SYSTEMS ENGINEERING RESEARCH CENTER

SERC TALKS

WELCOME

"What is the role of a Reference Architecture in research and development of autonomous and/or cooperative systems?"

October 2, 2019 | 1:00 PM ET

Dr. Dave Jacques, Lt-Col (USAF-Ret) Professor of Systems Engineering,
Air Force Institute of Technology
David.Jacques@afit.edu

- Today's session will be recorded.
- An archive of today's talk will be available at: www.sercuarc.org/serc-talks/ as well as on the SERC YouTube channel.
- Use the Q&A box to queue questions, reserving the chat box for comments, and questions will be answered during the last 5-10 minutes of the session.
- If you are connected via the dial-in information only, please email questions or comments to SERCtalks@stevens.edu.
- Any issues? Use the chat feature for any technical difficulties or other comments, or email SERCtalks@stevens.edu.

Use of Reference Architectures for Development of Autonomous Systems

October 2019

Dr. David Jacques
Mr. Jeremy Gray
Air Force Institute of Technology

Problem Statement

AFIT's Experimental Research Environment for Autonomous and Cooperative Systems

- Multiple projects spanning multiple researchers and departments and multiple student production cycles
- Too often, lack of continuity across projects
- Need to accelerate domain learning associated with the target vehicle agents
- Multi-disciplinary prototyping projects are challenging to execute within single student cycle – loss of knowledge between projects
- Minimal SE and system integration knowledge among non-SE students

Reference Architectures

- **Definition:** “an authoritative source of information about a specific subject area that guides and constrains the instantiations of multiple architectures and solutions”.*
- An RA is typically company/consortium or domain focused, captures a shared understanding across multiple products, and are based on concepts proven in practice
- **Role:****
  - Providing common language for the various stakeholders;
  - Providing consistency of implementation of technology to solve problems;
  - Supporting the validation of solutions against proven Reference Architectures;
  - Encouraging adherence to common standards, specifications, and patterns.

*Office of the Assistant Sec. of Defense, 2010
**Cloutier, et.al., *Systems Engineering,* 2010

Reference Architecture Examples (1)

- Meisson and Volsard* describe an RA for an Early Warning System (EWS) for disaster management
  - Provides a common understanding of functional elements
  - Serves as the basis for design and implementation, to include future evolution and reuse of modules
- Kruize, Wolfert et.al.** describe an RA to guide and accelerate development of Farm Software Ecosystems
  - RA defines a particular approach and configuration for connecting generically defined information technology elements associated with precision agriculture.
  - A set of functional requirements for a farm software ecosystem, and class diagrams defining the relationships between the components and actors

**International Conference on Intelligent Networking and Collaborative Systems*, 2010
***Computers and Electronics in Agriculture,* 2016

Reference Architecture Examples (2)

- Branscomb, Paredis et.al.* describe an RA as a component of a comprehensive Vehicle Architecture Modeling Framework
  - Provides a logical decomposition of the vehicle into its subsystems, and defining the various interfaces between those subsystems, and between the vehicle and the driver
  - Provides a common understanding of functional elements
- Behere, Torngren et.al.** describe an RA for cooperative driving
  - Guidelines for instantiation of the RA into a prototype system are provided
  - Application to a cooperative adaptive cruise control system for heavy duty commercial trucks

**Procedia Computer Science*, 16 (2013)
***Journal of Systems Architecture*, 59 (2013)

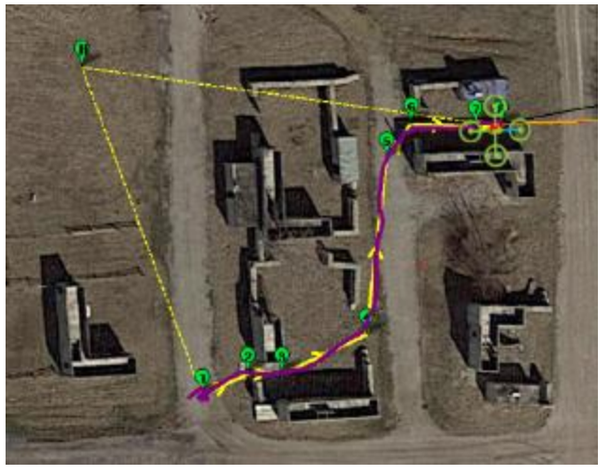
A Cubesat Reference Model

- Kaslow, Ayres, et.al.* have published a series of papers on the development of an MBSE Reference Model for development of Cubesats
  - Partnering with the Object Management Group (OMG) and the International Council on Systems Engineering (INCOSE)
  - Help Cubesat developers proceed from conceptual need to on-orbit operations
  - CRM provides logical, reusable architecture elements to facilitate development of a physical architecture; does not provide COTS component building blocks to use within the physical architecture

**IEEE Aerospace Conference*, 2015, 2017

AFIT's Experimental Autonomy and Cooperative Control Research

- Three course graduate sequence for Small UAS conceptualization, design, build and flight test
  - Rapid prototyping (9 months) against mission problem
  - Prepares students for follow-on research requiring flight test
- Recent research project highlights
  - Auto-routing in an urban environment
  - Convoy following
  - Autonomous communication relay
  - Close formation keeping (swarming, auto-air refueling)
  - Cooperative wide area search and surveillance
  - Flexible behavior architecture for sUAS
  - Human control of autonomous agents
  - Space proximity operations



AFIT Reference Architectures

- Autonomous System Reference Architecture
  - Focused on research for new autonomy concepts
  - Domain agnostic
  - Abstracts out hardware details
- SUAS Reference Architecture
  - Focused on rapid prototyping for education and research
  - Focused on functional, physical definition
  - Growing use within AFIT's flight test activities

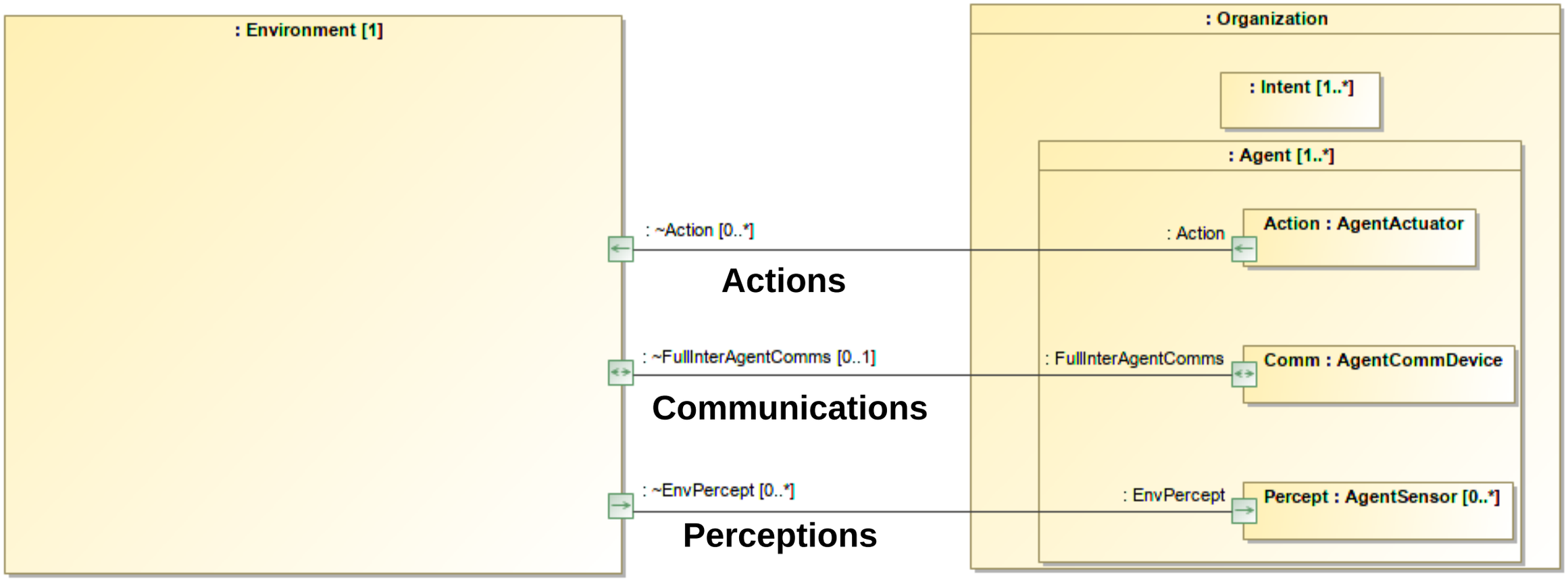
AFIT's Autonomous System Reference Architecture (ASRA)

- Objectives
  - Accelerate learning – especially in the areas of SE and system integration tools
  - Facilitate collaboration and reuse through use of a common architecture
  - Provide continuity across multiple research projects and successive student cycles
  - Provide support for rapid prototyping cycles
  - Provide scalability in scope
  - Provide flexibility in application and domain

