



One Semi-Automated Forces (OneSAF)

Capabilities, architecture, and processes

John Logsdon, Deputy PM, OneSAF

Dave Nash, Ph.D.

Mike Barnes

Report Documentation Page				Form Approved OMB No. 0704-0188	
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE 10 MAR 2008		2. REPORT TYPE N/A		3. DATES COVERED -	
4. TITLE AND SUBTITLE One Semi-Automated Forces (OneSAF) Capabilities, architecture, and processes				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) TRADOC Analysis Center				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release, distribution unlimited					
13. SUPPLEMENTARY NOTES 2008 DoD M&S (Modeling and Simulation) Conference Presentations held in Orlando, Florida on March 10 - 14, 2008, The original document contains color images.					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UU	18. NUMBER OF PAGES 44	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			

User communities

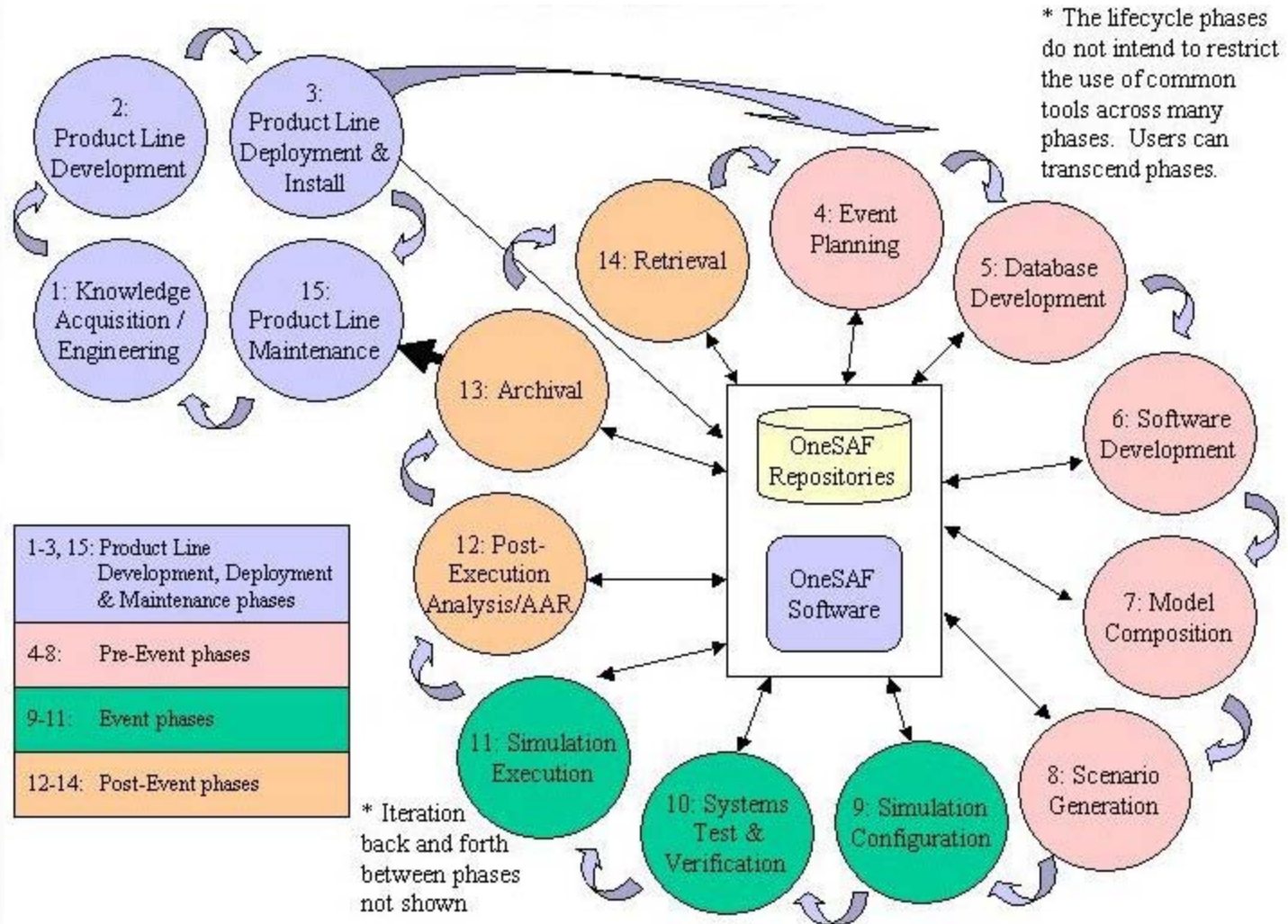
OneSAF is chartered to serve three communities, each of which has its own requirements and priorities:

- Advanced Concepts and Requirements (ACR)
- Research, Development and Acquisition (RDA)
- Training, Exercise and Military Operations (TEMO)



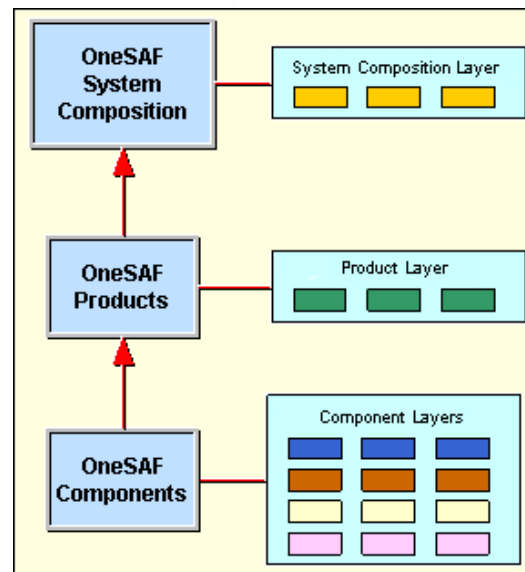
In addition to these, PM OneSAF also directs support that is to be provided to various joint and civilian organizations, and international customers.

The Simulation Lifecycle

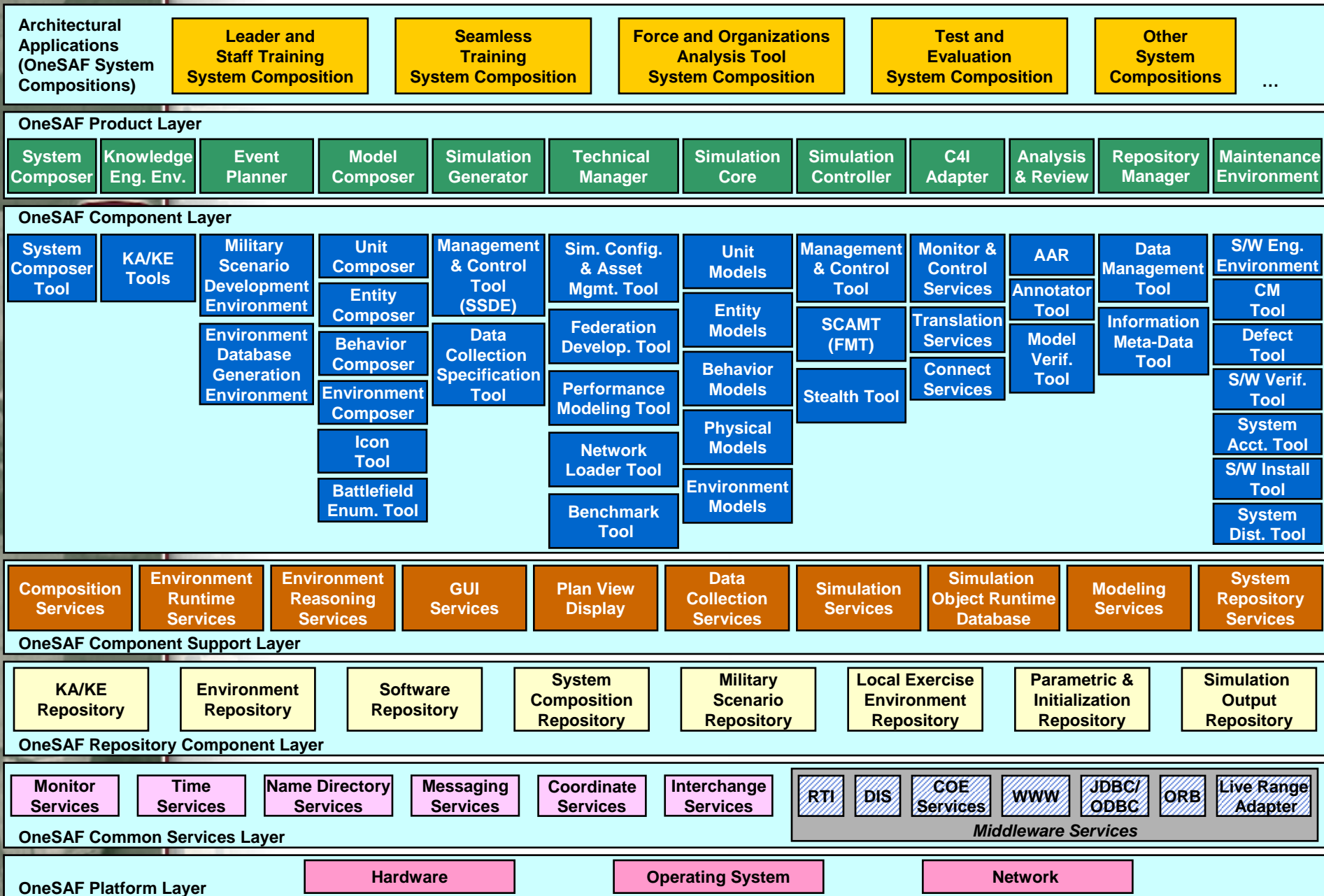


Composable Product Line Concept

- OneSAF is not a single system or product.
- OneSAF is a set of:
 - *Components*,
 - *Products* comprising OneSAF components,
 - *System compositions* comprising OneSAF products
- Tools are provided to create compositions of various types (including system compositions).



