Validating Interoperability for LVC Simulation for T&E

Dr. David Gross
Lynn D. Stuckey, Jr.

The Boeing Company
Lynn.stuckey@boeing.com

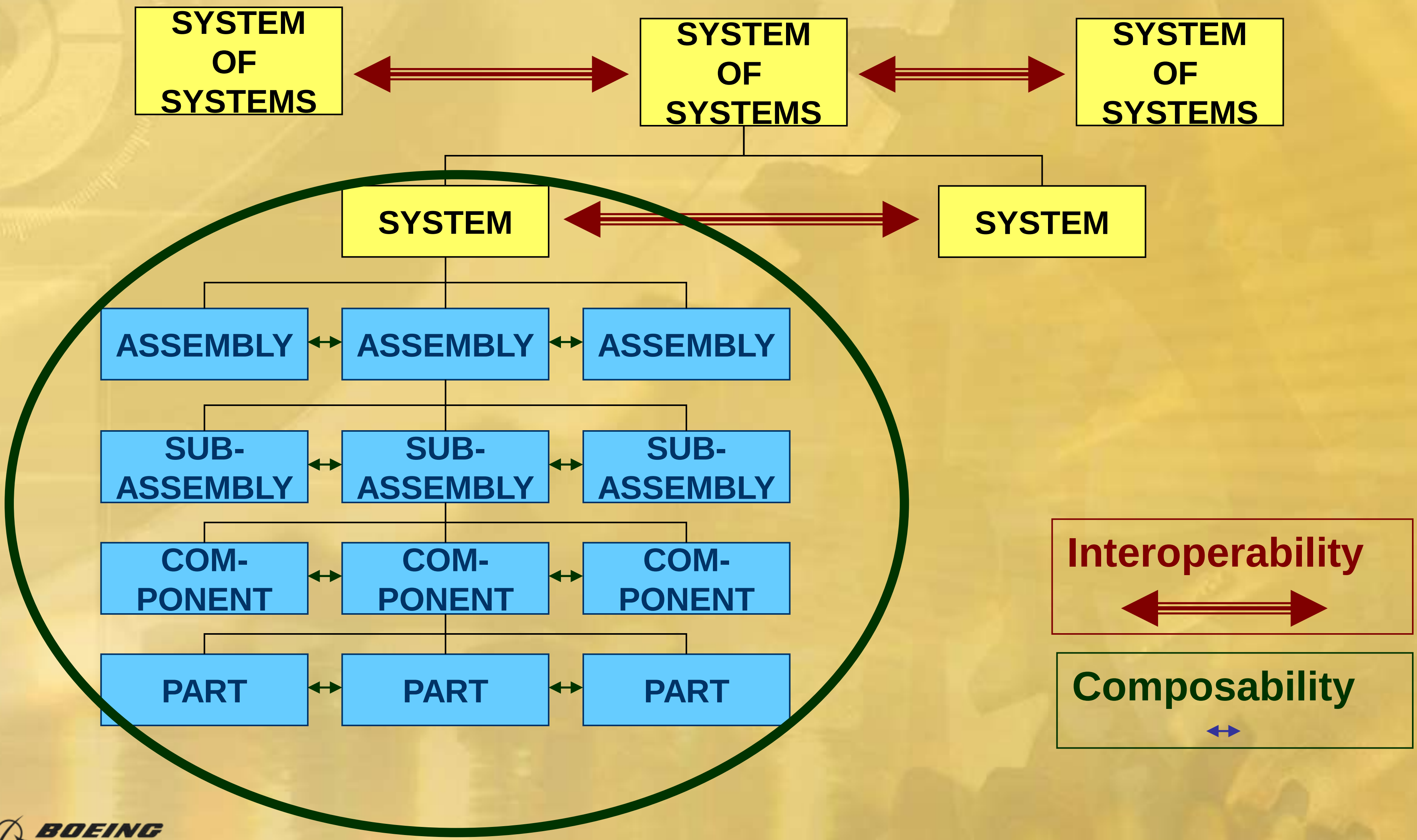
The Problem

- **Test and Evaluation is the most demanding of contexts for simulation validity**
- **Interoperability between LVC assets and the System Under Test is inevitably a difficult requirement in this context**
- **Interoperability is weakly understood in all domains**

Bottom Line:

If we are going to accredit simulation as a test apparatus, we will have to demonstrate valid interoperability between simulation and system under test

Composition & Interoperation



What is composability?

*Capability to **select and** assemble components into systems **to satisfy specific requirements.***

Ability to combine alternate components or to combine them in different ways into systems that meet different requirements

After Petty, 2002

What is interoperability?

*The ability of systems**, units, or forces to provide services to and accept services from other systems, units, or forces and to use the services so exchanged to enable them** to operate effectively together**.***