AFIT's Autonomous System Reference Architecture (ASRA)

- Architecture Modeling
  - SysML model defined in Cameo System Modeler
  - Structure to support CONOPS, requirements, functions, hardware/software components, interfaces, messaging, etc.
  - Provides sufficient definition to support software and hardware development
- Modules
  - Model driven software for compatibility
  - Software modules to support common autonomy schemas
  - Linkages to component libraries and example agent builds
  - Flexible and extendable messaging ICDs
  - Previously tested hardware configurations
  - Interface definition for existing simulation environments

ASRA – High Level Overview



A high-level overview of important concepts, connections, and elements of the proposed Autonomy Framework

ASRA – Agent View

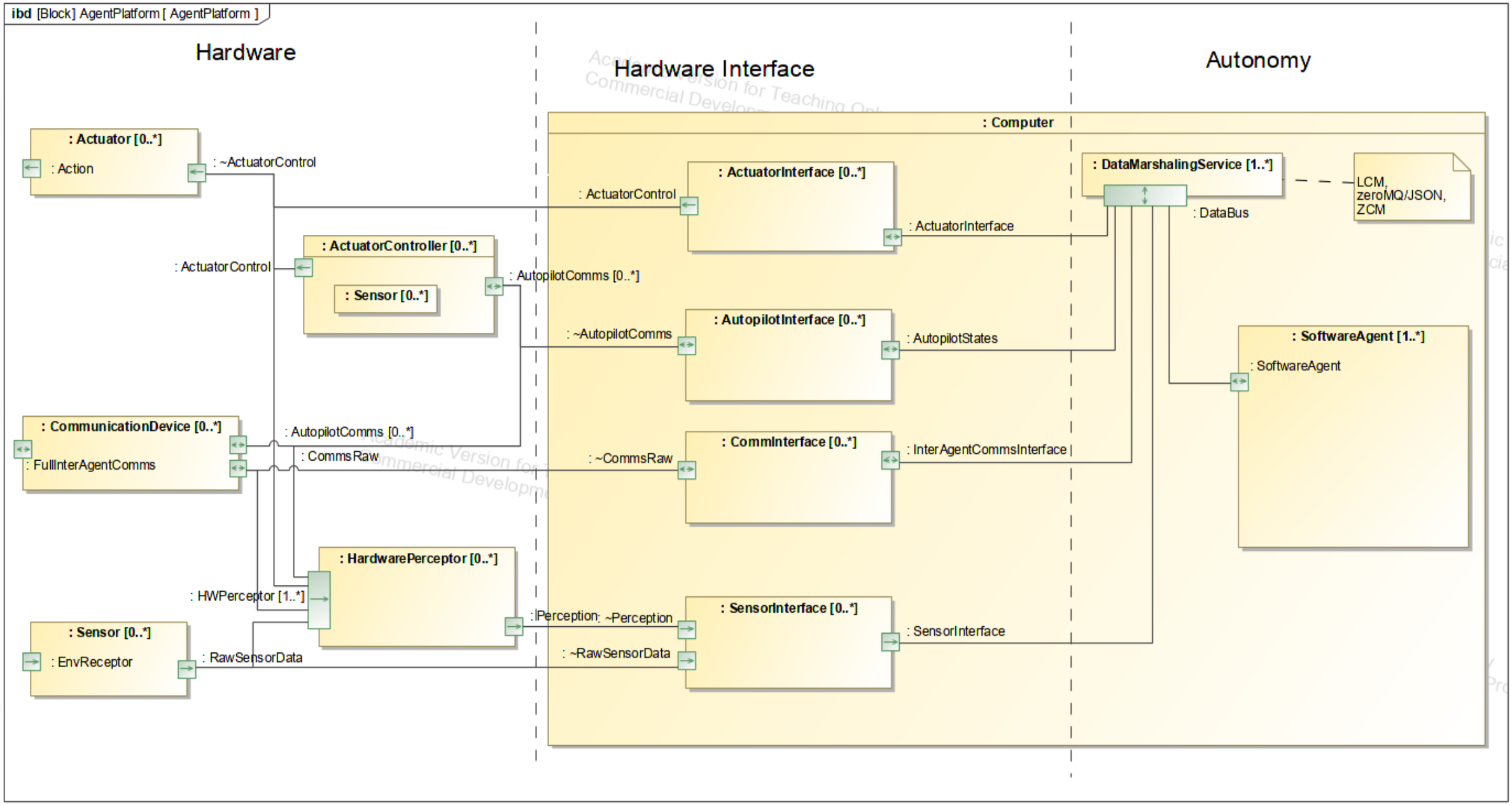
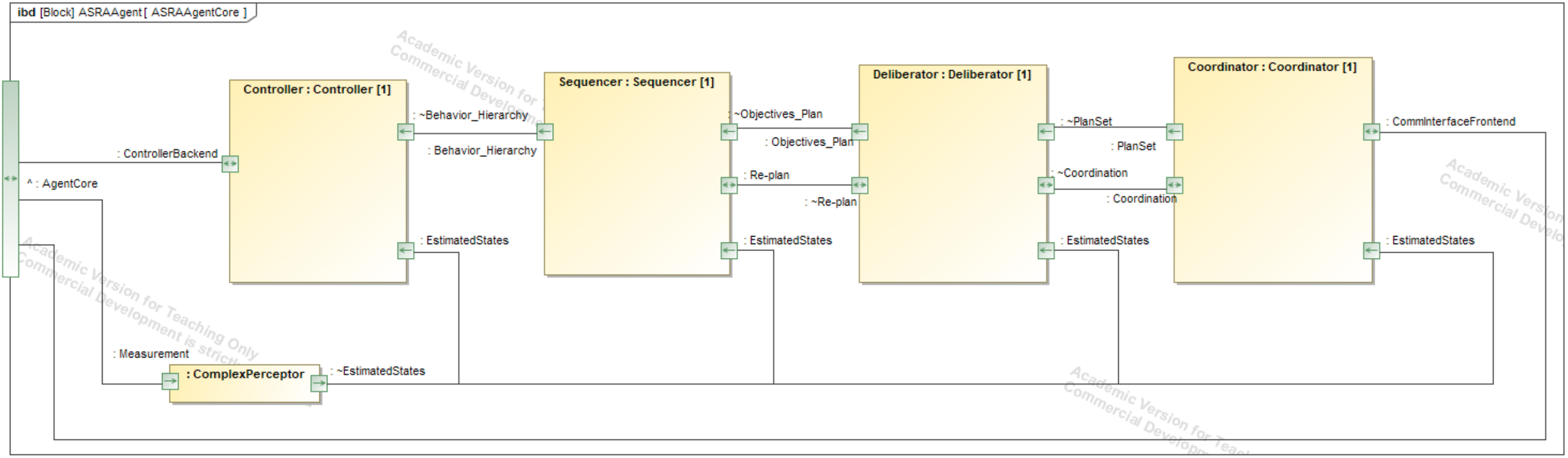


Illustration of the architecture of an agent platform within the ASRA

ASRA – Software Agent View

Agent Core utilizing Hybrid Architecture for Multiple Robots (HAMR)



Controller: Manages Behaviors and their implementation

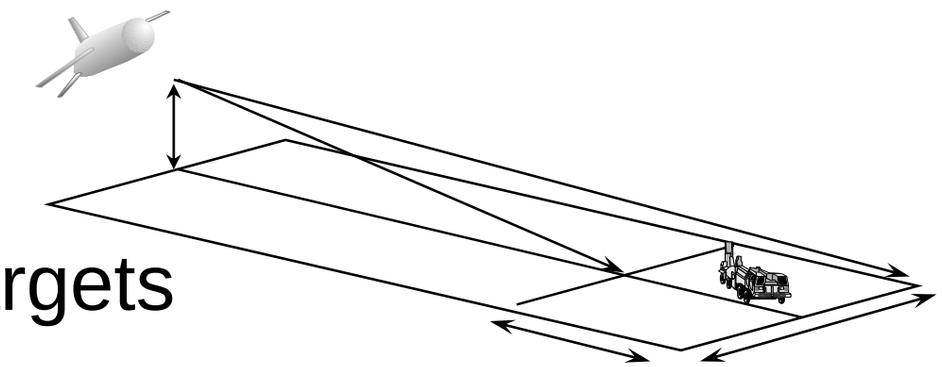
Sequencer: Enables and disables Behaviors where required; maintains internal state to alert Deliberator if new plan is needed

Deliberator: Performs high level reasoning tasks that includes task decomposition, task allocation, and planning

Coordinator: Expands on capabilities, allowing prioritization of tasks and managing negotiations for multi-agent applications

Application*: Wide Area Search and Surveillance (WASS)