The Product Line Architecture Framework



MSDE / AAR

- Military Scenario Definition Environment (MSDE)
 - PowerPoint-based
 - Creates scenario definition (units, command relationships, dispositions) in Military Scenario Definition Language (MSDL)
 - MSDL is imported into a OneSAF scenario for execution
- After-action Review (AAR) Tool
 - Generates slides of the simulated scenario
 - Statistics of “snapshot” data items

Composability

- Composability:
 - The ability to assemble reusable software elements (“components”) into customized software.
 - A central design theme of OneSAF.
- Fairly recent capability in software development enabled by:
 - increased computing power,
 - success in developing reusable software libraries, and
 - the development and distribution of software and software components over the internet to different computer systems.
- Instead of developing complete computer programs for each computer system, software developers develop platform independent components to be assembled on the user’s computer.

Composition nomenclature

- Use of the term "composition" is ubiquitous in OneSAF, and usually requires context to know exactly what is being described:
 - *System composition*: a JAR file consisting of a set of system components
 - *Component composition*: an XML file containing pointers to various Java classes; used to imbue actors with capabilities such as:
 - Vulnerability, mobility, and sensing (physical components)
 - Responsiveness to orders, ability to process fire missions (behavioral components)
 - Actor compositions:
 - *Entity composition*: an XML file containing metadata and pointers to component compositions defining a single, indivisible battlespace object
 - *Unit composition*: an XML file containing metadata and pointers to component compositions defining a single, indivisible battlespace object
 - *Behavior composition*: an XML file containing metadata, inputs, (in the case of composite behaviors) outputs, and control definitions defining the effect of a behavior

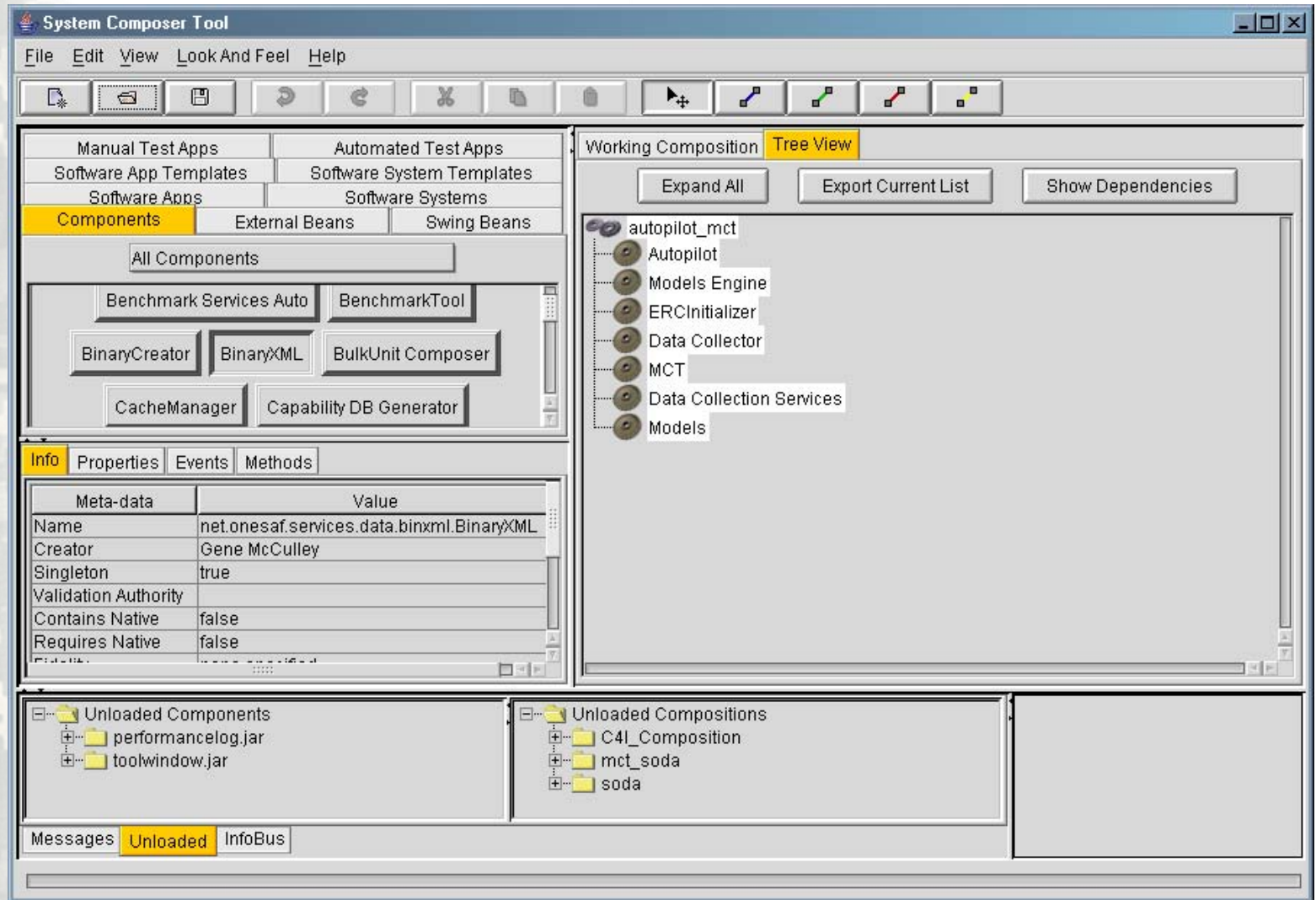
OneSAF System Components

- A OneSAF *system composition* is an assembly of system components. In some ways a system composition is analogous to an executable program.
- A *system component* is a modular software artifact that makes specific functionality available to the system composition.
- Some examples:
 - The Formation Editor is a system component providing a graphical tool that may be used to interactively design formation data files.
 - The formationEditor system composition consists of the Formation Editor and the Desktop components.
- OneSAF components are JavaBeans with extensions; e.g.:
 - OneSAF specific component metadata such as releasability, dependencies, and fidelity
 - Non-Java code support to allow reuse of code written in other languages.
- Components can contain any subset of:
 - Data
 - Software source code
 - Software executable code
 - Documentation