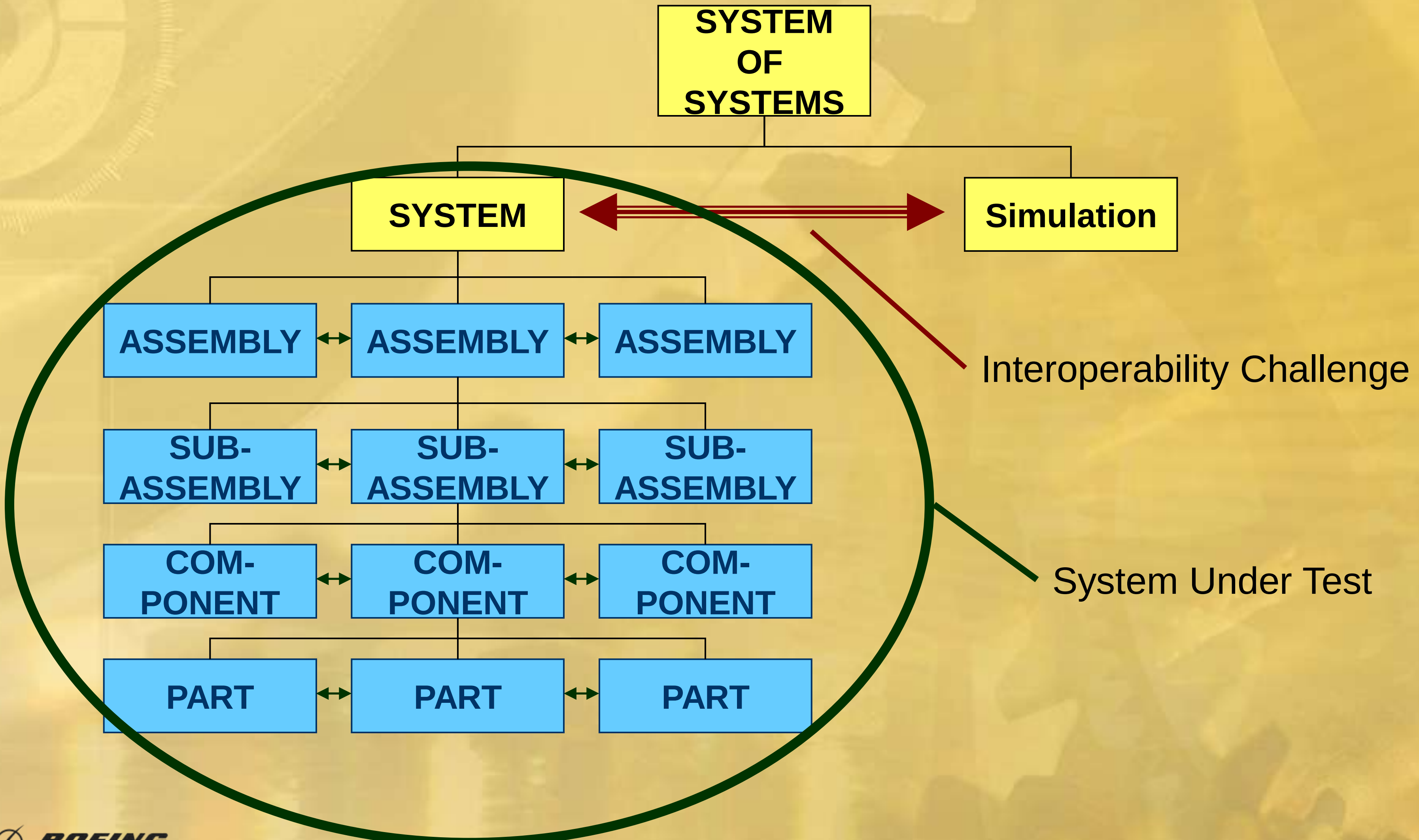
JCS Pub 1

BOEING

Implications

- Composability is of existing components,
- Interoperability requires independent operability, so it is a system of systems problem
- In either case, users have to be able to understand and exploit the meaning (semantics) of the component's existing design
  - For example, intent, limitations, & assumptions.

LVC in T&E


SYSTEM OF SYSTEMS
SYSTEM
Simulation
ASSEMBLY
ASSEMBLY
ASSEMBLY
SUB-ASSEMBLY
SUB-ASSEMBLY
SUB-ASSEMBLY
COM-PONENT
COM-PONENT
COM-PONENT
PART
PART
PART
Interoperability Challenge
System Under Test

How do we solve it?

- **Defining Interfaces?**
- **Tests and measures?**
- **Design principles?**
- **New Knowledge (science)?**

Decomposing Interoperability

- Technical Interoperability
  - **the capability of federates (simulations) to physically connect and exchange data**
- Substantive (or ***Semantic***) Interoperability
  - **the capability of federates when joined together to provide adequate, accurate, and consistent simulated representations which adhere to the principles of "fair fight", and address the mission objectives**

Resolving technical interoperability problems insures that the federation will run but does not guarantee that the federation can accomplish its mission.

Technical Interoperability

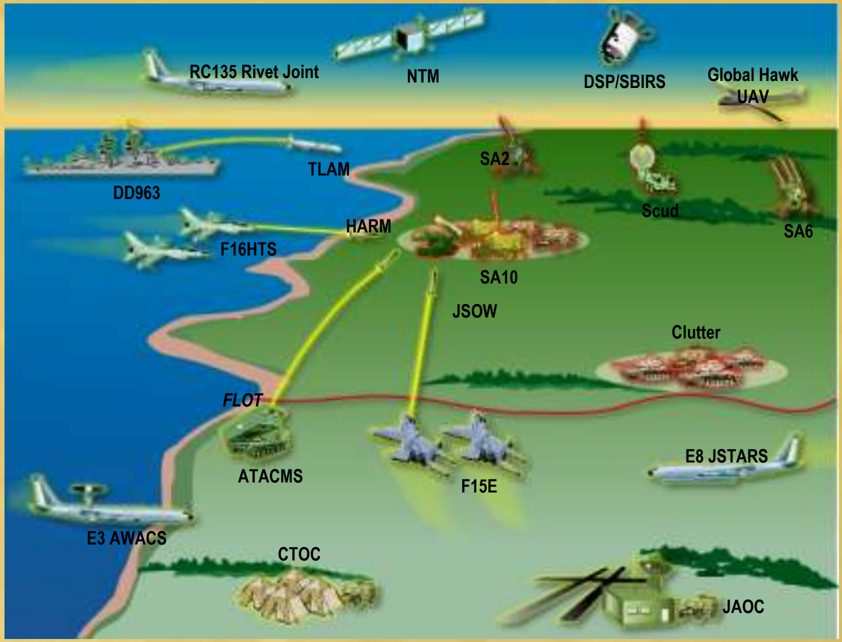
- Characterized by the ability of federates to physically connect and exchange data in accordance with the FOM
- Involves the use of common standards, compatible interfaces and coordinated data structures
- Elements of technical interoperability
  - Hardware compatibility
  - Standards compatibility
  - Time management coordination
  - Coordinated use of RTI services
  - Security issues

supported by HLA infrastructure for example

BOEING

Substantive Interoperability

- Simulations abstract the actual world in their representations, sometimes in very different ways.
- These differences cause validity problems when the simulations share aspects of their representations to build an integrated simulated world.
- Substantive interoperability addresses validity problems at the ***representational level***.