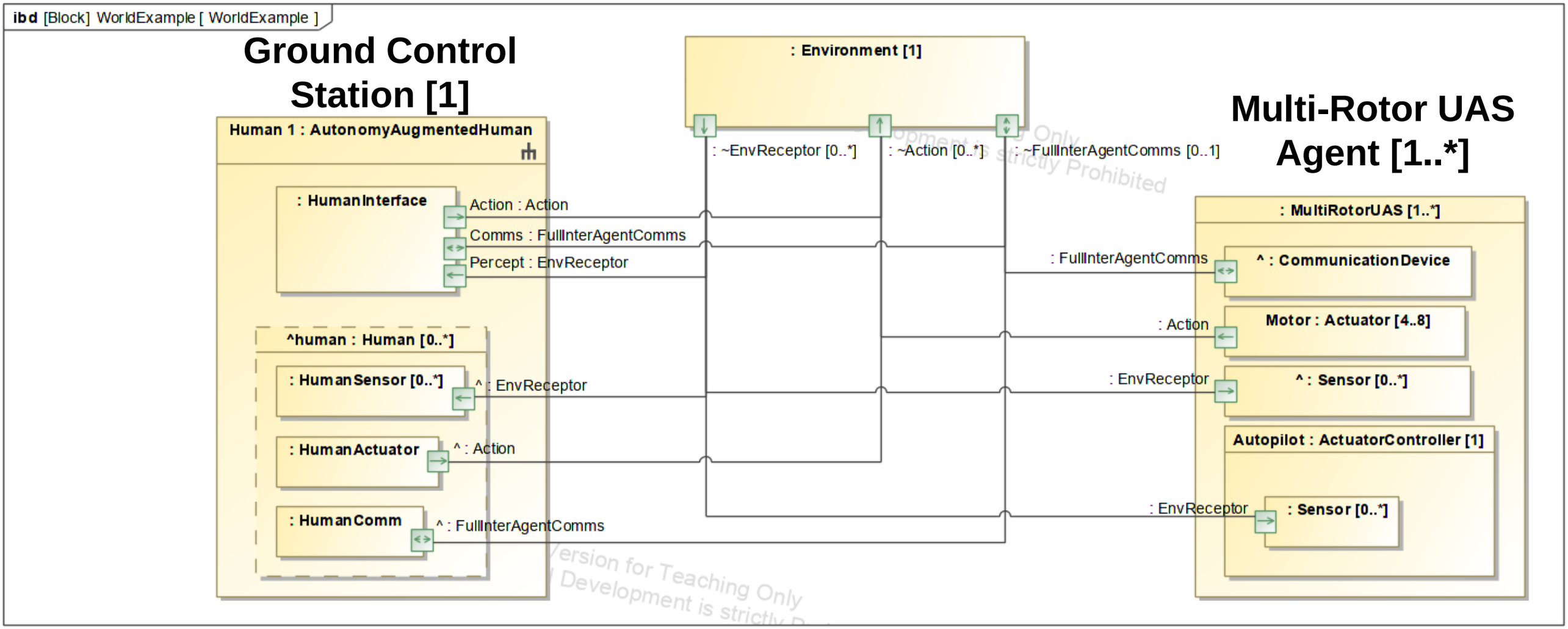
- Concept: Multiple autonomous and/or cooperating agents attempting to find targets in an uncertain environment, maintaining surveillance and/or engaging high priority targets
  - Imperfect target recognition capability
  - Imperfect navigation, geo-location of targets
  - Imperfect inter-agent communication
  - Limited agent mission time



- Example autonomous objectives: search, surveil, engage
- Possible cooperative decisions: redirect search, confirm target, expand surveillance, help engage target

*** Current MS thesis project conducted by Lt. Katherine Cheney and Lt. David King**

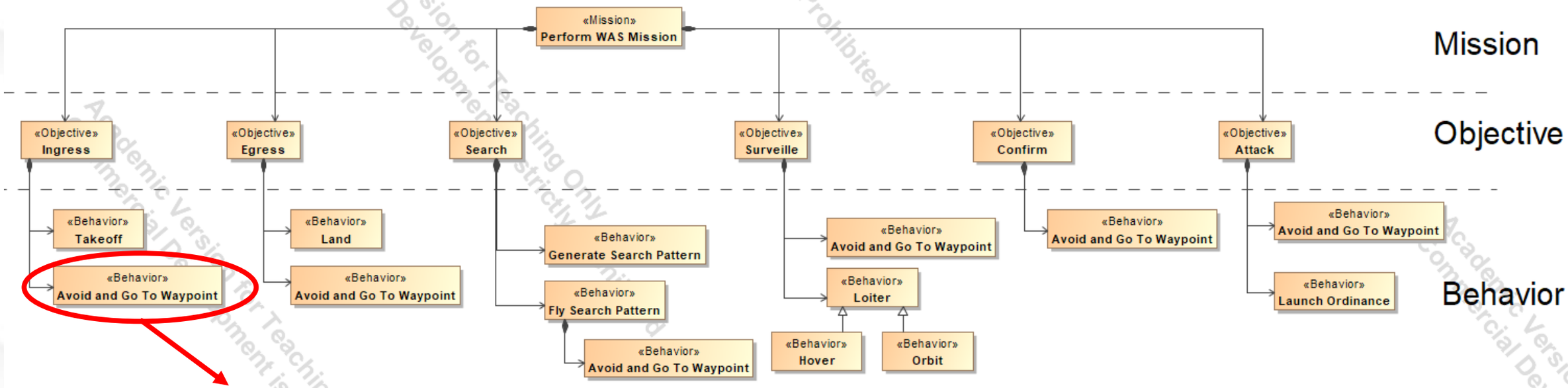
WASS SoS View

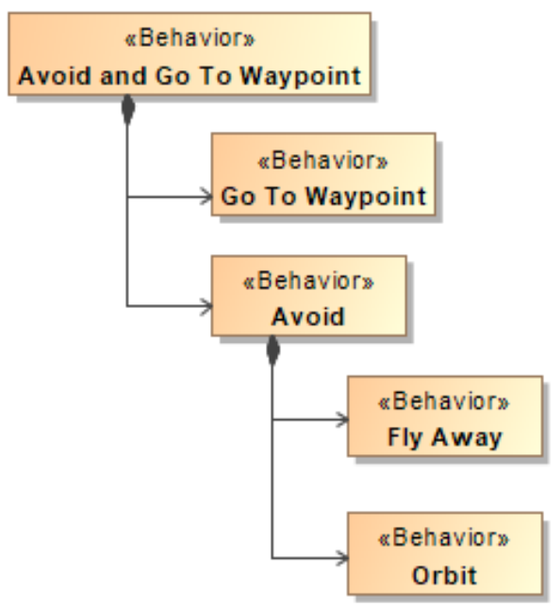


WASS Application

- Autonomous Objectives
  - Ingress/Egress
  - Search and detection
  - Target confirmation
  - Surveillance
- Cooperative Decision Logic
  - Initially implementing local decisions based on globally broadcast information
    - Planning to investigate centrally determined and/or negotiated decisions
  - Agents weigh benefits of continued search or surveillance against cooperative confirmation and surveillance requests
    - Can easily add task for engagement with limited resources

WASS Behavior Tree

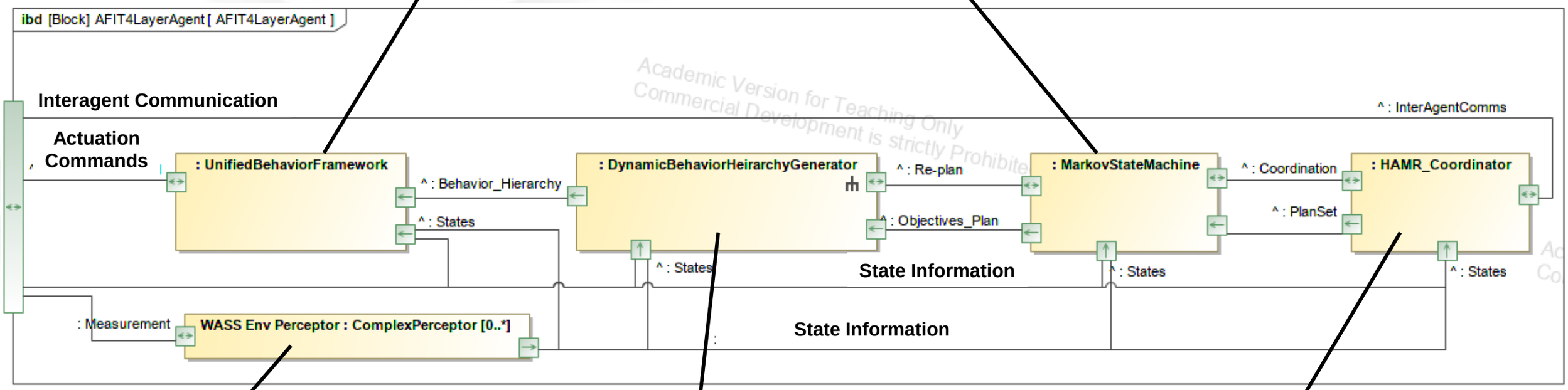




WASS Software Agent

Behavior Controller (UBF)