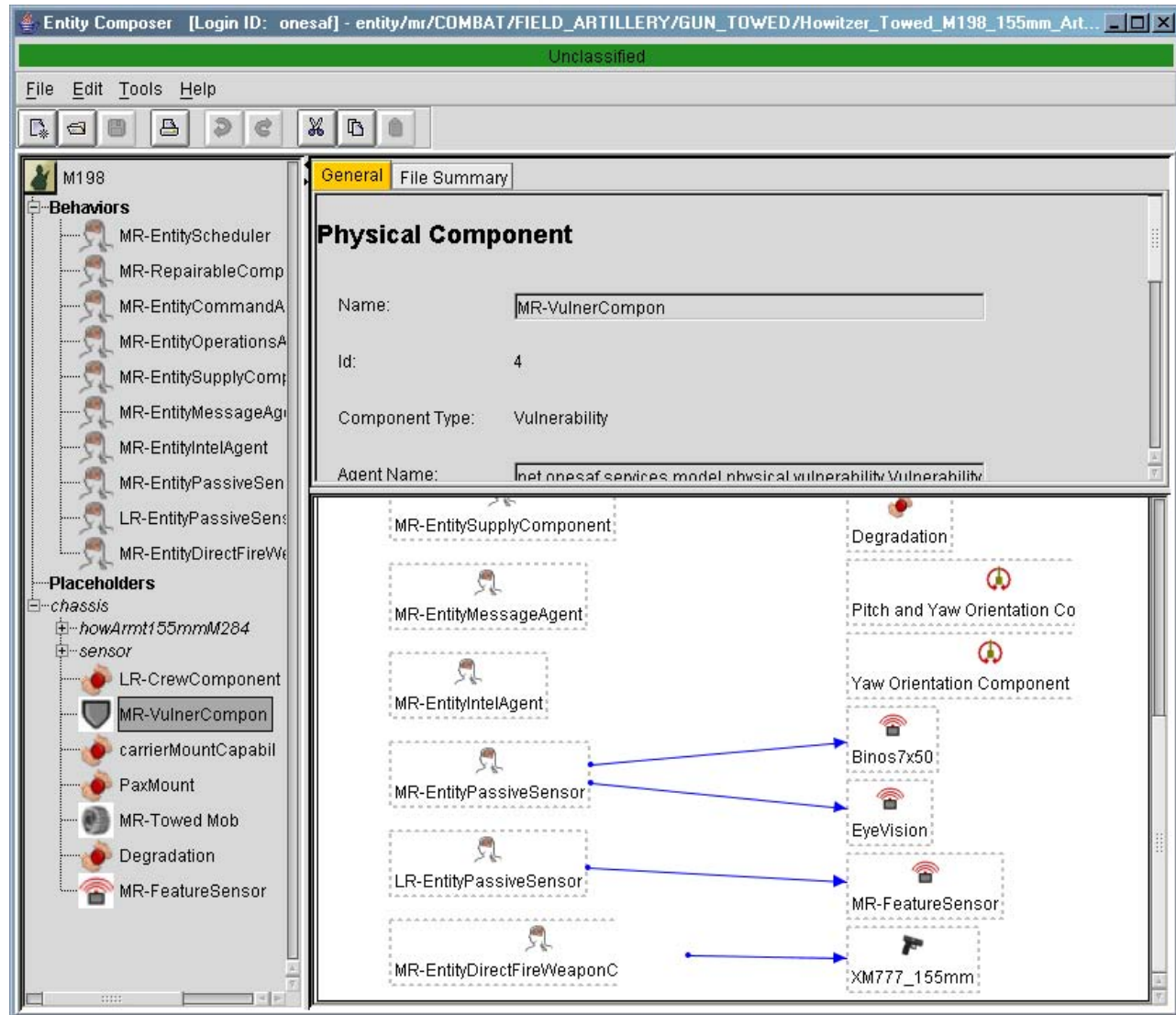
System Composer

- The System Composer is a graphical tool used to create OneSAF system compositions:
 - The System Composer is based on JavaBean application builders but works with OneSAF extended JavaBeans (OneSAF system components).
 - The System Composer creates a JAR file for each composition (the composition and JAR file have the same name) containing:
 - Information the runtime loader script uses to run the composition.
 - Composition metadata; e.g.,
 - Description
 - Specification and Implementation Version
 - Validation Authority and Date
 - The System Composer provides the software “glue” connecting components.
- It is executed from the command prompt by running the composer script within `SWR/bin/services/sys/composition`
 - This is the only stand-alone application within OneSAF; all other applications are started using the runtime loader script in the same directory

System Composer



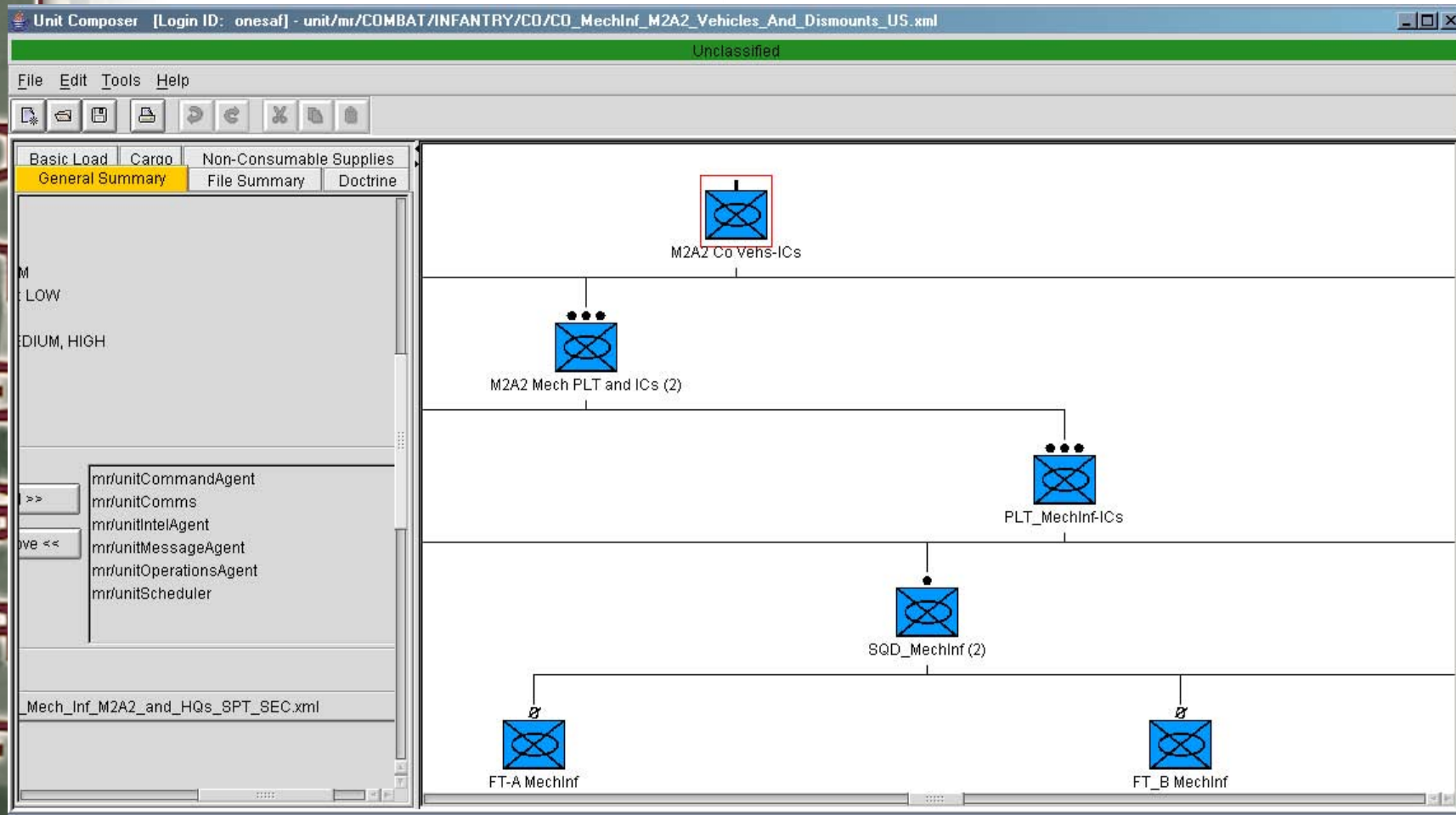
Entity Composer



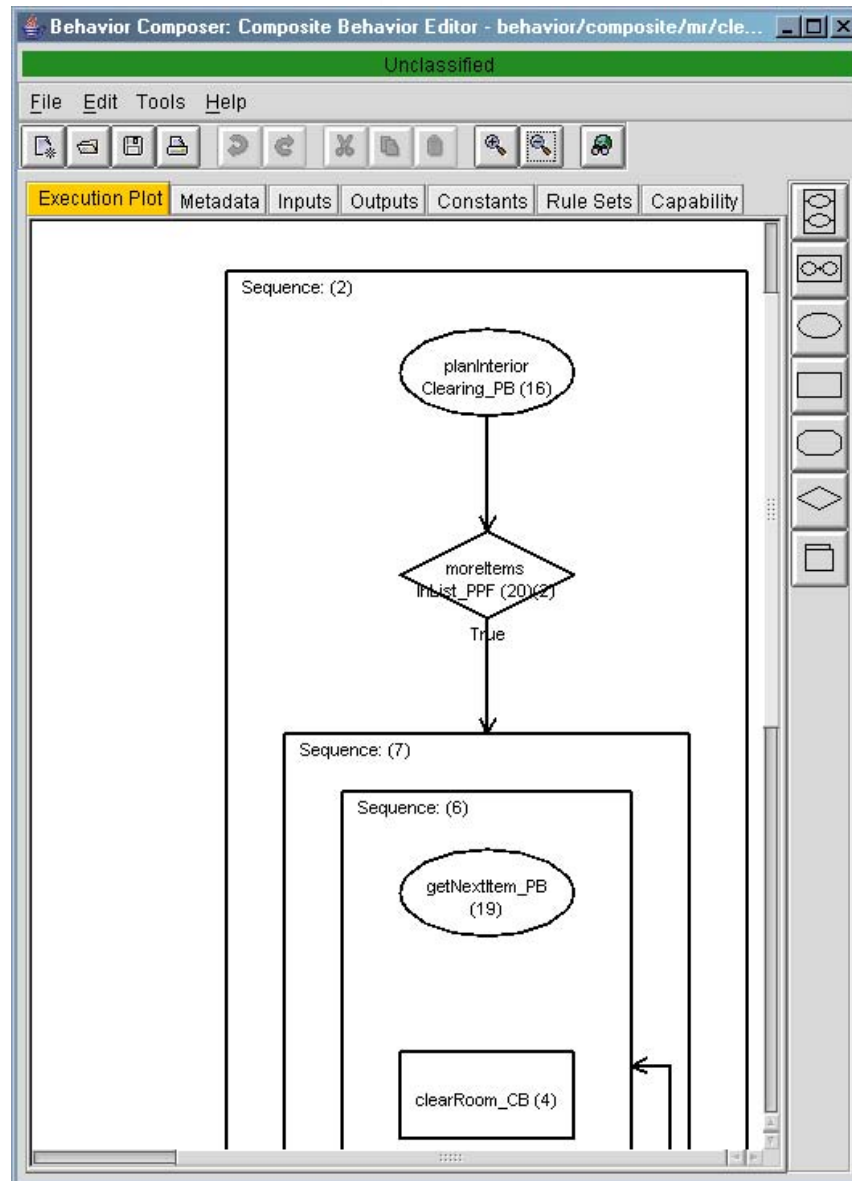
Using the Entity Composer

- System composition = composers, therefore \$BINDIR/runtimeloader composers
- Typical components
 - Physical
 - Sensors (eyeballs, FLIR, etc.)
 - Weapons
 - Mobility
 - Vulnerability
 - Behavior
 - Direct fire controller
 - Operations
 - Intel
 - Supply