How do we get Substantive Interoperability?

- **The Essence of Substantive Interoperability is *Semantics* (or Meaning)**
- **The semantic are in "the model behind the model"**
  - **"Unspoken" constraints**
  - **Implicit assumptions about context**
    - **Such as between the components of the physical environment: ground, ocean, atmosphere, weather, infrared, and electromagnetic**
  - **Un-described relationships**
    - **Temporal**
    - **Spatial**
    - **Logical**
- **How do we represent/extract meaning?**

BOEING

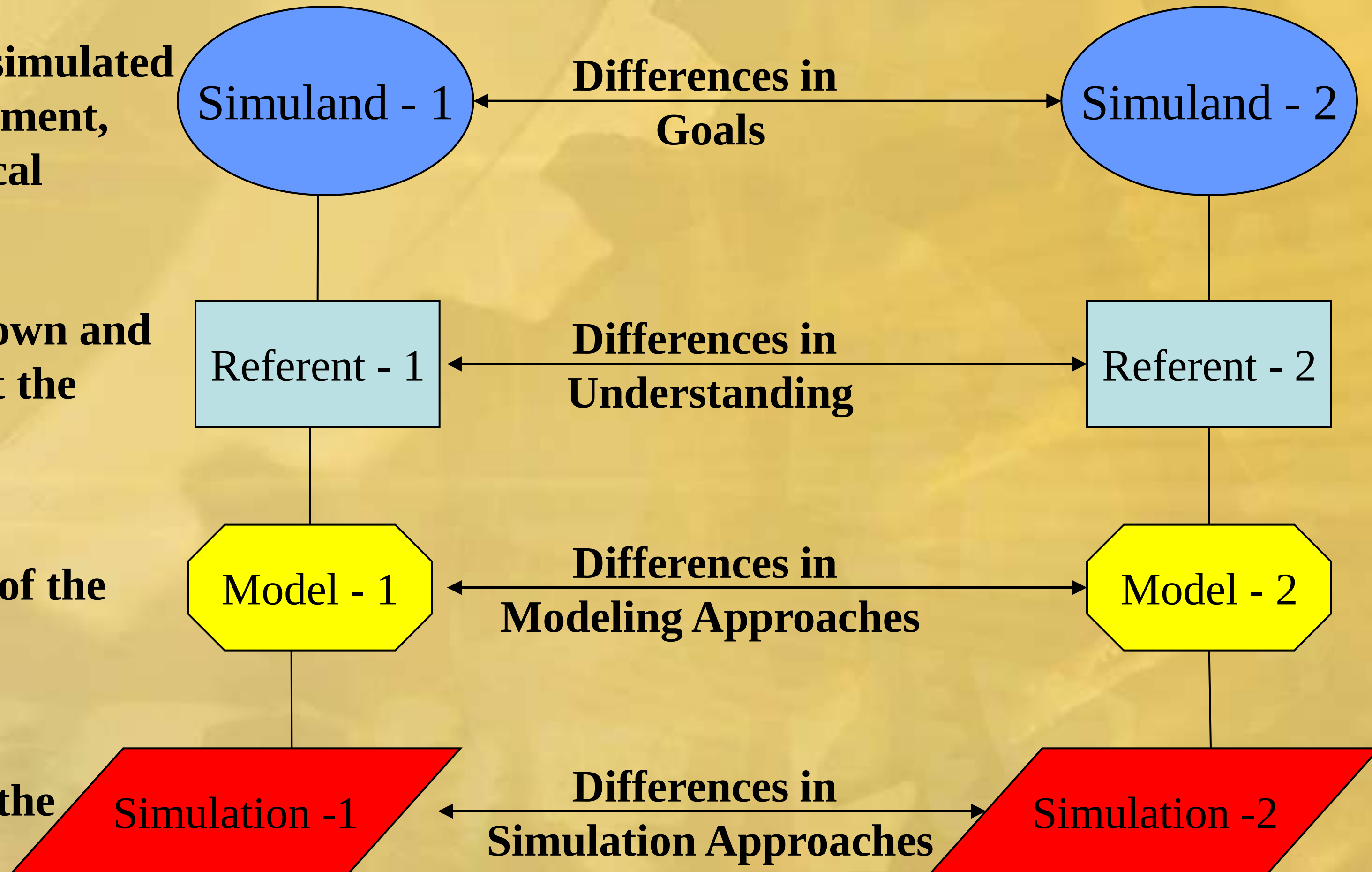
A Diagnostic Approach:
Focus on What Could Be Misunderstood