State Machine, local decision rules



Sensor and estimation models

Health & status monitoring, generates behavior hierarchies

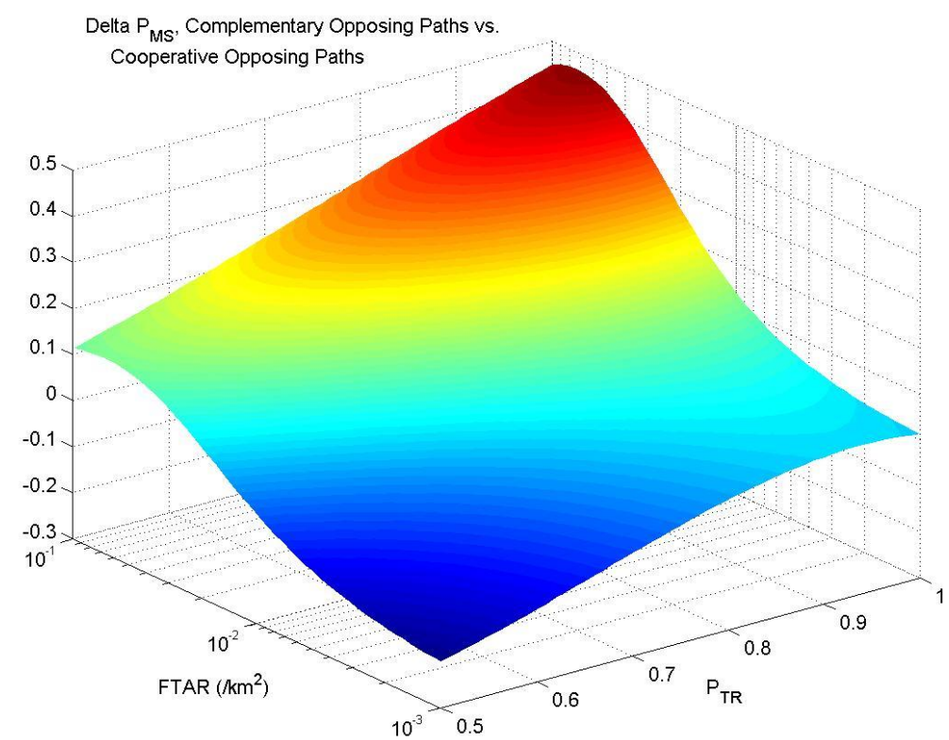
Interface with other agents, cooperative decision rules

Simulation Environment

- ASRA is simulator flexible
- Enables rapid test and evaluation cycles
- Ardupilot Software in the Loop (SITL) vehicle simulator
  - Generated behavior actions from UBF to SITL
  - Vehicle states from SITL to perceptors or perception states
- WASS software environment handles world model consisting of targets and non-target objects
  - Perceptors interface with this model (e.g. target sensor)
- Simulation environments being investigated
  - Microsoft Air Sim
  - Gazebo
  - AFSIM

Simulation Test Plan

- Design parameters
  - Area coverage rate (altitude, FOV, speed)
  - Sensor quality (altitude, FOV, resolution)
  - Target/false target density (surface area, # objects)
  - Task priorities (weights)
- Response variables
  - % targets detected, ID'd, or surveilled
  - False alarm rates
  - Area coverage
  - Target ID confidence
  - Responsiveness
  - Robustness



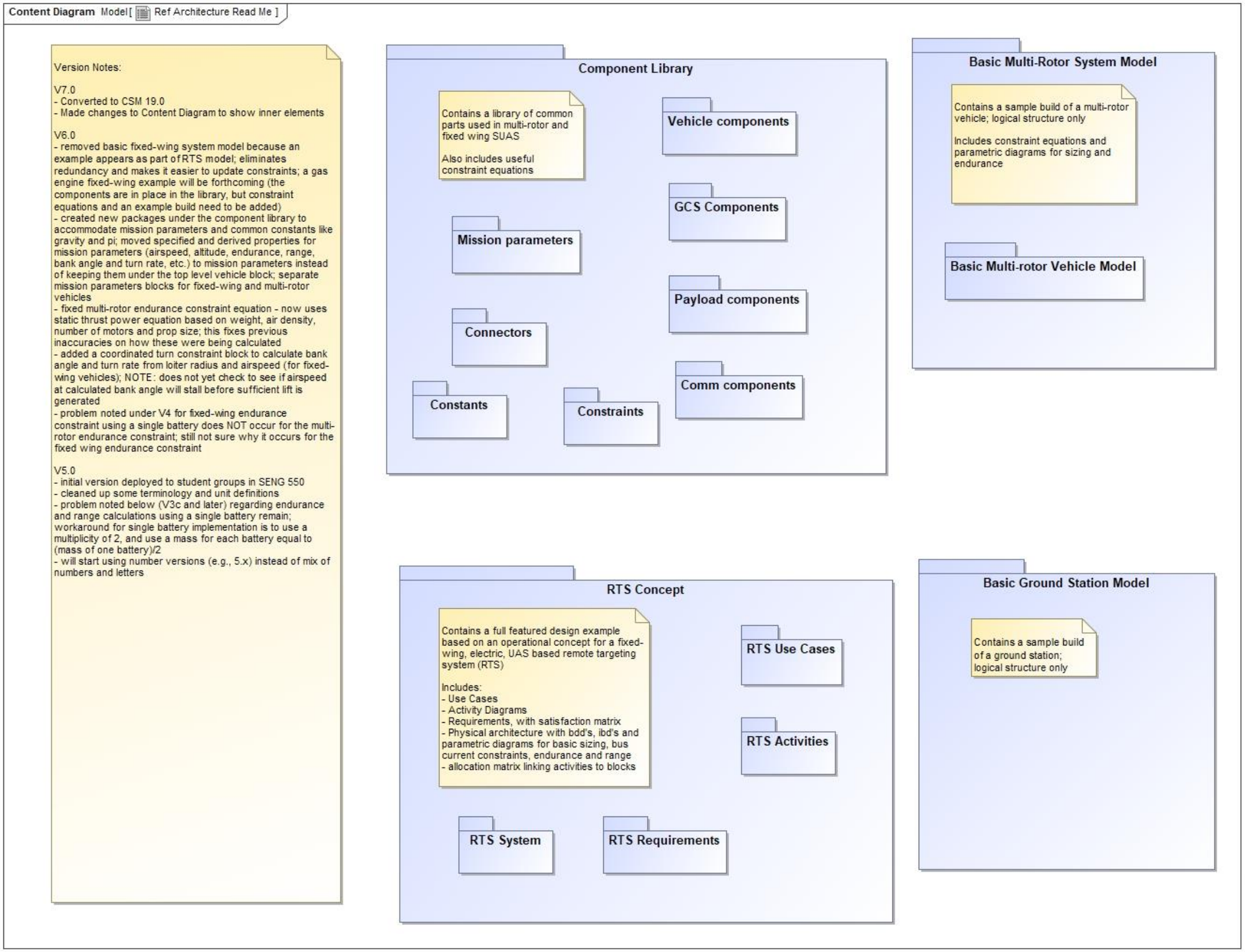
- Compare fully autonomous with cooperative concepts with varying levels of cooperation