Unit Composer



Behavior Composer



The Management and Control Tool

MCT [Login ID: onesaf] - COE_Sniper_Basic_T012-A1

Unclassified

File Edit View Manage Replication Exercise Control Checkpoint Tools Window Help

PVD

1:482.379 Views Select Any

Task Organization

Task Org Mission Description

Name	Type
Coalition	
UHS convoy_mr	Transportation
UHS 1/convoy_m	Transportation
EHS convoy_mr:HMMWV	
UHS snipers	Infantry
EHS snipers:AUTIC, Loaded	
EHS snipers:FIR IC, Loaded	
Insurgents	
EHS opforSniper	IC, Loaded
EHS svdSniper	IC, Loaded
civilian	
nc1	IC, Normal
nc2	IC, Normal
nc3	IC, Normal

GROUND FLOOR

LEVEL 3

ROOF

Mission Editor

	Phase 1	Phase 2
On Command	On Command	On Command
EHS opforSniper	CP Change Posture CP Move Tactically	CP OccupySniperPosition CP
EHS svdSniper	CP Change Posture CP Move Tactically	CP OccupySniperPosition CP
S civilian		

Engage ICs Engage Convoy Engage Civ

Status - opforSniper

General Supplies Command Relationship External Components

Supplies Roll-up Class III 100.0 % Class V 100.0

Composition entity/mr/COMBAT/INFANTRY/Sniper_SVD_Inf_MRC_RS_IC

Type IC, Loaded

Name opforSniper

Activity Requesting Report Message

Edit

Mission Editor Status - opforSniper PVD Task Organization

POV: Coalition Elapsed Sim Time: 00:00:00:00 Sim Time: Jan 01 00:00:00 Sim Scale: 0.00 Clock: May 21 14:24:43 Node: localhost Current sim state is 'Loaded' Alerts

Sides and Forces

- A *force* is a collection of entities, units, or forces.
- A *side* is a collection of entities, units, forces, or sides.
- Sides have relationships which determine certain behaviors.

The screenshot displays the MCT - jsceITeDIED software interface. The top menu bar includes File, Edit, Scenario, Replication, Object, Simulation, Checkpoint, Tools, Network, View, and Window. Below the menu is a toolbar with various icons. The main interface is divided into two primary panels: Task Organization on the left and PVD (Physical View Display) on the right.

Task Organization Panel: This panel shows a hierarchical tree structure under the 'Task Org' tab. The 'Mission Description' sub-tab is active. The 'Org View' is set to 'Commanded'. The tree structure is as follows:

- Top Level
 - AI-Jahari
 - Coalition
 - convoy (Type: trans...)
 - convoy:SQLM10...
 - convoy:TRIM10...
 - convoy:TRIM10...
 - convoy:TRIM10...

PVD Panel: This panel shows a 'Side Relationships' table. The table has columns for 'Side' and 'Relationship'. The rows represent the relationships between the sides: AI-Jahari, Coalition, and Insurgents.

	AI-Jahari	Coalition	Insurgents
AI-Jahari	FRIENDLY	SUSPECT	SUSPECT
Coalition	SUSPECT	FRIENDLY	HOSTILE
Insurgents	SUSPECT	HOSTILE	SUSPECT

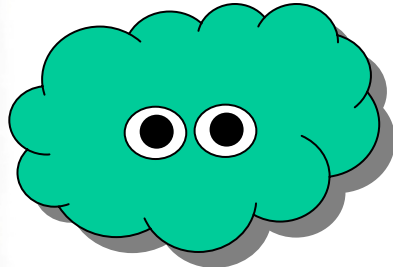
Below the table, there are additional relationship options: NEUTRAL and FRIENDLY.

Variable fidelity

- Support for variable levels of fidelity makes it possible to tailor the simulation in order to maximize satisfaction of diverse use cases.
- Accurately and effectively represents activities within the Army warfighting functions:
 - Intelligence
 - Movement and maneuver
 - Fire support
 - Protection
 - Sustainment
 - Command and control

Environmental Runtime Component (ERC)

Obscurants / contaminants

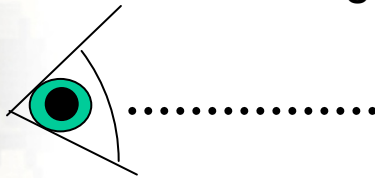


Features

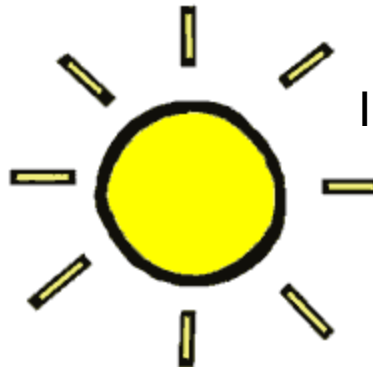


Weather

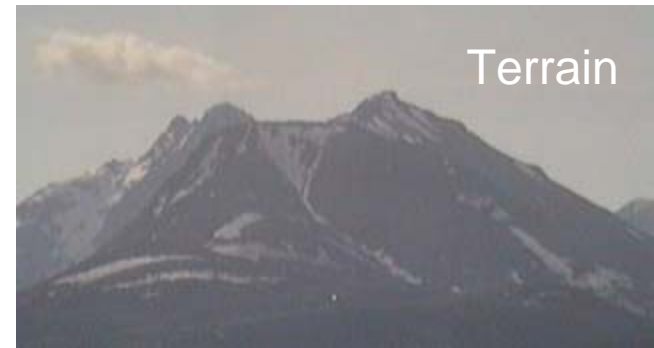
Line of Sight



Illumination



Terrain

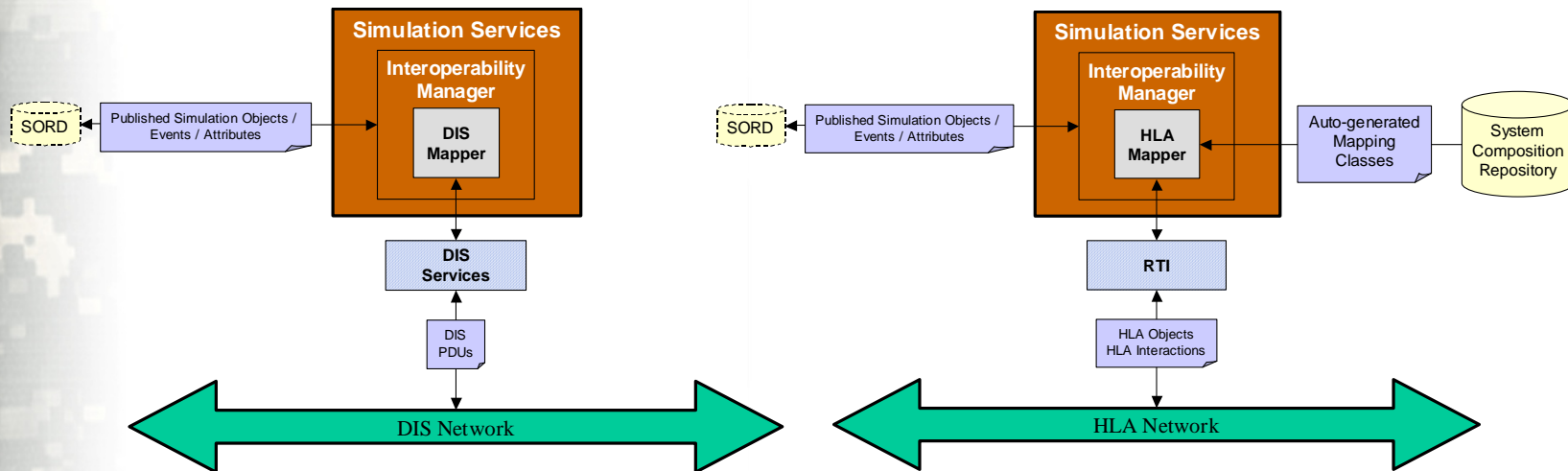


Distributed Simulation

- OneSAF is a distributed simulation system.
- Workstations may be assigned different computational roles, such as:
 - Simulation core (SimCore)
 - Management and Control Tool interface
 - C4I adapter
- Distributed design allows:
 - Individual workstations to balance loads
 - System recovery of individual workstations without interrupting a simulation in progress