The things to be simulated and their environment, real or hypothetical

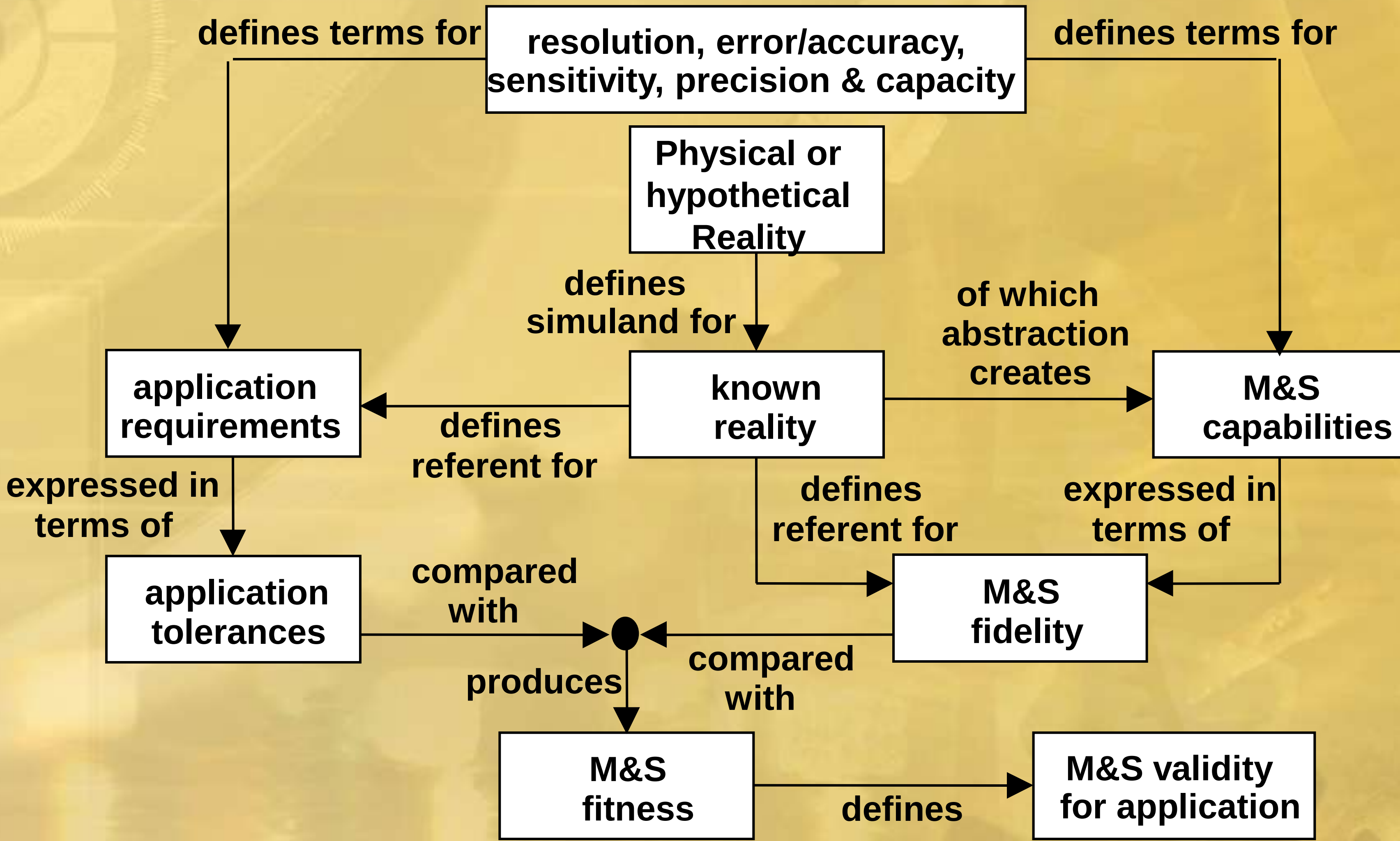
That which is known and understood about the reference

A representation of the simuland

The execution of the model over time



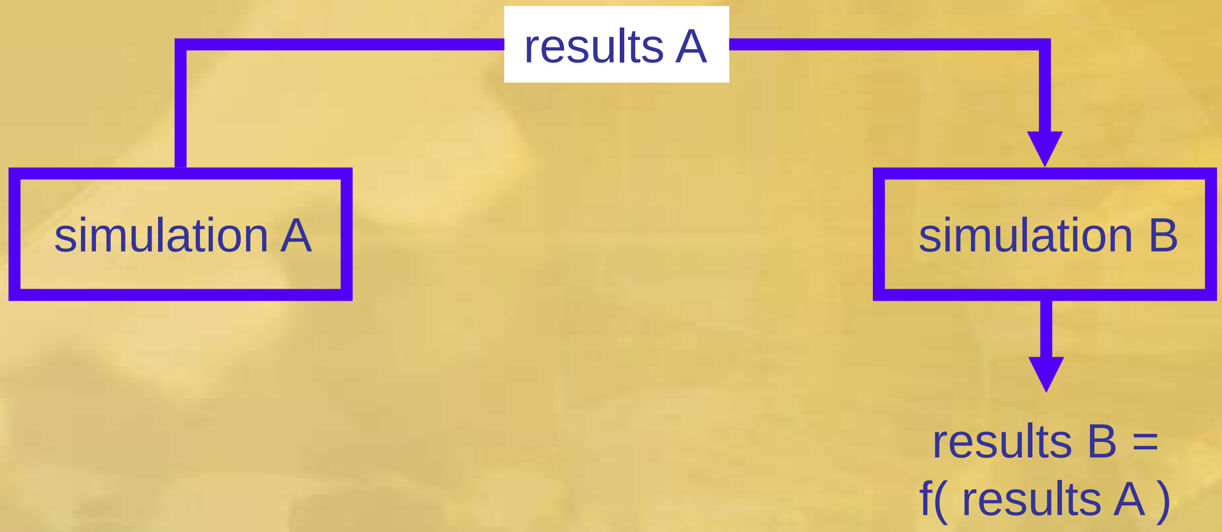
The Simulation Fidelity Framework



BOEING

Interoperability Potential Point of Failure: Functional Composition

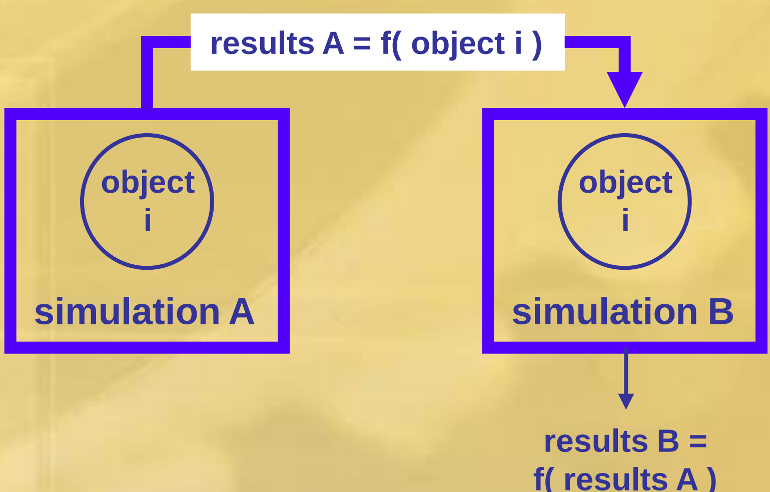
Functional compositions occur when the computation of one or more object states in one simulation depend upon the data provided by another simulation.