Transitioning to Hardware Experimentation

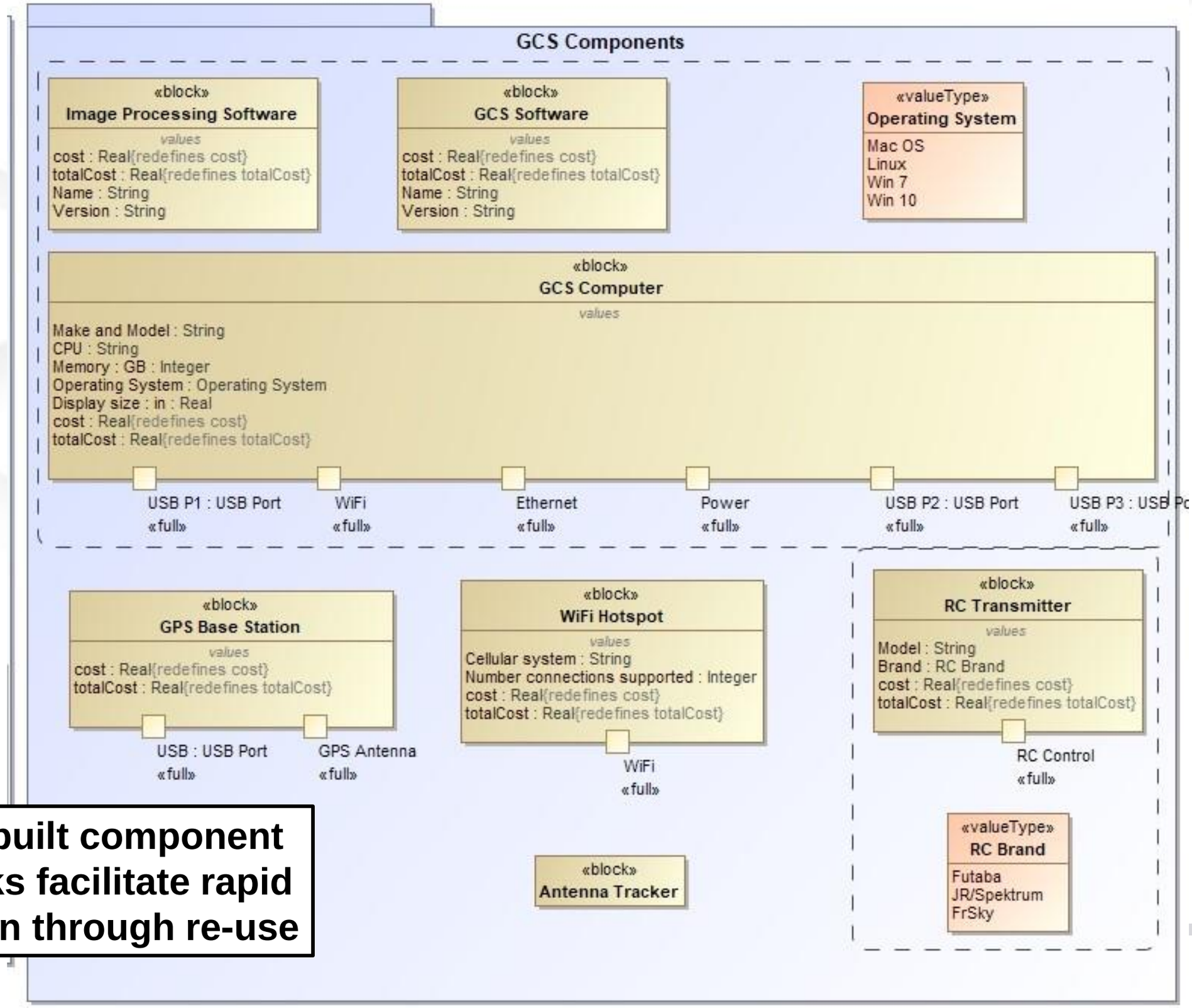
- Challenges associated with legacy approach
  - Extensive time spent building architectures at the expense of analysis
  - Inconsistent quality associated with architecture definition and requirements traceability
  - Analysis sometimes disconnected from architectural design
- Objectives of SUAS Reference Architecture
  - Improve student learning objectives in the SUAS track
  - Teach Systems Engineering to non-SEs
  - Capture extensive knowledge of componentry and typical configurations
  - Provide extensible basis for research designs

AFIT's SUAS Flight Test Program

SUAS Reference Architecture Home Page

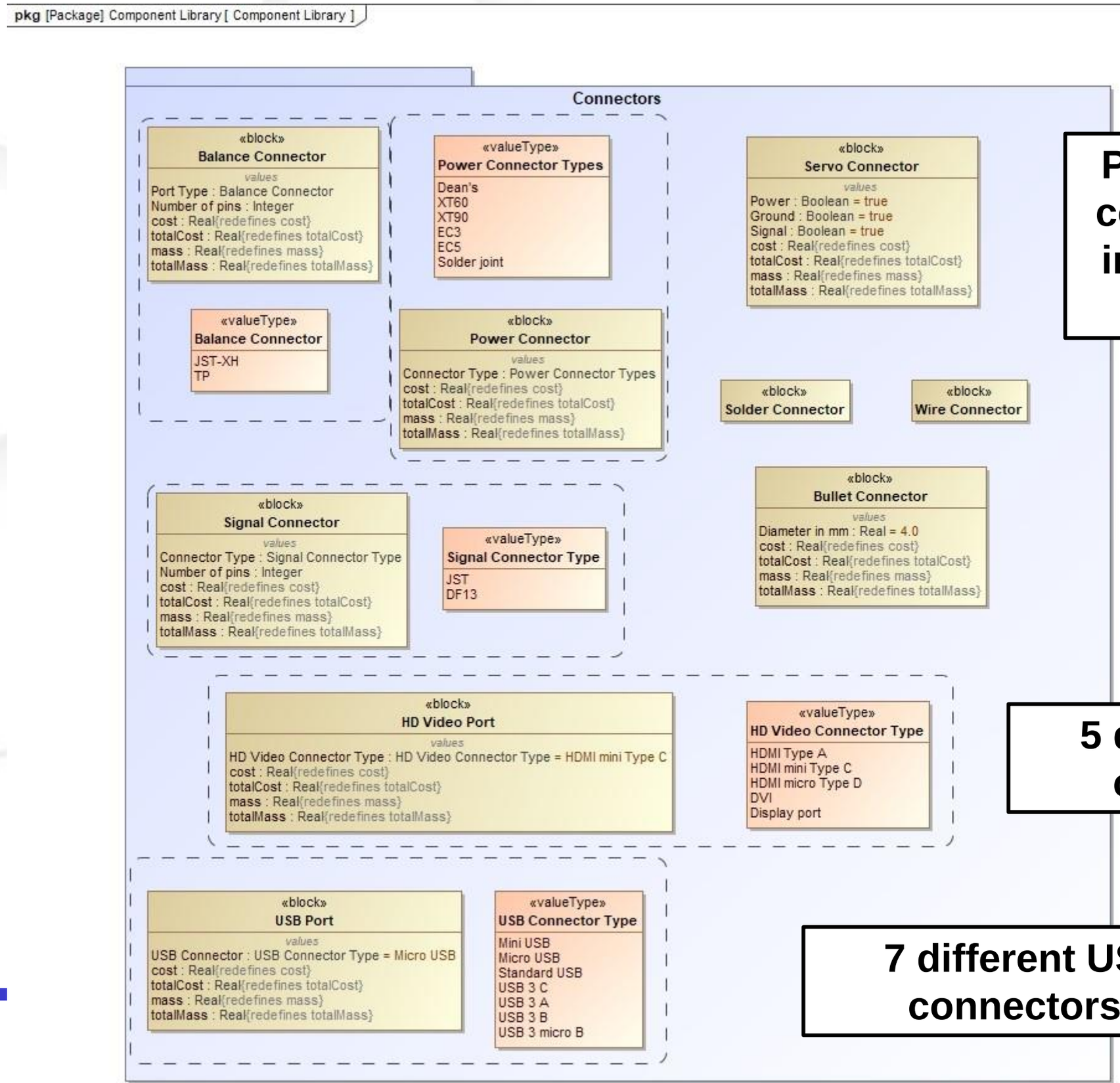


SUAS Reference Architecture Ground Control Components



Pre-built component blocks facilitate rapid design through re-use

SUAS Reference Architecture Connectors



Pre-typed ports and connectors facilitate integration of COTS components

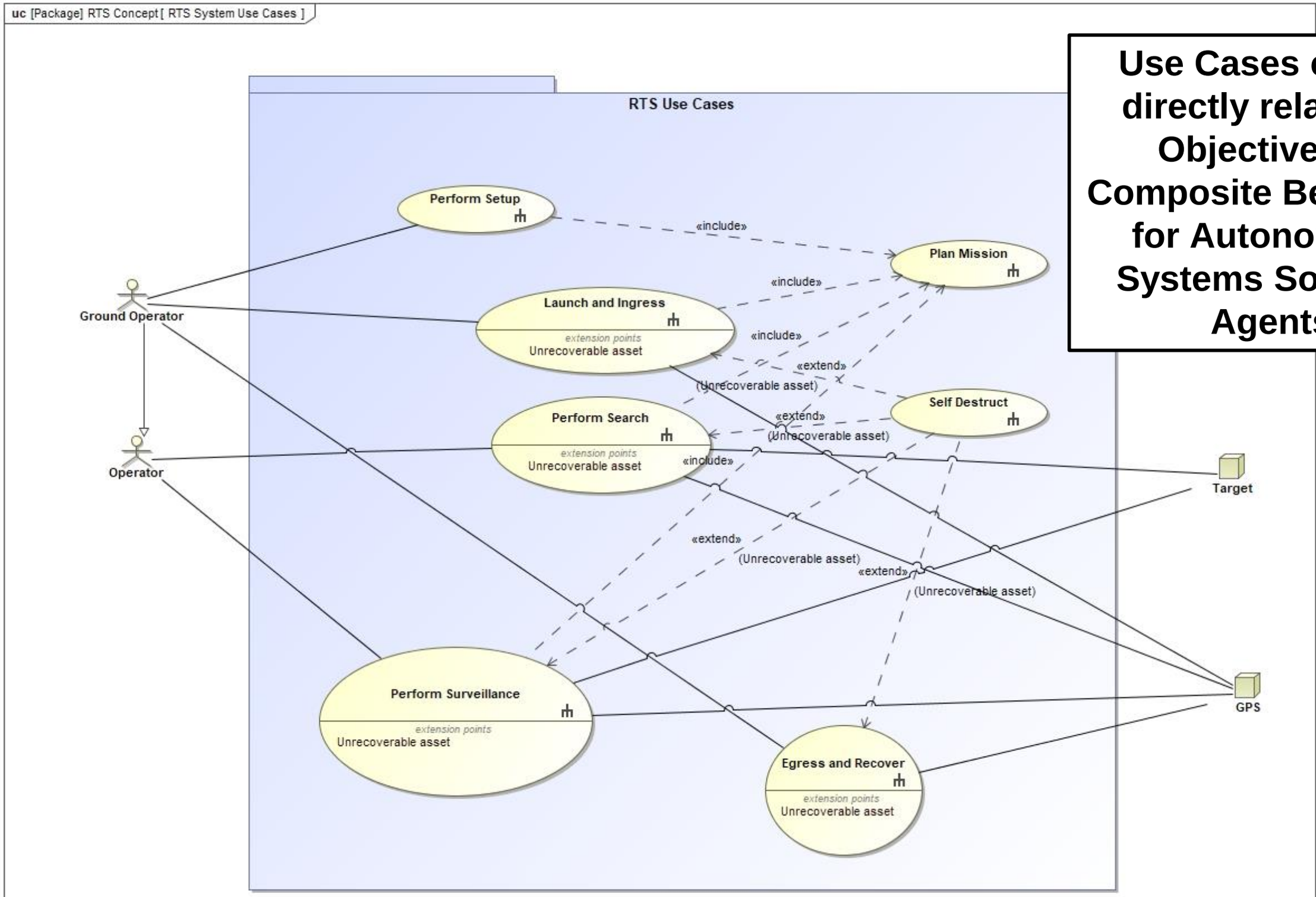
5 different HDMI connectors !