Interoperability

- OneSAF instances interoperate via:
 - Internal protocols
 - OneSAF messaging services
 - C4I Adapter
 - External protocols
 - HLA
 - DIS



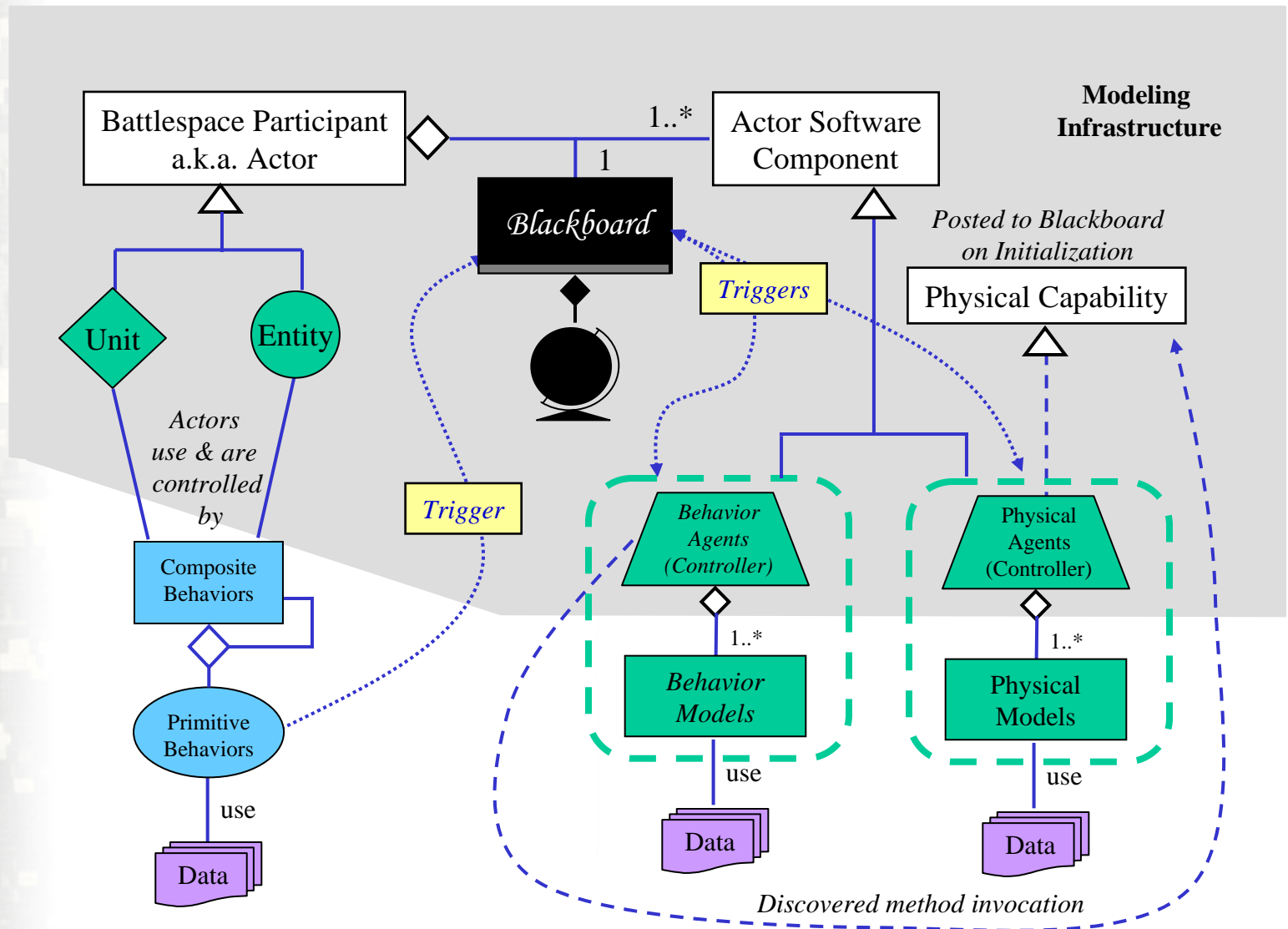
Overview Summary

- OneSAF is a highly configurable suite of systems for the generation of semi-autonomous forces.
- Simulation behaviors and models are data driven to the maximum extent possible, to support customization and “what-if” analysis
- System capabilities are exposed by powerful visual tools.
- High degree of interoperability through DIS and HLA, and usage of common formats including XML for state.

