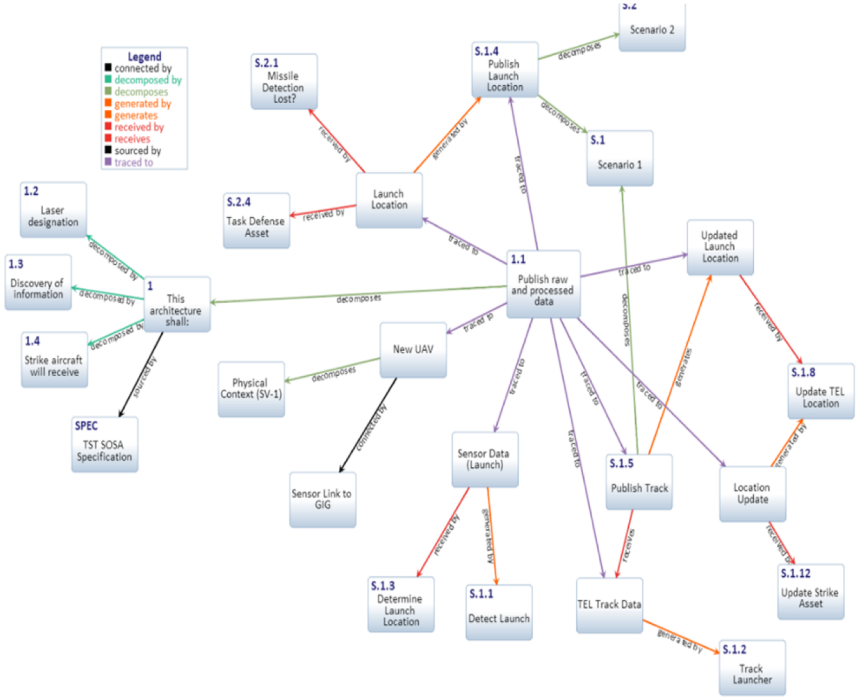
- Automatically bi-direction to enable traceability up and down the classes
- Common verbs for relationships in each direction
- Same verbs for parent-child (decomposes/ decomposed by) and peer-to-peer (relates/related to) relationships