7 different USB connectors !

Using the SUAS Reference Architecture

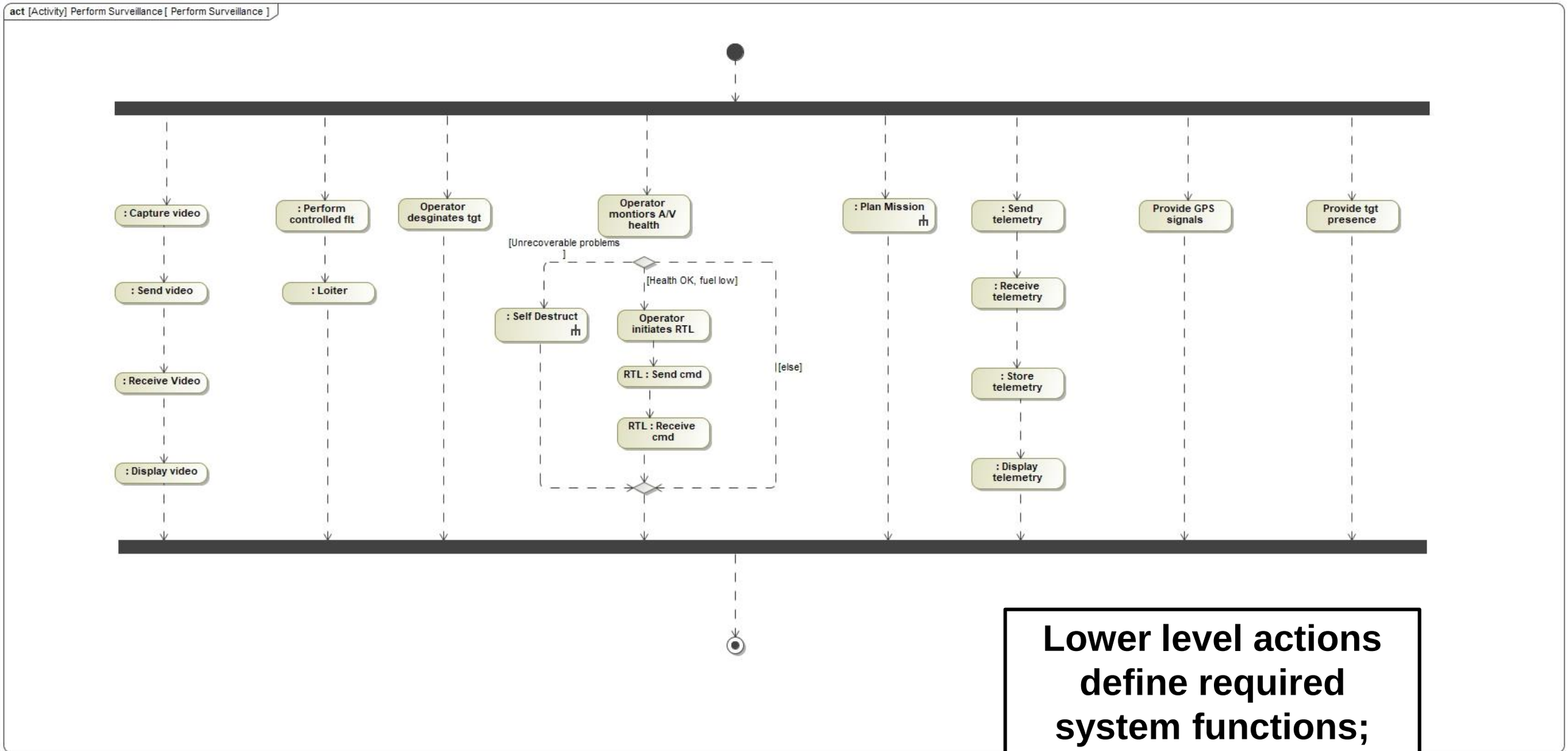
- Students are typically given a Draft CONOPS representing user's capability needs
- CONOPS scenarios are refined with Use Cases and requirements are defined
- Use Cases support functional decomposition, with traceability to requirements
- Functional allocation used to define a physical architecture (to include external systems and operators)
- Test cases established for verification and validation
- Architecture models are deliverables at PDR, CDR and SVR (final outbrief)

Remote Targeting System (RTS)



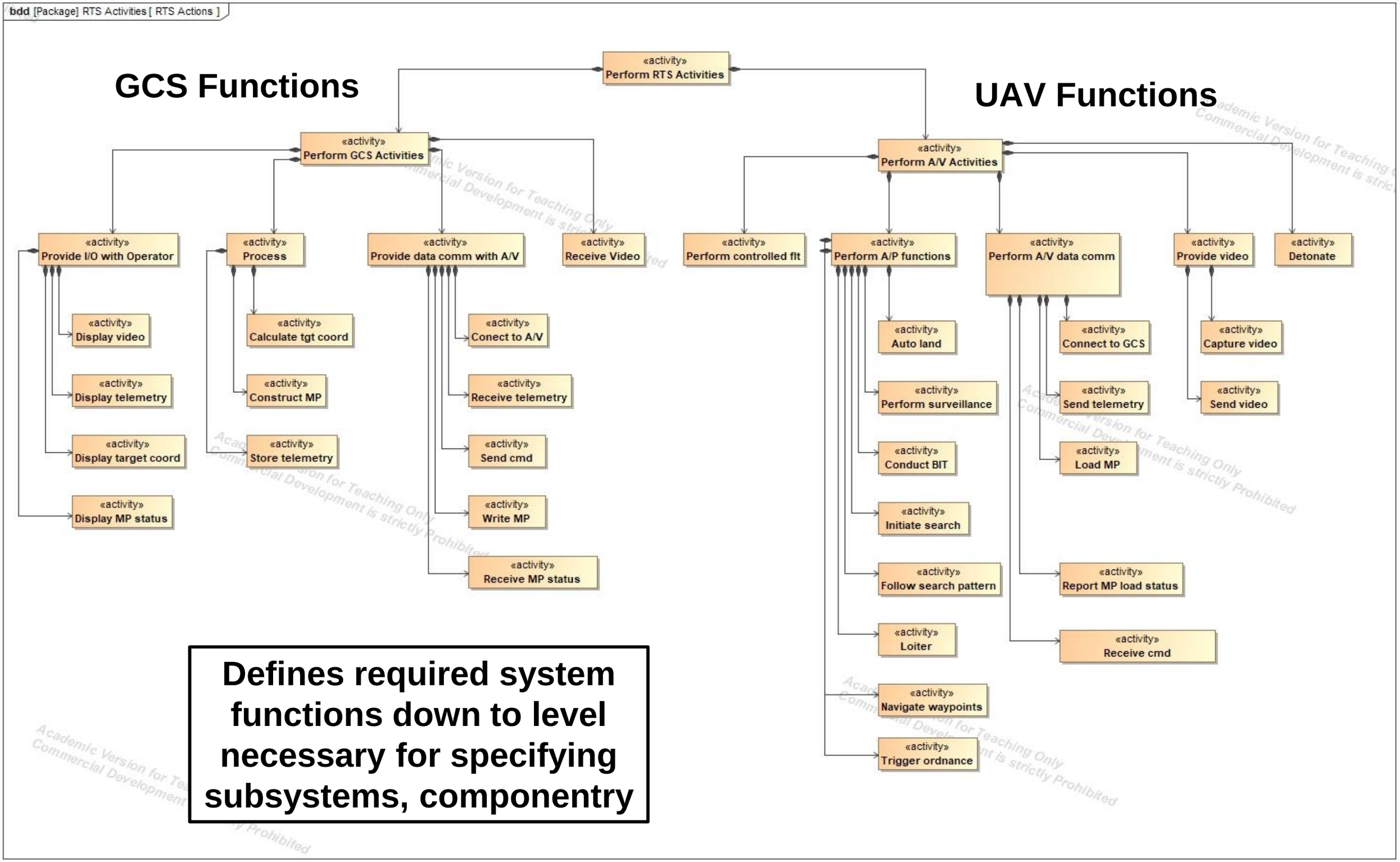
Use Cases can be directly related to Objectives or Composite Behaviors for Autonomous Systems Software Agents