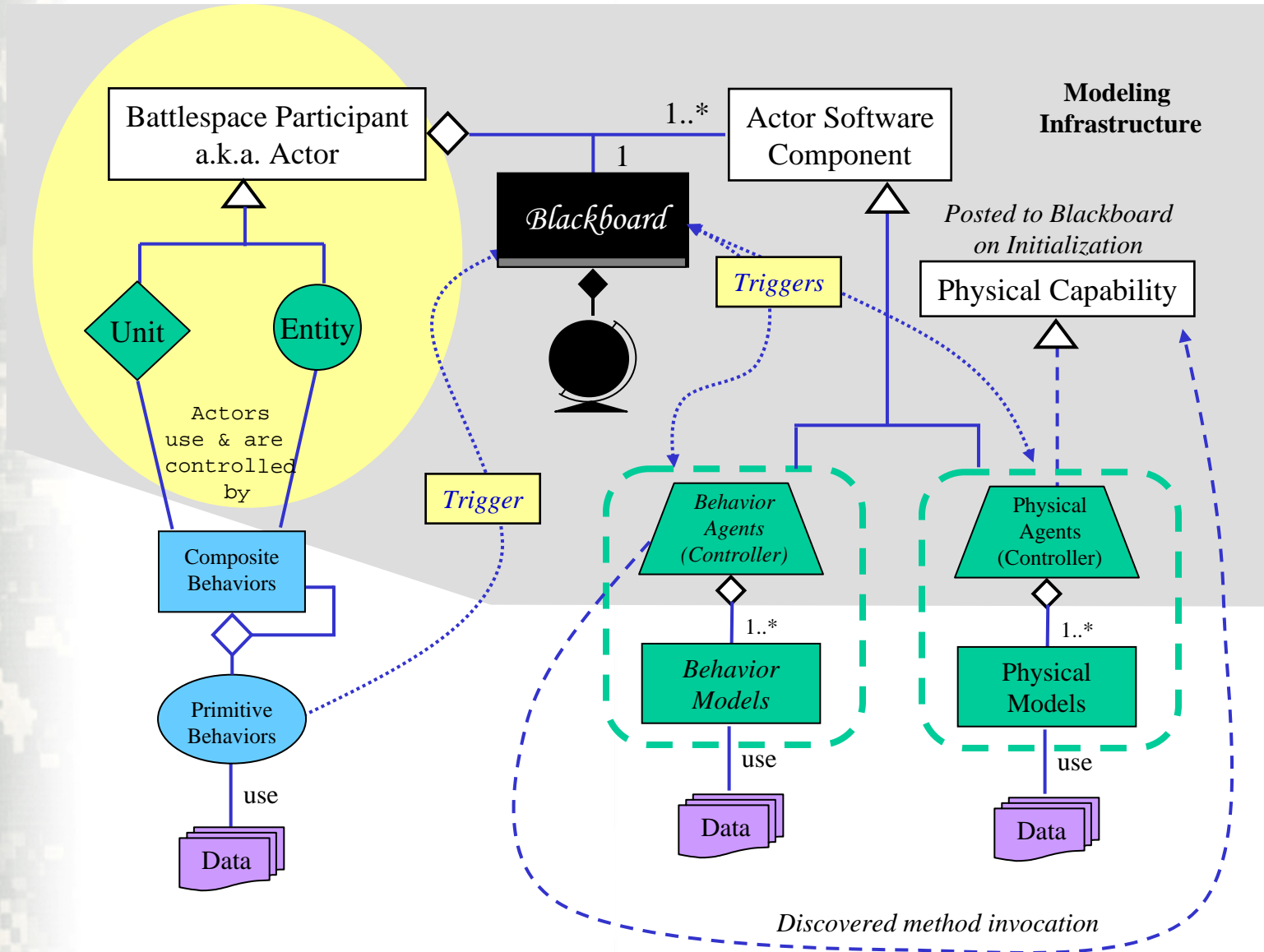


Introduction to the OneSAF architecture

The OneSAF Agent Architecture



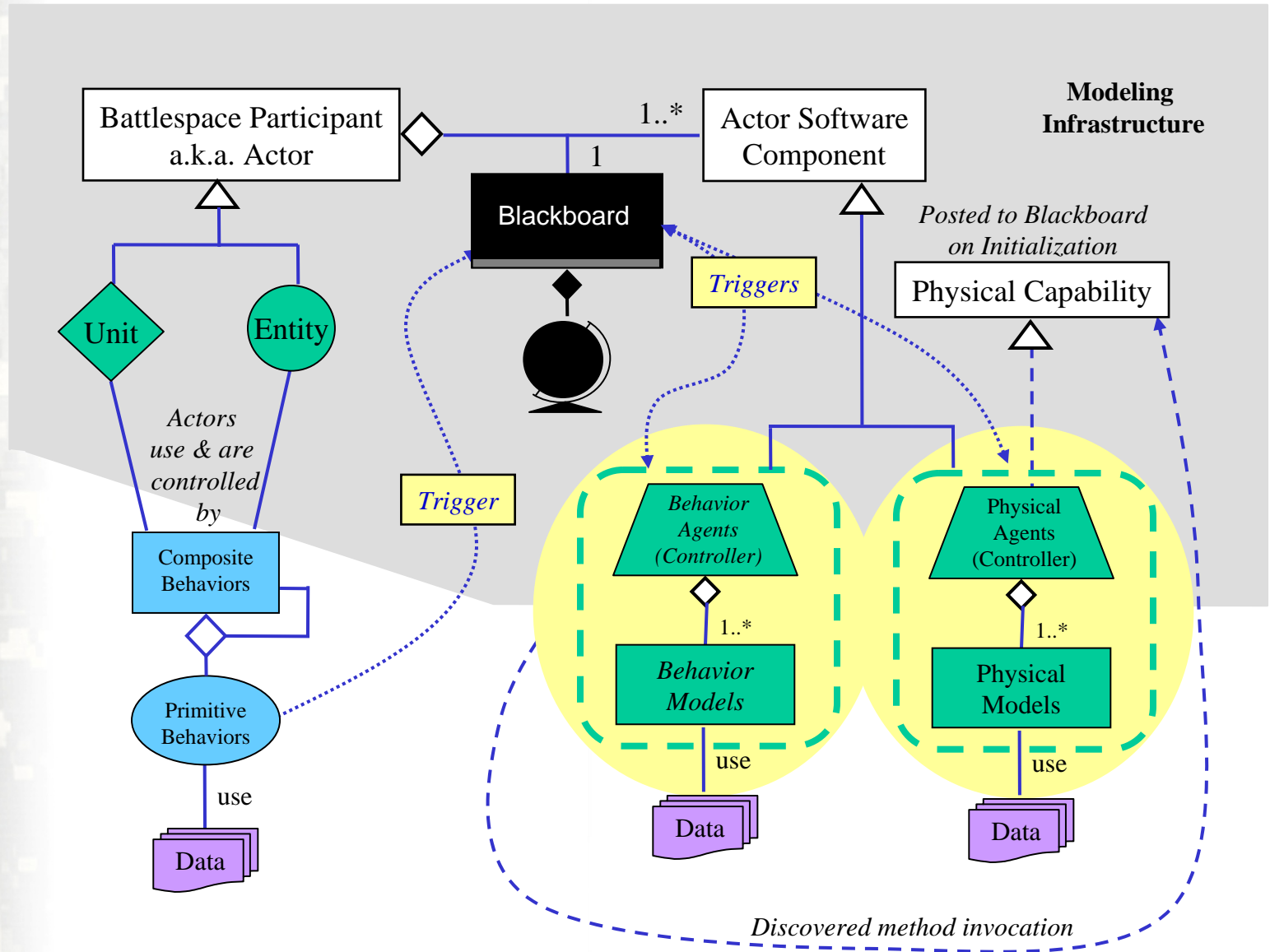
Battlespace participants (actors)



Battlespace participants (actors)

- Actor:
 - An entity or unit
 - A simulated thing that can be instantiated in a battlespace and has a location
- Entity:
 - The smallest discrete, stand-alone actor.
 - Implemented as a composition of physical components and behavioral components.
- Unit:
 - An organized collection of actors and their capabilities.
 - Alternatively ... “a collection of actors”
 - A simulation object representing the combined command and control of a collection of subordinate actors (entities and sub-units).
 - Alternatively ... “a collection of components”

Components



Components

- Agents:
 - Behavioral agents provide command and control capabilities, such as planning, plan execution, and situation assessment.
 - Physical agents are the “middlemen” between behaviors, the physical world, and physical models.
- Models:
 - Behavioral models answer behavior agents’ questions and represent the reasoning of agents.
 - Physical models provide physical capabilities, such as mobility, weapons, vulnerability, sensing, and communications. They represent the effectors and perceptors of simulated platforms and the physics of the simulated world.

Example Agents

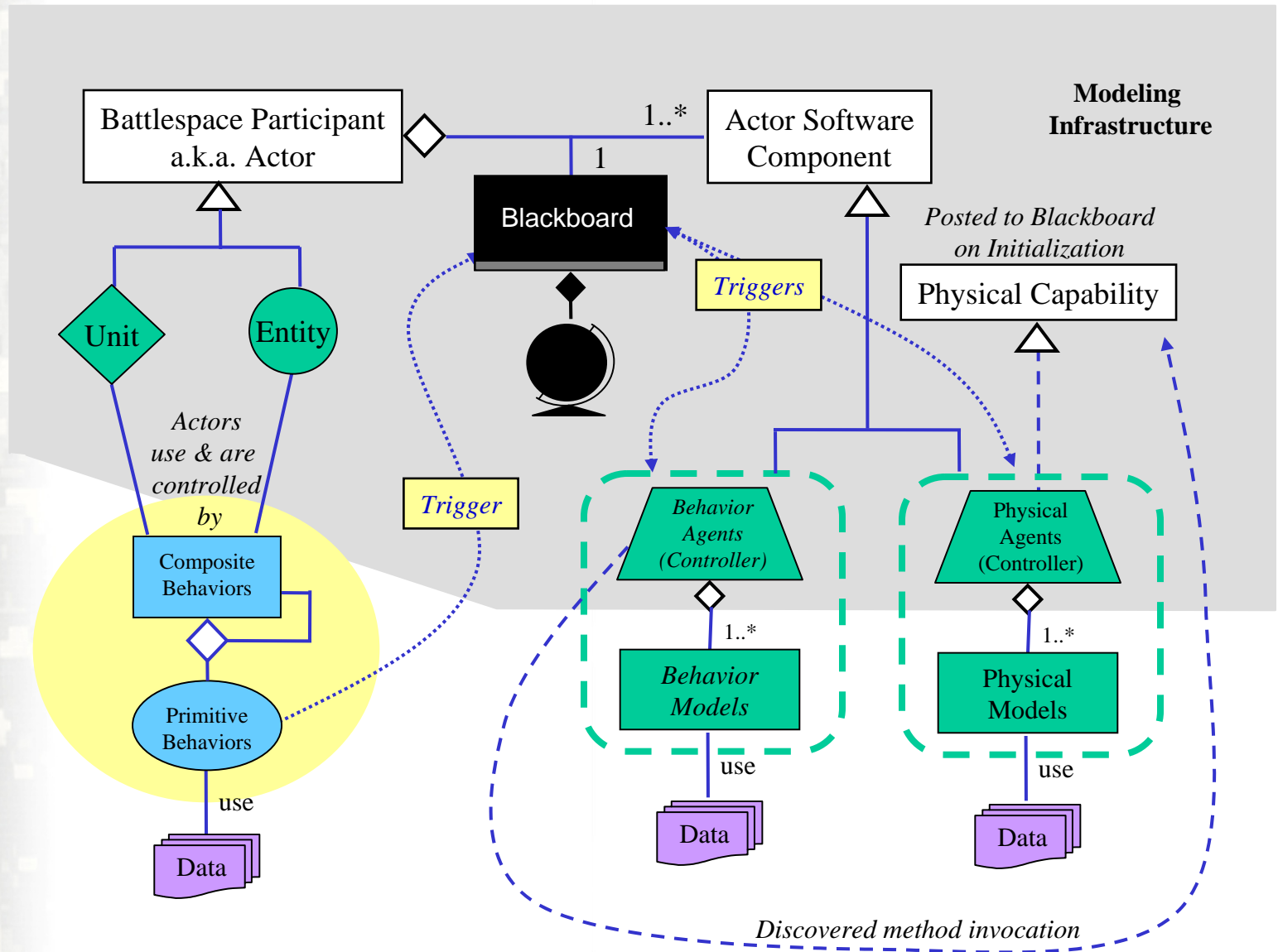
- **Behavioral Agents:**

- Command
- Scheduler
- Intel
- Message
- Operations
- Driver
- Direct Fire Weapons Controller
- Fire Direction Center
- Caller For Fire
- ADA Target Handoff

- **Physical Agents:**

- Weapon
- Radio
- Sensor
- Mobility
- Transport
- Vulnerability

Behaviors



Composite behaviors

- Composite Behaviors:
 - Represent tasks and missions
 - Are composed of primitive and other composite behaviors
 - Are created with the Behavior Composer.
- Temporal organization:
 - Sequential
 - Parallel
- Conditional Branching
- Looping

Primitive behaviors

- Primitive behaviors:
 - Provide chunks of functionality from which more complex behavior models are built.
 - Are parameterized with inputs, and may have outputs
 - Interact with behavioral agents.
- Are implemented as Java classes, with an XML wrapper to allow composition

Office



Content carriers

- Event
 - Information object that is sent from an actor to the external world over the network.
 - Events are “sent” on the EventRouter.
 - Also referred to as a “Simulation Event.”
- Trigger
 - Information object sent internal to an actor. May be sent/received by agents and behaviors.
 - Triggers are “posted” on the blackboard.
 - Also referred to as a “Command.”
- Directive
 - A special event that can also function as a trigger.
 - The Directive may be “sent” or “posted.” When it is received as an event, it will automatically be posted as a trigger on the blackboard.
 - Also referred to as an “Intervention” (on the MCT).
- Fact
 - Information object that is stored in the blackboard.
 - Facts may be simple “Information Facts” that store information or they may be “Fuzzy Facts” that can evaluate its stored information and modify it over time.
- Message
 - A class of events used by the communication framework.