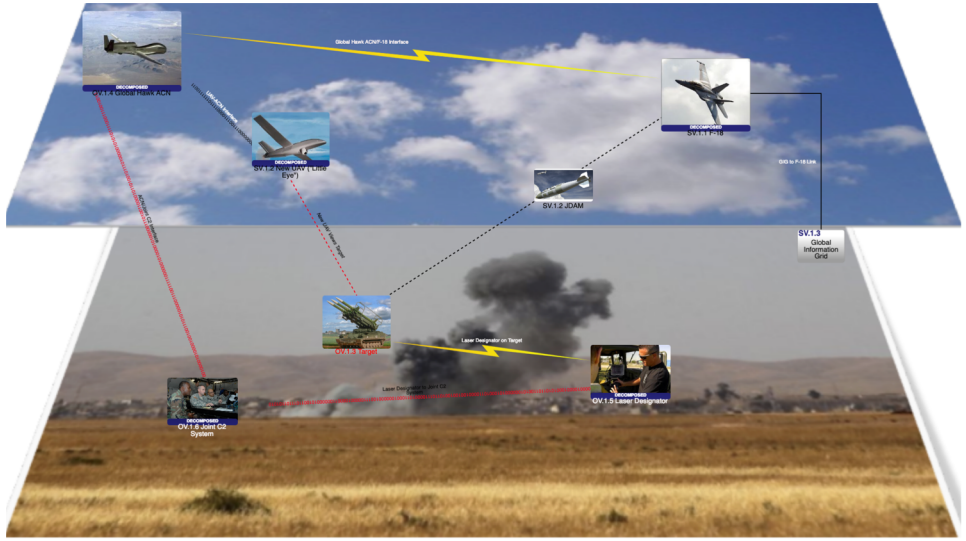
	Action	Artifact	Asset (Resource)	Characteristic (Measure)	Connection (Conduit, Logical)	Cost	Decision	Input/Output	Location (Orbital, Physical, Virtual)	Risk	Statement (Requirement)	Time
Action	decomposed by* related to*	references	(consumes) performed by (produces) (seizes)	specified by	-	incurs	enables results in	generates receives	located at	causes mitigates resolves	(satisfies) traced from (verifies)	occurs
Artifact	referenced by	decomposed by* related to*	referenced by	referenced by specified by	defines protocol for referenced by	incurs referenced by	enables referenced by results in	referenced by	located at	causes mitigates referenced by resolves	referenced by (satisfies) source of traced from (verifies)	occurs
Asset (Resource)	(consumed by) performs (produced by) (seized by)	references	decomposed by* orbited by* related to*	specified by	connected by	incurs	enables made responds to results in	-	located at	causes mitigates resolves	(satisfies) traced from (verifies)	occurs
Characteristic (Measure)	specifies	references specifies	specifies	decomposed by* related to* specified by*	specifies	incurs specifies	enables results in specifies	specifies	located at specifies	causes mitigates resolves specifies	(satisfies) spacifies traced from (verifies)	occurs specifies
Connection (Conduit, Logical)	-	defined protocol by references	connects to	specified by	decomposed by* joined by* related to*	incurs	enables results in	transfers	located at	causes mitigates resolves	(satisfies) traced from (verifies)	occurs
Cost	incurred by	incurred by references	incurred by	incurred by specified by	incurred by	decomposed by* related to*	enables incurred by results in	incurred by	located at	causes incurred by mitigates resolves	incurred by (satisfies) traced from (verifies)	occurs
Decision	enabled by result of	enabled by references result of	enabled by made by responded by result of	enabled by result of specified by	enabled by result of	enabled by incurs result of	decomposed by* related to*	enabled by result of	located at	causes enabled by mitigated by result of resolves	alternative enabled by traced from result of	date resolved by decision due occurs
Input/Output	generated by received by	references	-	specified by	transferred by	incurs	enables results in	decomposed by* related to*	located at	causes mitigates resolves	(satisfies) traced from (verifies)	occurs
Location (Orbital, Physical, Logical)	locates	locates	locates	locates specified by	locates	locates	locates	locates	decomposed by* related to*	locates mitigates	locates (satisfies) traced from (verifies)	occurs
Risk	caused by mitigated by resolved by	caused by mitigated by references resolved by	caused by mitigated by resolved by	caused by mitigated by resolved by specified by	caused by mitigated by resolved by	caused by incurs mitigated by resolved by	caused by enables mitigated by results in resolved by	caused by mitigated by resolved by	located at mitigated by	caused by* decomposed by* related to* resolved by*	caused by mitigated by resolved by	occurs mitigated by
Statement (Requirement)	(satisfied by) traced to (verified by)	references (satisfied by) sourced by traced to (verified by)	(satisfied by) traced to (verified by)	(satisfied by) specified by traced to (verified by)	(satisfied by) traced to (verified by)	incurs (satisfied by) traced to (verified by)	alternative of enables traced to results in	(satisfied by) traced to (verified by)	located at (satisfied by) traced to (verified by)	causes mitigates resolves	decomposed by* traced to* related to*	occurs (satisfied by) (verified by)
Time	occurred by	occurred by	occurred by	occurred by specified by	occurred by	occurred by	date resolves decided by occurred by	occurred by	occurred by	occurred by mitigates	occurred by (satisfies) (verifies)	decomposed by* related to*