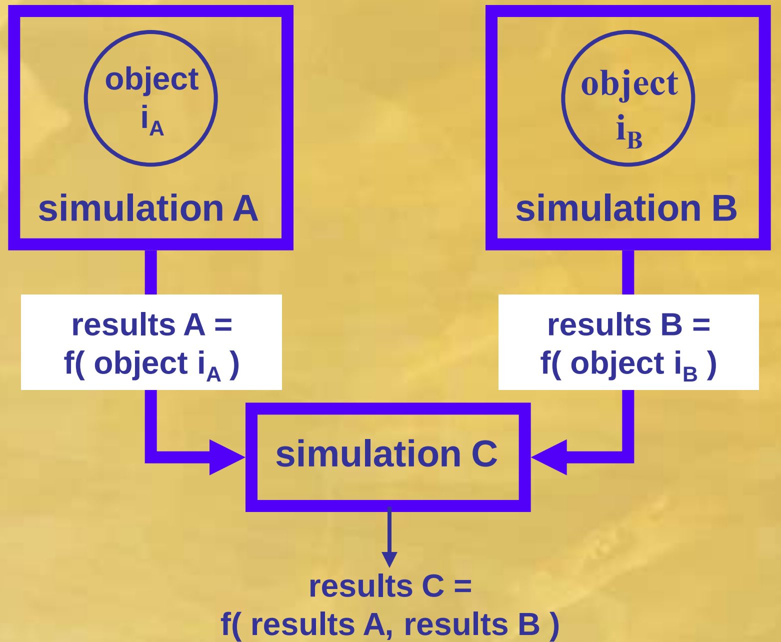
Per Scott Harmon

Interoperability Potential Point of Failure : Manifold Representation

Manifold representations occur when two or more simulations represent the same state or behavior of the same object & interact either directly or indirectly.



Directly Interacting Simulations



Indirectly Interacting Simulations

Potential Interoperability Problems

1. **Range Consistency**
2. **Sensitivity Consistency**
3. **Temporal Representation**
4. **State Consistency**
5. **Event Phase Ordering**
6. **Validity**
7. **Adherence to Performance, Input, Output and Environmental Requirements**

BOEING

A Decomposition of Interoperability Anomalies

- Interoperability Anomalies
  - Transport
    - Hardware compatibility
    - Standards compatibility
  - Syntactical
    - Time management
    - Simulation Framework Services
  - Semantic
    - Physical Coherence
    - Contextual Coherence
    - State Coherence
    - Event Order
    - Event Phase

Analysis Classes Applicable to All Coupled Simulations

Classes	*Discussion*
Scenarios Class	The effect of external conditions or scenarios for the interoperating simulations must be consistent. This class of analysis actions compares scenario dependent data, setup or responses common to scenarios input across the simulations.
Timeline Class	A validly interoperating simulation set will execute functions and generate data in the same order as the "real world" system. Therefore the within the integrated simulations, state changes, data / message generation should follow the same characteristics as the system. A comprehensive approach to validate that the coupled simulations function the same as the system is to define timelines in this class of semantic interoperability metrics to measure this capability.
Action - Reaction Class	End-to-end timelines are usually too complex to allow detailed stimulus-response or causality analysis. In addition to timelines, this class addresses the properties of Action-Reaction and Causality. Action-Reaction and Causality are both required to define necessary and sufficient conditions for timing interoperability. Explicit validation of Action - Reaction (Stimulus-Response) is required because it is possible for coupled simulations to exhibit a consistent timeline and causality without semantic interoperability. This condition occurs when a control variable is passed from one simulation to induce a response in a second simulation. There is no violation of Causality yet the lack of a response from the second simulation, when there should be one, indicates that the integrated simulations are not interoperable.

Analysis Classes Applicable to All Coupled Simulations, cont'd

Classes	*Discussion*
Causality Class	An interoperable coupled simulation set will execute functions and generate data in the same order as the "real world" system. Therefore the within the integrated simulations, state changes, data / message generation should follow the same characteristics as the system. Action-Reaction and Causality are both required to define necessary and sufficient conditions for timing interoperability. Transmission of a control variable from one simulation does to a second simulation could stimulate a response that exhibits a consistent timeline and satisfy Action-Reaction but could violate Causality if the simulated time of the response occurs prior to the simulated time of the stimulus. To complete the analysis of timing, we must address causality in addition to the timeline and Action-Reaction. While stimulus - response ensures that there was a response to an input, verifying causality ensures that any of the functions did not execute prior to the point in time where it should have.
Temporal Accuracy Class	"Temporal Accuracy" is a combination of the domain, range, sensitivity and resolution of time as it is maintained by or passed between simulations. This class of semantic interoperability metrics addresses the characteristics and use of time with in each simulation.
Natural Law Class	This class of semantic interoperability metric ensures that the simulations "Follows the Applicable Laws of Nature." Analyses in this class specifically address the underlying process, chemistry or physics represented by the simulations.
State Consistency Class	It is possible that the same object / entity is represented in two different simulations. This class of semantic interoperability metrics refers to the states of entities that may be duplicated in two different simulations and ensures that the states of the entity in each simulation are with acceptable tolerances.