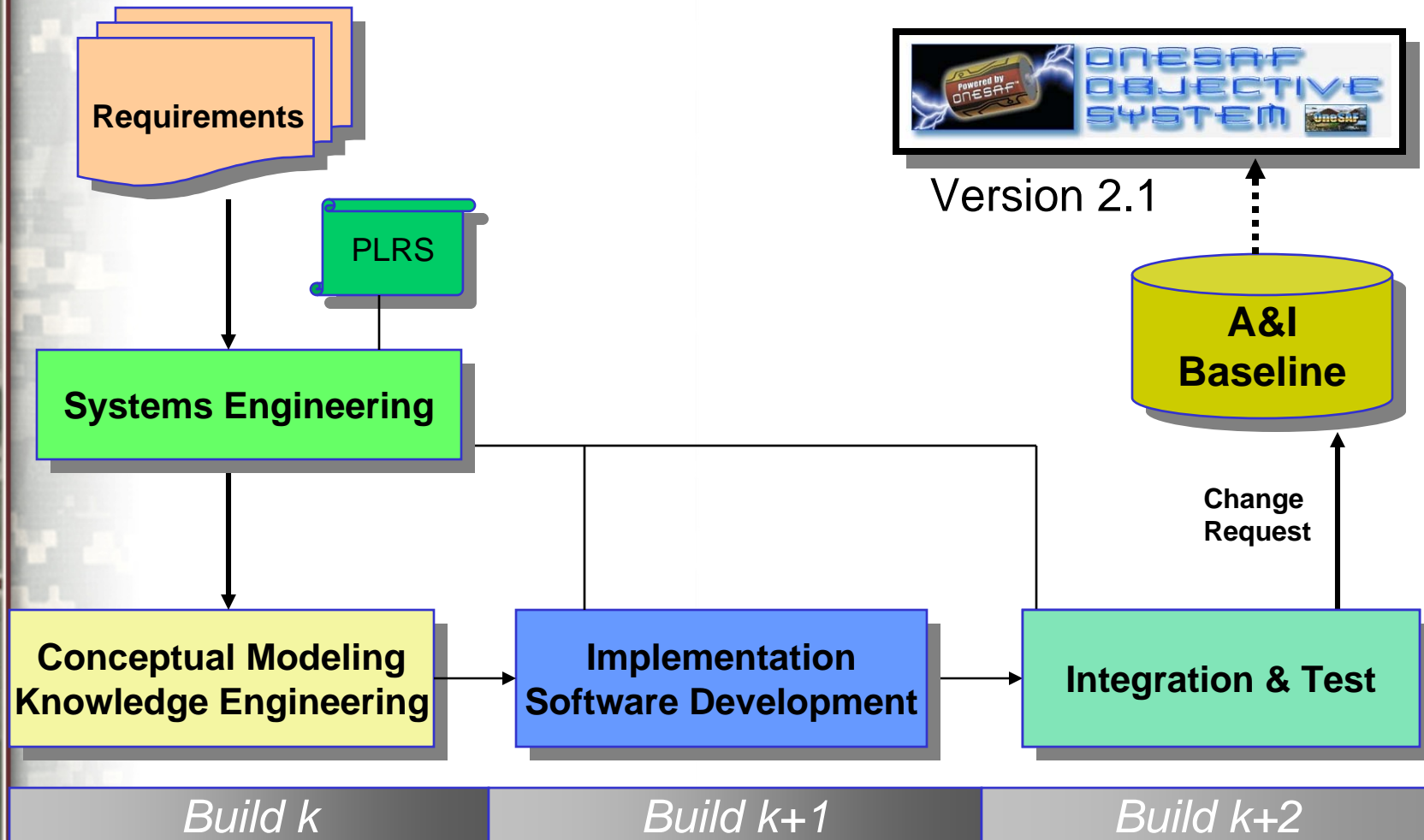
The Blackboard

- The Blackboard is an Agent control mechanism:
 - Agents subscribe to triggers on the blackboard.
 - When the blackboard receives a trigger, the it notifies that trigger's subscribers (and only those subscribers).
- Blackboard:
 - “World Model” for an actor containing perceived truth.
 - » Send and receive (internal) triggers
 - » Contains facts
 - » Access physical capabilities
 - One instance per actor; cannot be shared with other actors.

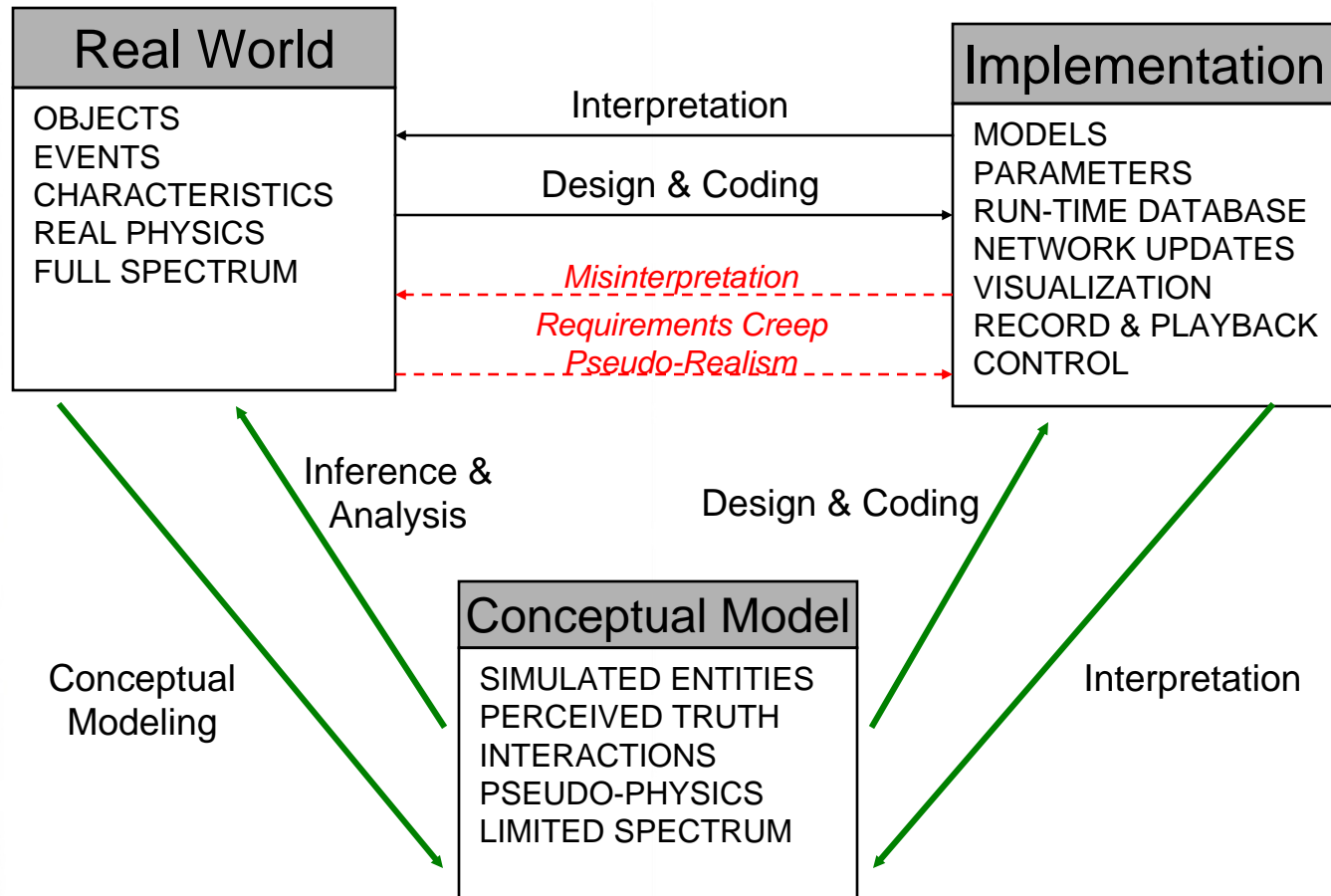


OneSAF processes

Capability Development Process Flow



What problem does Conceptual Modeling address?