Simple Diagrams

- Action
- Asset
- Spider
- Others as needed

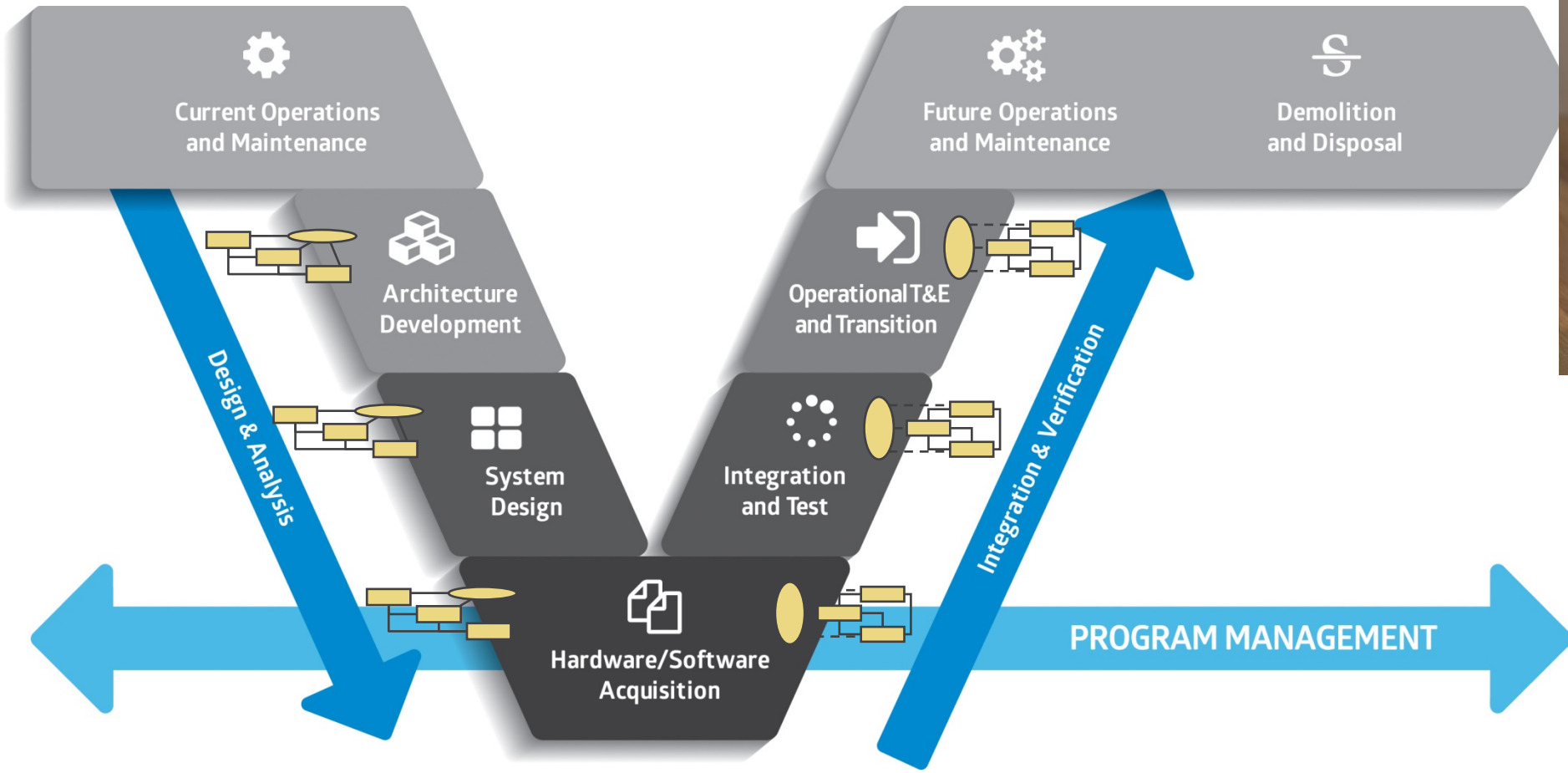


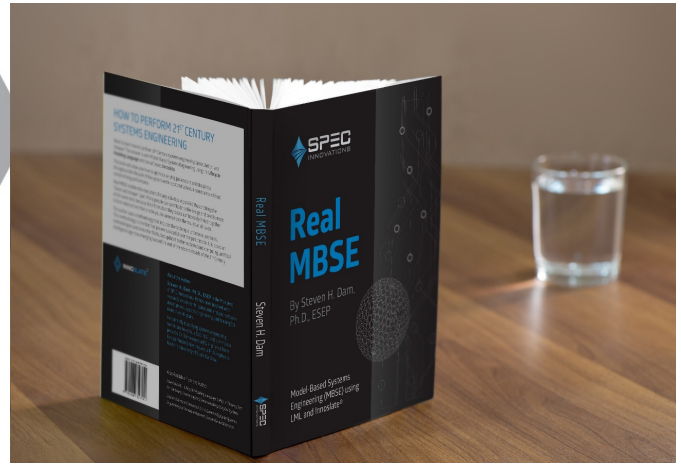




Processes

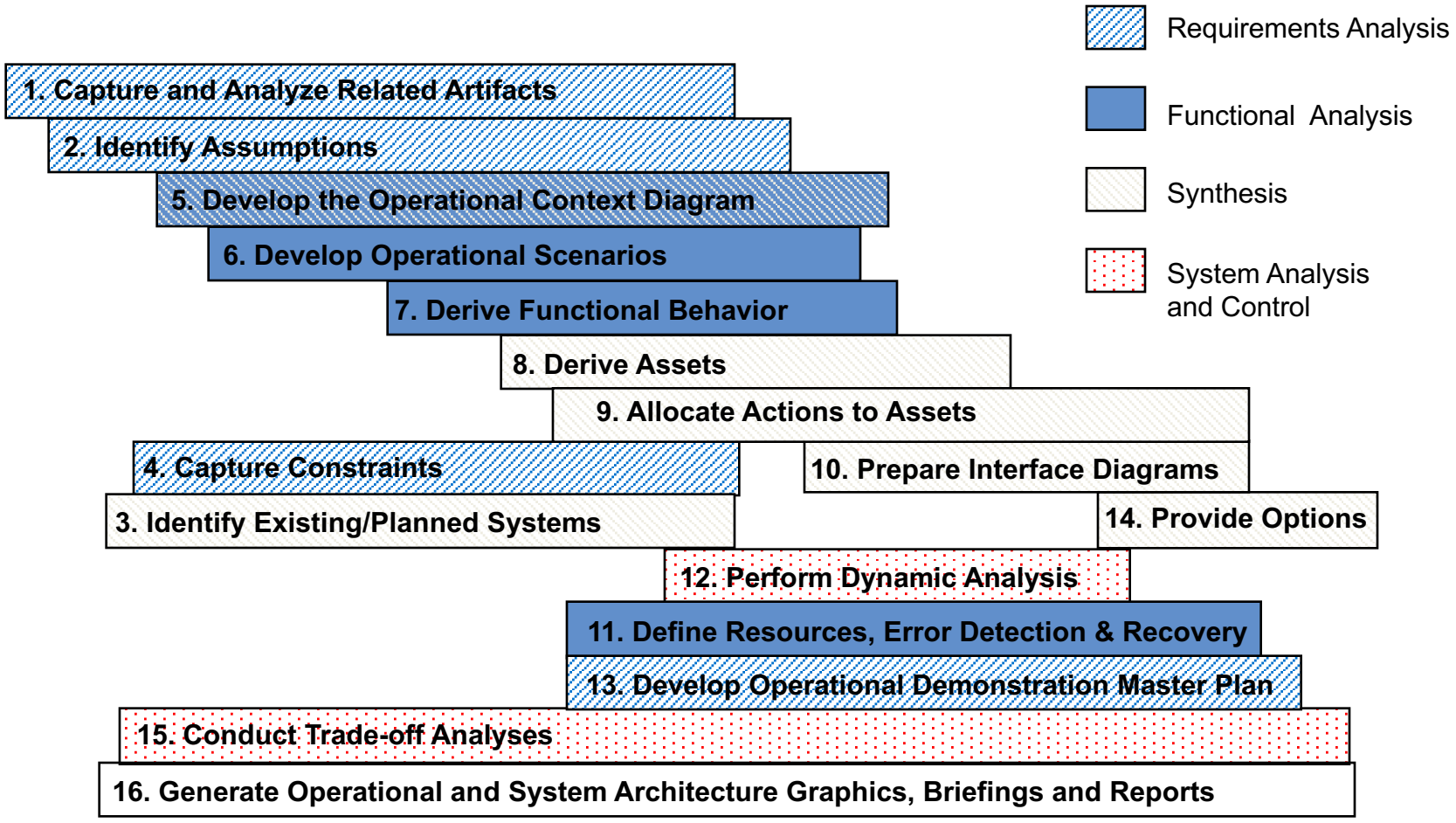
Full Lifecycle Processes

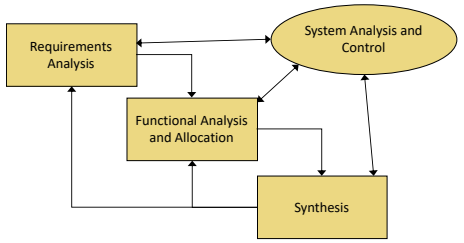




- *Real MBSE: Model-Based Systems Engineering using LML and Innoslate*
- *Available at Amazon.com*
- *Lab Manual coming soon!*

Highly Optimized Middle-Out Process





- Used in the Design & Analysis Phase
- Repeatable at each level of decomposition
- Produces SE products, including DoDAF and UAF
- Highly parallel procedures enable rapid product development

A Real MBSE Tool – Innoslate®

Innoslate® – How We Are Advancing the Future

- Cloud Native: potentially unlimited data availability
  - Ability to store and share knowledge & research
  - Collaboration features (real-time indicators, chat, commenting)
- No downloads or plug-ins
- Highly secure
  - Available on classified networks, private servers
  - 100% developed in the United States
- Major AI advancements (NLP/ML) integrated into Innoslate®
  - Quality checker
  - Traceability assistant
  - Intelligence view
- Discrete Event and Monte Carlo Simulators for cost, schedule, and performance
- Risk charts (matrix & burndown)
- Extremely fast and scalable
- Documents as a model