Analysis Classes Applicable to Interfaces between Coupled Simulations

Class	*Discussion*
Data Type Consistency Class	**This class of semantic interoperability metrics refers to the data exchanged by the coupled simulations. Since we have established syntactic interoperability, we know that the data is correctly passed between the simulations. Here we ensure that the type of data is consistent with its use in each simulation. A complete set of analyses for this class would explicitly validate every data item that is exchanged between the simulations.**
External Input Class	**This class addresses the operating environment for the integrated simulations and explicitly addresses the "Domain of the External Input." Since we assume that the intended use that drove the development of the individual simulations did not include the intended use of the coupled simulations, it's entirely possible that the operating environment of the coupled set of simulations is greater than either of the individual simulations. Consequently, we need to validate that the following data are properly used:Inputs; Initial Conditions; Parameters; Scenarios**
Output Class	**This class of semantic interoperability metrics addresses the "Range of the Output" to make sure that a simulation will not generate an output that is passed to a second that exceeds its domain. The "range" of the output in this context can also include the: Norm (magnitude of a scalar, vector or matrix); Orientation; Rate of Change**
Input Class	**Here we address the "Domain of the Input" to ensure that a simulation can correctly accept and utilize the input. For interoperability, the domain of the simulation must be a superset of the range of simulation which output the value to be used as input. We need to consider this because the domain and range of the variable can be dynamic and uncoupled.**

BOEING

Analysis Classes Applicable to Interfaces between Coupled Simulations , cont'd

Class	*Discussion*
Output Precision Class	**Precision is defined the level of resolution or granularity with which a parameter can be determined. In this class of semantic interoperability metrics we consider the "The Precision of the Output" to ensure that small changes in the state of a simulation results in outputs or changes to its outputs of that would be seen by a receiving simulation.**
Input Sensitivity Class	**Sensitivity is defined as the ability of a component, model or simulation to respond to a low level stimulus. In this class of semantic interoperability metrics we consider the "The Sensitivity of the Input" to ensure that changes in the inputs to a simulation are observed by the simulation and result in an appropriate response.**
Input Resolution Class	**Resolution is defined as the degree of detail and precision used in the representation of real world aspects in a model or simulation. Resolution therefore matches the magnitude of the spatial and temporal units in the simulations.**
Domain Constraint Class	**This class of semantic interoperability metrics is designed to address those kinds of issues that are unique to one of the simulations by explicitly adding any domain constraints and testing for the violation of this limit.**

Semantic Metadata

- **Simulation metadata can capture the assumptions and decisions (or lack thereof) that cause these anomalies, and can be exploited to measure or predict the degree of semantic interoperability of a simulation built from the subsystem simulations that this metadata describes.**
- **Metadata such as:**
  - **Inputs and Outputs**
  - **Initial Conditions**
  - **Parameters**
  - **Scenarios**
  - **Distributions**
  - **Controls**
  - **State Variables**

Metadata for each Object in the Simulation

Metadata	*Discussion*
Initial Conditions	**Static scenario inputs (not generated by other simulations)**
State Variables	**Variables internal to the simulation which determine the simulation's state, or ability to process inputs and produce outputs.**
Resolution	**identify the level of representation**
Capacity	**how many of this object may be represented**
Existence of / potential for functional compositions	**when the computation of one or more object states in one simulation depend upon the data provided by another simulation**
Existence of / potential for manifold representations	**occur when two or more simulations represent the same state or behavior of the same object & interact either directly or indirectly**

Metadata for each Control (input) of each Object

Metadata	*Discussion*
Range	**the set of values the control may take on**
Frequency	**the update rate for the control**
Precision	**the smallest change possible in the control**

Metadata for each <u>Behavior</u> (output) of each Object

Metadata	*Discussion*
Error	**the difference between an observed, measured or calculated value and a correct value**
Sensitivity	**the smallest non-null change possible in a model's outputs**
Precision	**the smallest input change that produces a change in the model's output**