RTS: Surveil Activity Diagram



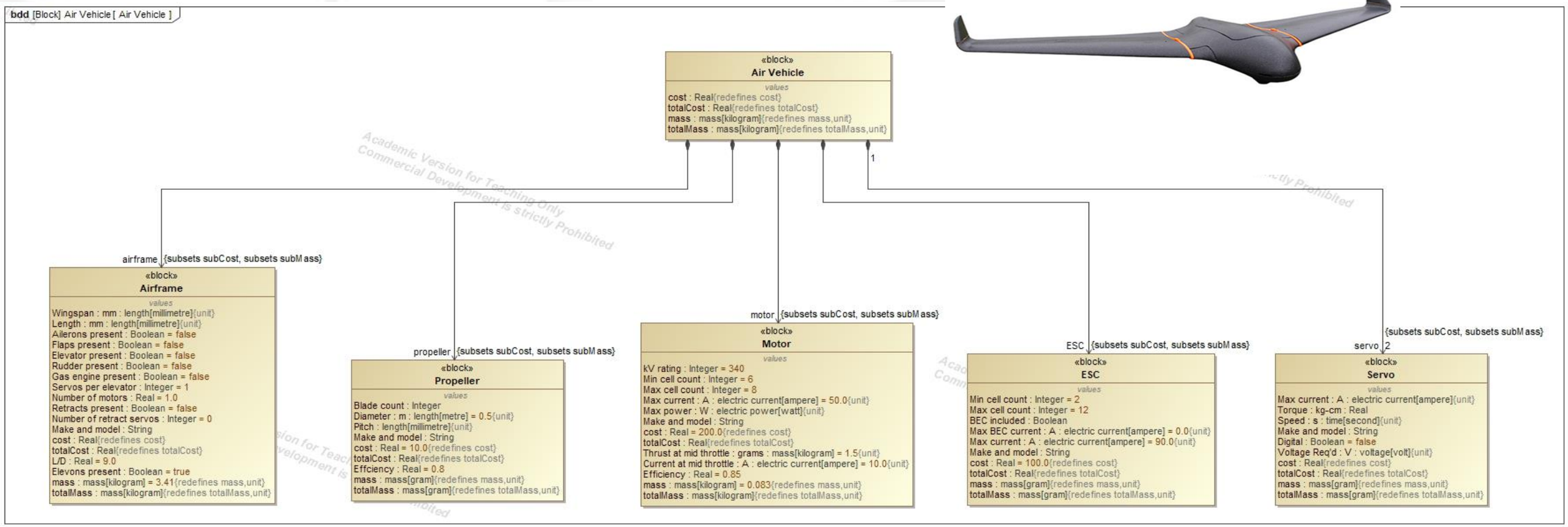
Lower level actions define required system functions; traceable to composite behaviors

RTS: Activities



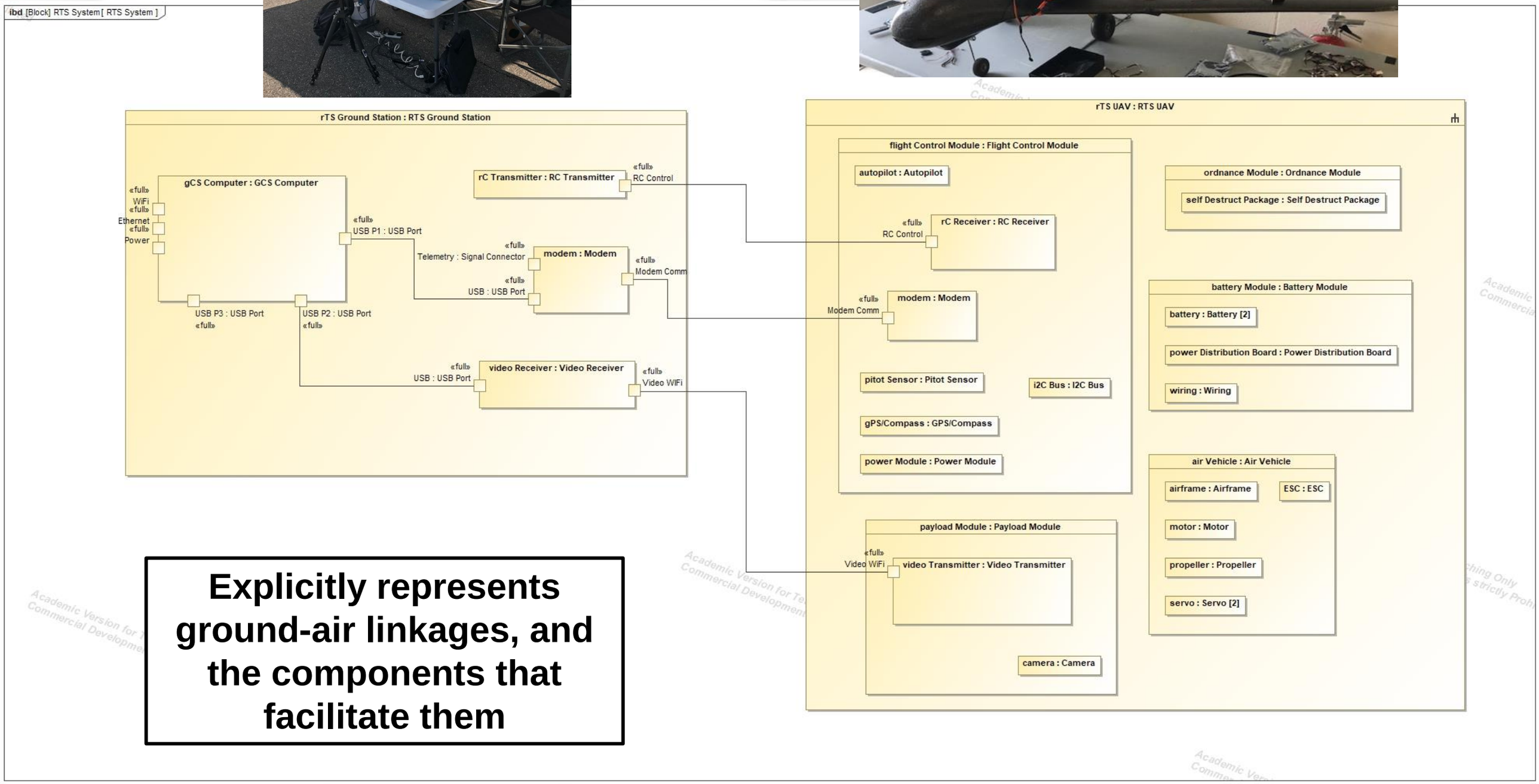
Defines required system functions down to level necessary for specifying subsystems, componentry