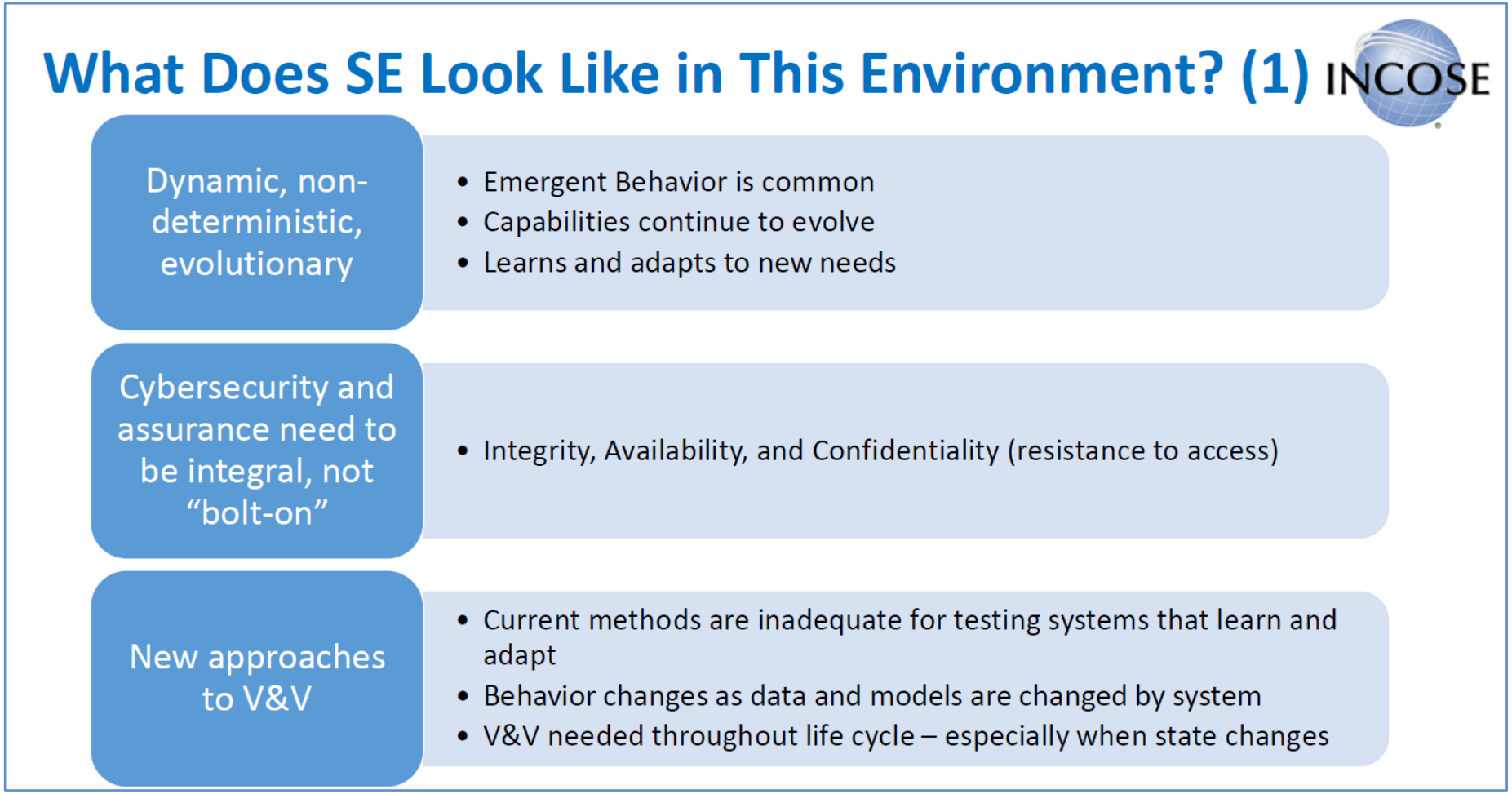
Check it out for free at www.innoslate.com

Innoslate – Full Lifecycle Support

- All views generated from the data and rules – concordance
- Includes 24 diagram types and counting, including all 9 SysML diagrams
- Support integrated program management and systems engineering

The Future of Systems Engineering is Here

INCOSE President's Vision for the Future of SE



Source: Slide from "Shaping Systems Engineering for the Future" – Garry Roedler, ESEP