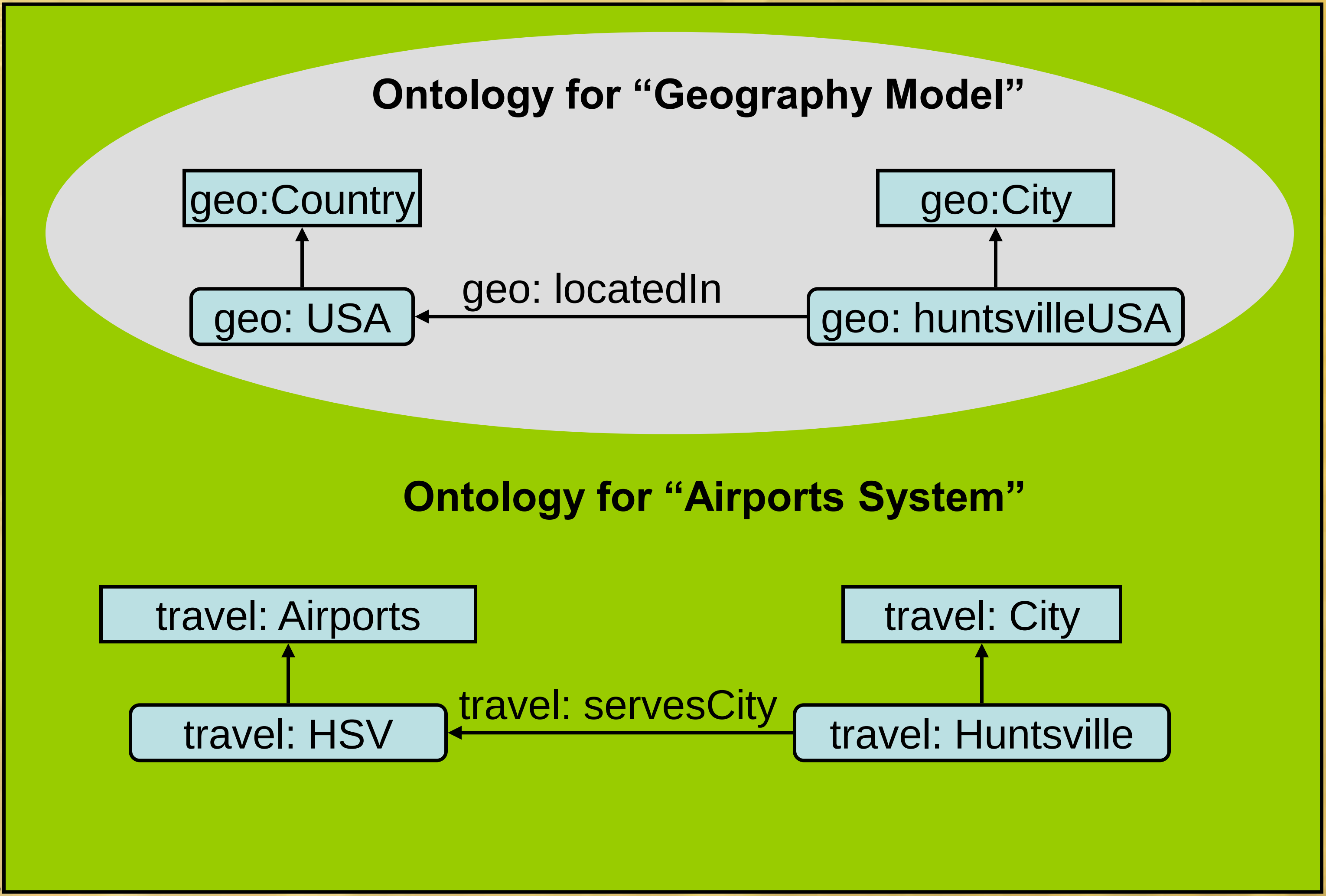
Ontologies Offer a Vehicle to Manipulate the MetaData

- An ontology is a data model that represents a set of concepts within a domain and the relationships between those concepts.
- It is used to reason about the objects within that domain.
- Ontologies are used in artificial intelligence, the semantic web, software engineering, biomedical informatics and information architecture as a form of knowledge representation about the world or some part of it.
- Ontologies generally describe:
  - Individuals: the basic or "ground level" objects
  - Classes: sets, collections, or types of objects[1]
  - Attributes: properties, features, characteristics, or parameters that objects can have and share
  - Relations: ways that objects can be related to one another
  - Events: the changing of attributes or relations

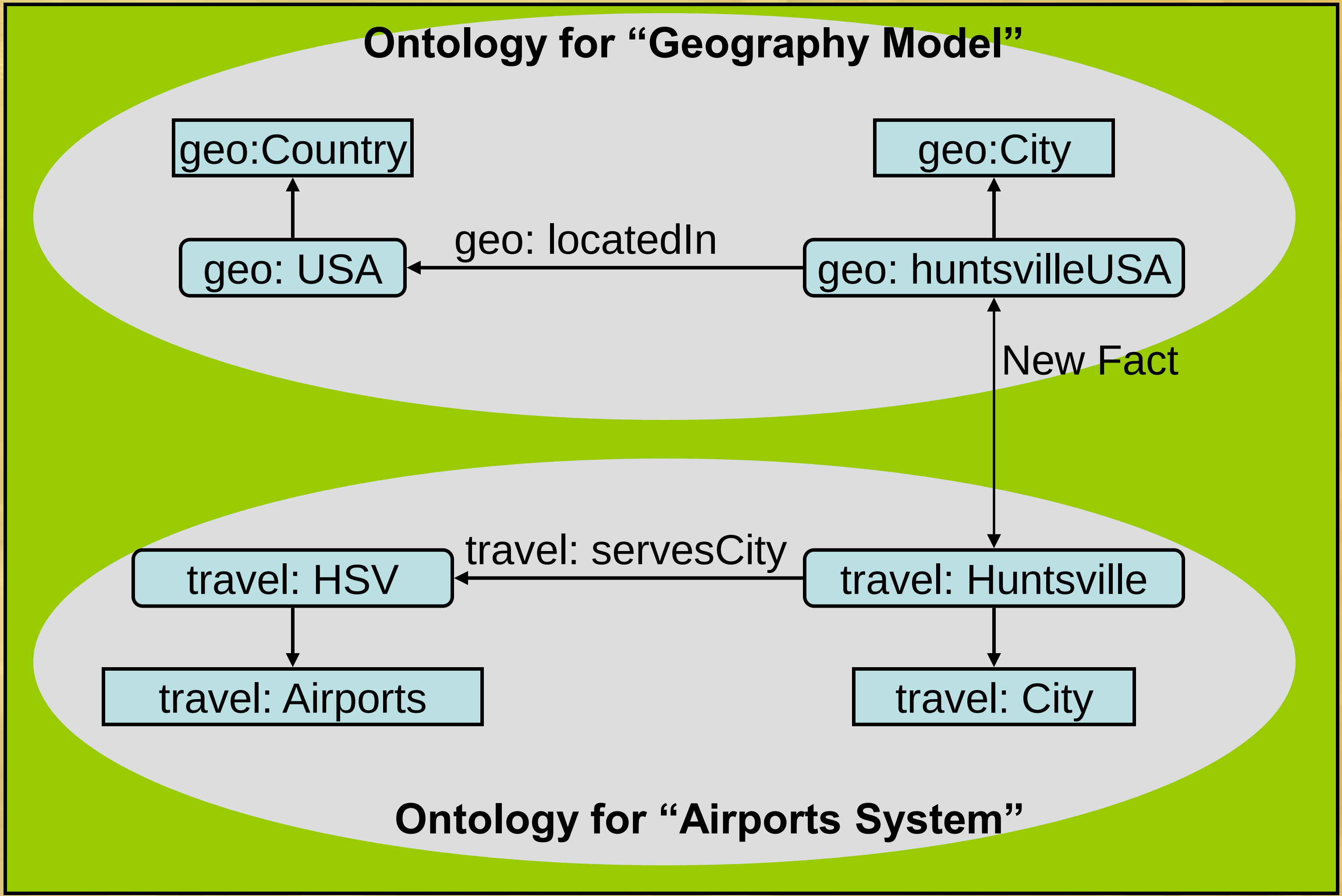
Web Ontology Language (OWL)