Simulating the real world is difficult and error prone.

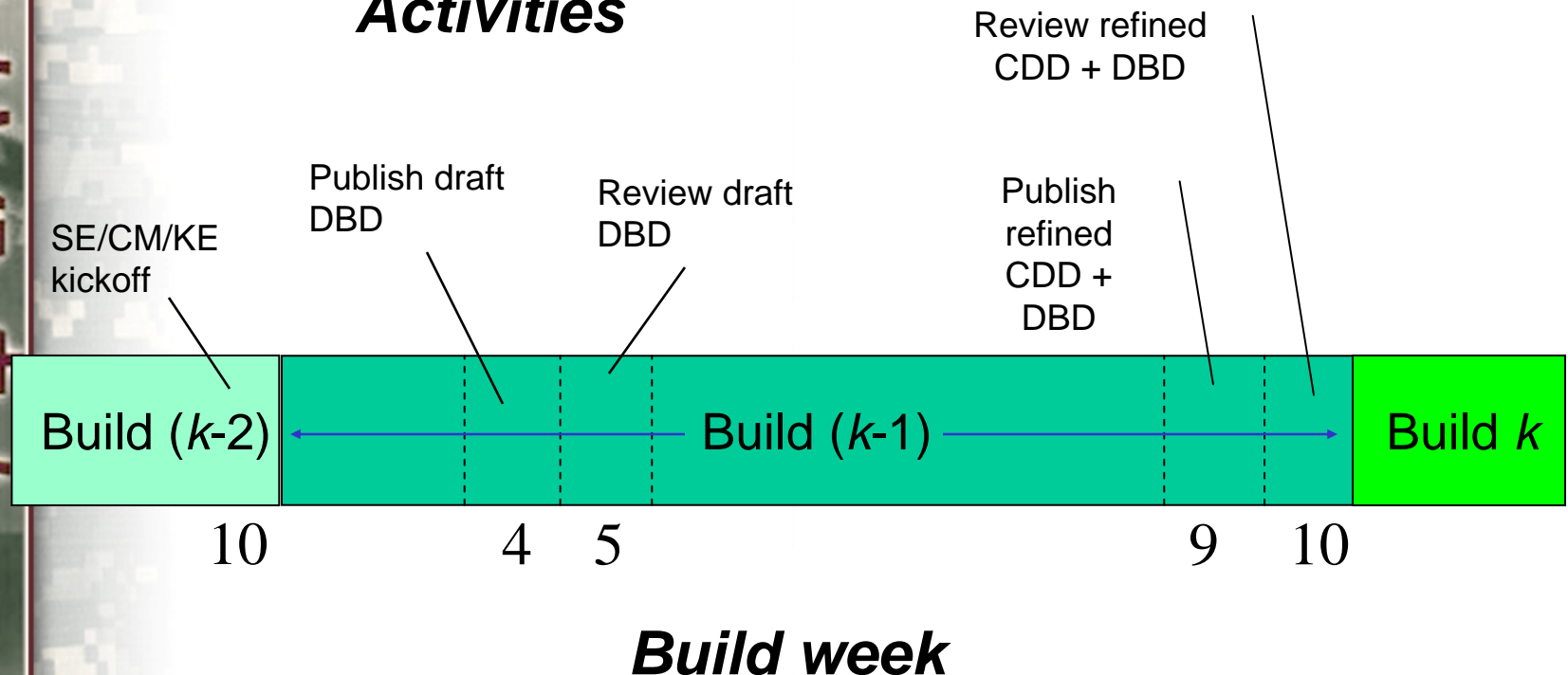
A well-conceived, consistent intermediate model eliminates many problems by providing a model of the battlespace usable by all participants (customer, domain expert, developer, and user).

Conceptual Modeling and Knowledge Engineering Process

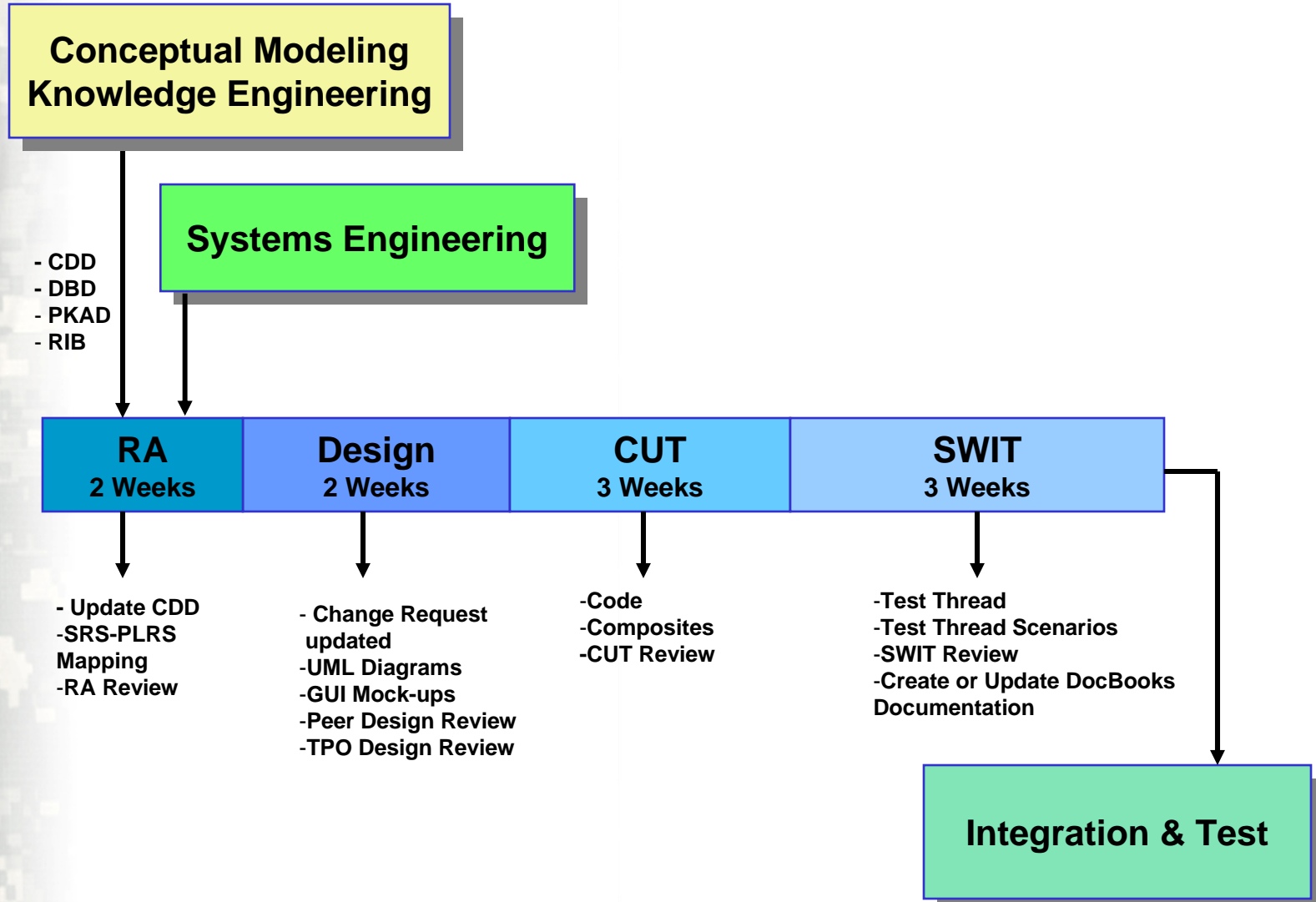
Main products:

- o Capability Description Document (CDD)
- o Domain Behavior Description (DBD)
- o Physical Knowledge Acquisition Document (PKAD)

Activities



Implementation Build



የኢትዮጵያ ፌዴራላዊ ዲሞክራሲያዊ ሪፐብሊክ

General handover criteria

- *Integration with OneSAF:* It is expected that handovers are submitted for the purpose of integration into OneSAF, and that they are ready for integration.
- *Development baseline:* Capabilities must be developed against the most recent generally available release.
- *Architectural compliance:* Handover products must comply with architectural and design guidelines, protocols, language compatibilities, code structures, and documentation standards.
- *Testing:* Capabilities must be tested during development to ensure that they operate as designed are not injurious OneSAF.
- *Classification:* At present, all handover products must be unclassified.

Co-developer handover process

- To be considered for inclusion with a baseline release, co-developers participate in several activities:
 - Registration (co-developer)
 - Initial review (OneSAF Architecture and Integration)
 - Handover package submission (co-developer)
 - Engineering Change Control Board review
 - Peer review (OneSAF Architecture and Integration)
 - Baseline integration and test (OneSAF Architecture and Integration, with co-developer participation desired)
- Required artifacts:
 - Source code and data
 - Design documentation (use cases, sequence and class diagrams)
 - Domain documentation (PKAD / DBD / CDD)
 - Usage instructions
 - Unit tests and system test cases
 - Architectural compliance assessment



Questions?