Is the Future Here? (Part 1)

- **“Dynamic, non-deterministic, evolutionary”**
  - LML takes relationship approach. With over 50+ relationships the systems engineer is able to manage the system by understanding the parts *and their relationships*
  - Innoslate® has both discrete event and Monte Carlo simulators
- **“Cybersecurity and assurance need to be integral not “bolt-on”**
  - LML Action Diagrams enable using decision points at every level, which can be allocated to the CS and IA activities
  - Innoslate® is the most cybersecure for US defense. Made 100% in the US and available on classified networks and private servers
- **“New approaches in V&V”**
  - Innoslate® provides direct V&V support for the lifecycle. Systems engineers can easily see behavior changes as data and models are changed
  - TestCenter and Documents View provide means to conduct test planning early and capture results during and after V&V

INCOSE President's Vision for the Future of SE

What Does SE Look Like in This Environment? (2)

Ongoing modeling and simulation challenges	• Robust modeling and simulation capabilities are needed, but … • How is M&S kept current and controlled when system learns and adapts?
Ongoing operational changes	• Less human dependent, changing Rules of Engagement and Concept of Operations • Changes to training and mission/business parameters
Changes required for a literate workforce	• Much greater man-machine interface, and machine may have the leading role • Need for skilled personnel at all lifecycle phases • Adaptable workforce, as roles will change more quickly - get past culture change issues

Source: Slide from "Shaping Systems Engineering for the Future" – Garry Roedler, ESEP

Is the Future Here? (Part 2)

- **"Ongoing modeling and simulation challenges"**
  - Innoslate's simulators have the ability to simulate decision processes that artificial intelligence would make.
- **"Ongoing operational changes"**
  - By taking a digital engineering approach to CONOPS, Innoslate creates a steady flow of ongoing operational changes for stakeholder awareness.
- **"Changes required for a literate workforce"**
  - The combination of lifecycle modeling language and the collaborative platform in Innoslate makes it possible to have an adaptable workforce and removes confusing language barriers from personnel coming from different backgrounds.

How Will Technology Improvements Enable Better SE?

- Some emerging/available technologies of the future:
  - Cloud computing (already here!)
  - Artificial Intelligence (Natural Language Process is already here!)
  - Graph Databases (already here!)
  - Optical Computing (coming soon)
- How can they help us?
  - Cloud computing provides a means to collaborate worldwide today … SE tools need to take advantage of this capability
  - Artificial Intelligence can help us find design problems or potential problems early
  - Graph Databases enable greater storage capacity
  - Optical Computing will enable create speed of computations, thus allowing for higher fidelity modeling and simulation

Why Do We Have to Wait Years?

- We don't. It's already here!
- As noted, many of these technologies exist today
- The Lifecycle Modeling Language (LML) provides a starting point for your language (www.lifecyclemodeling.org)
  - It's an open standard, free for use
  - It's designed to be the "80%" solution
  - It's a simple language that can be extended it to meet your particular needs
- Innoslate® already uses cloud computing and AI (NLP/ML) technologies and was designed to scale
- With lightening fast speed and scalability, Innoslate can handle AI technologies
- Other tools are beginning to realize these capabilities and are migrating to the cloud

The Future of Systems Engineering is Here

Upcoming Webinars (tentative schedule)

Who	What	When
TBA	TBA	Wednesday 18th March 2020 at 11am EDT
Guy Andre Boy	Human Systems Integration	Wednesday 15th April 2020 at 11am EDT

Invitations will be emailed in advance and informational updates will be placed on www.incose.org

Go to http://www.incose.org/products-and-publications/webinars for more info on the webinar series, including a way to view the last 133 Webinars and soon – this one!

Information on the webinars is now being posted in INCOSE Connect, in the INCOSE Library area, at https://connect.incose.org/Library/Webinars/Pages/INCOSE-Webinars.aspx . Joining instructions will added around two weeks before the webinar is scheduled to take place.

INCOSE Systems Engineering Professional PDU Credit

Please note that you can claim 1PDU credit towards your Systems Engineering Professional re-certification by attending this webinar. The webinars may also apply to the PDU requirements of other organizations, depending on the subject matter

To qualify, you must have attended through at least 75% of the webinar for webinars that last less than one hour, or through 45 minutes of the webinar for webinars that last for 1 hour or longer.

You can also claim credit for previous webinars you have attended; please contact info@incose.org if you wish to know which webinars you attended and if you met the qualification requirements