- **Note: OWL is not a "settled" technology, so discussions of what it provides and does not provide have a certain risk of being overcome by events.**
- **OWL provides syntax for capturing this information, but more importantly enables reasoners which operate on this data.**
- **OWL reasoners are typically *open-world reasonsers*, which is the inverse of how we typically think about inferences.**
  - **In closed world reasoning, the assumption is that two items are related only if they are expressly stated to be.**
  - **In open world reasoning, two items are assumed to be possibly related unless expressly stated not to be.**
  - **Closed world reasoner say "it is"; open work reasoners say "it could be".**
- ***We can use the reasoner to discover new possibilities in the federated ontologies, and to discuss inconsistencies between them.***

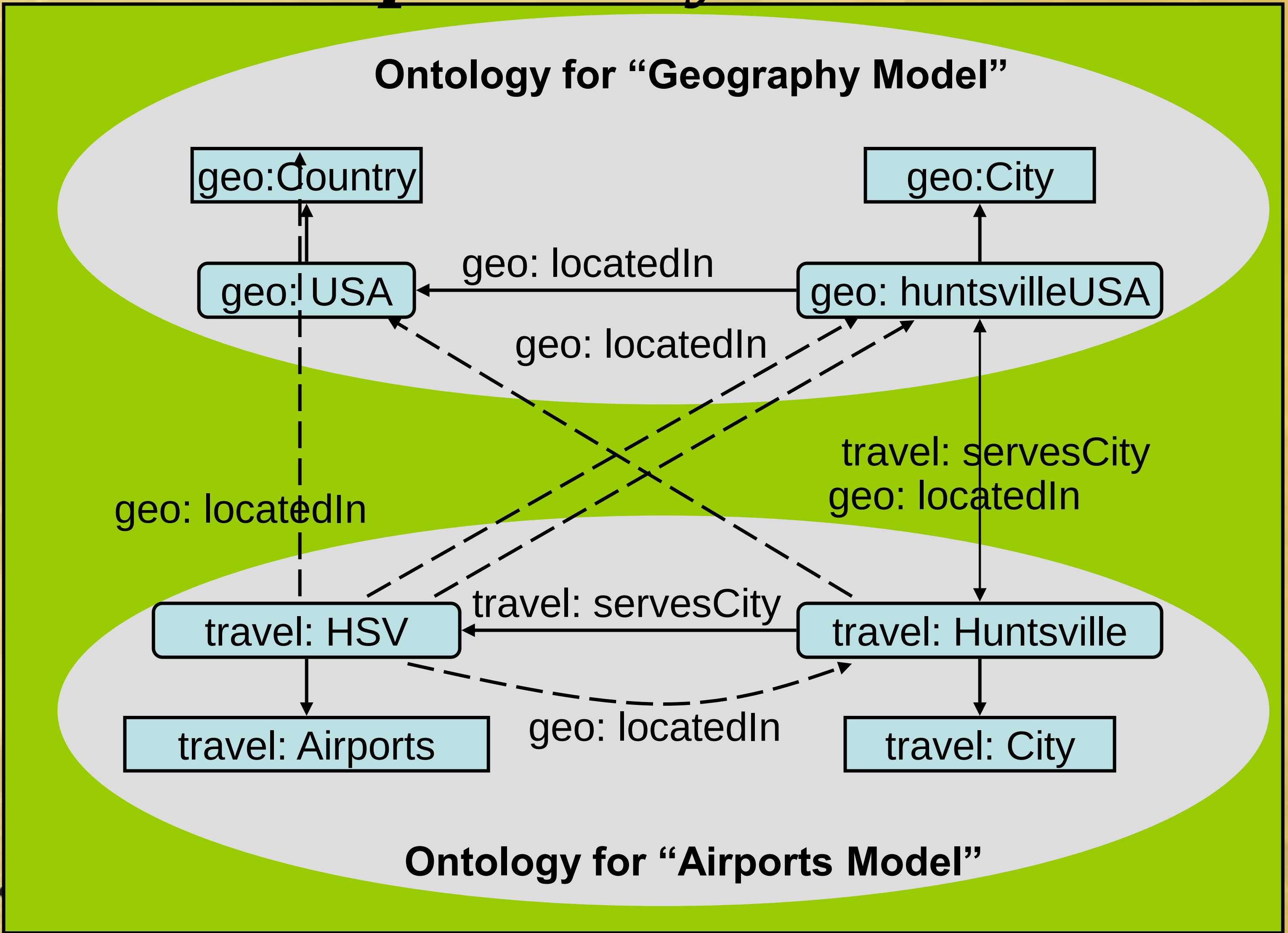
Example: Ontologies for Simulations and System



One New Fact Asserted as an Interoperation



Results in Many New, Testable Interoperability Assertions



Summary

- To achieve Semantic Interoperability between a System Under Test and a Simulation, we need to understand the LVC simulation
- Understanding comes from capturing the semantic context of the simulation.
- This semantic context can be captured in metadata, which must be extracted from the existing component or captured at design time.
- The nature of the Systems of Systems problem, and the simulation of such, defines the kind of metadata required.
- This metadata needs to be tested for internal consistency, accuracy (consistency with simuland), as well as completeness.
- Specific technologies such as OWL offer purchase on this problem of representing and reasoning on the metadata.