RTS: Vehicle bdd



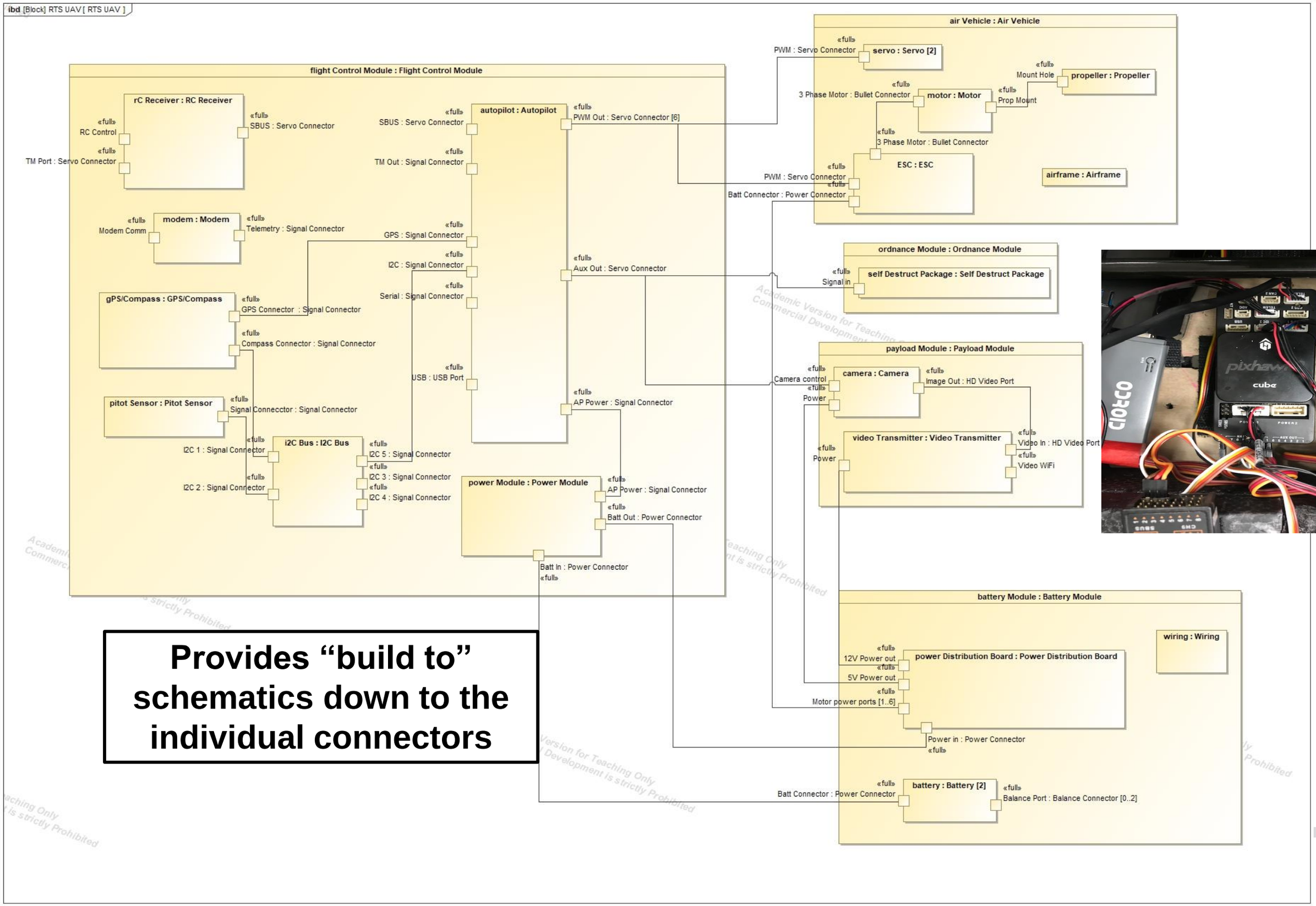
- **This bdd reflects design choices to combine with stock airframe choice**
- **Does not include autopilot, communication, and payload equipment, which combine with vehicle to form UAV**

RTS: System Level ibd



Explicitly represents ground-air linkages, and the components that facilitate them

RTS: Vehicle ibd

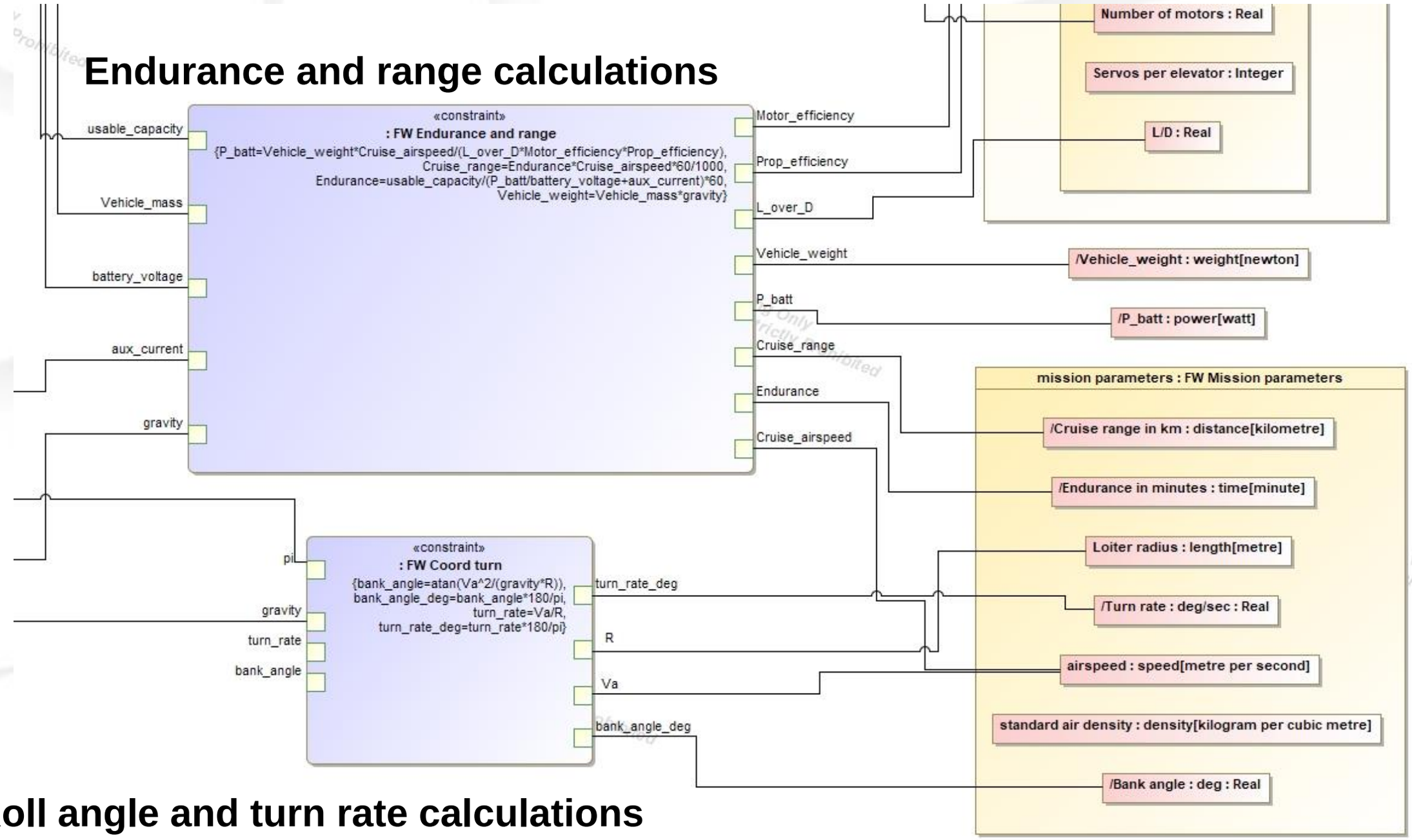


Provides "build to" schematics down to the individual connectors

RTS: Example constraints

Component block property values

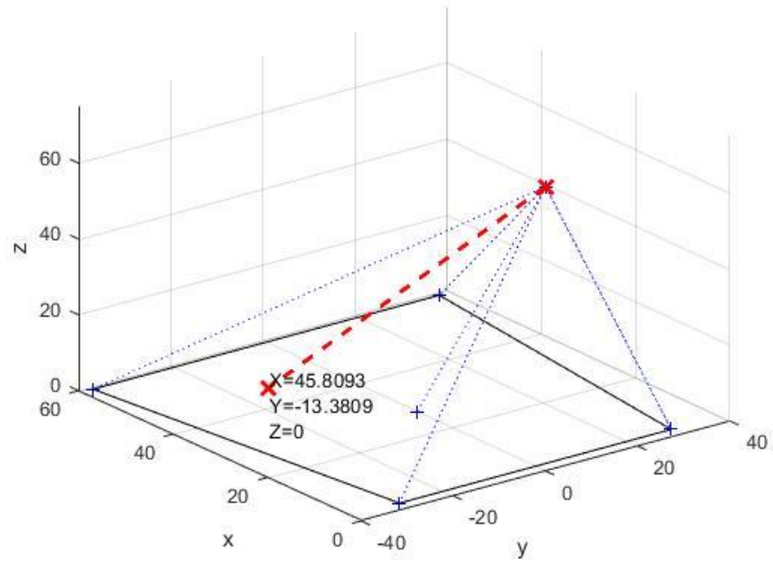
Endurance and range calculations



Roll angle and turn rate calculations

Impact – Improved Analysis and Requirements Traceability

- Example trade studies performed
  - Cost vs. performance
  - Weight vs. endurance
  - Battery capacity vs. endurance
  - Hover vs. endurance for VTOL
  - Ground resolution vs. camera and operating parameters
- All trade studies directly tied to architectural variations
  - Good traceability of design decisions
- Requirements directly traced to component selection and performance parameters
- Test cases directly traceable to requirements and architectural assemblies



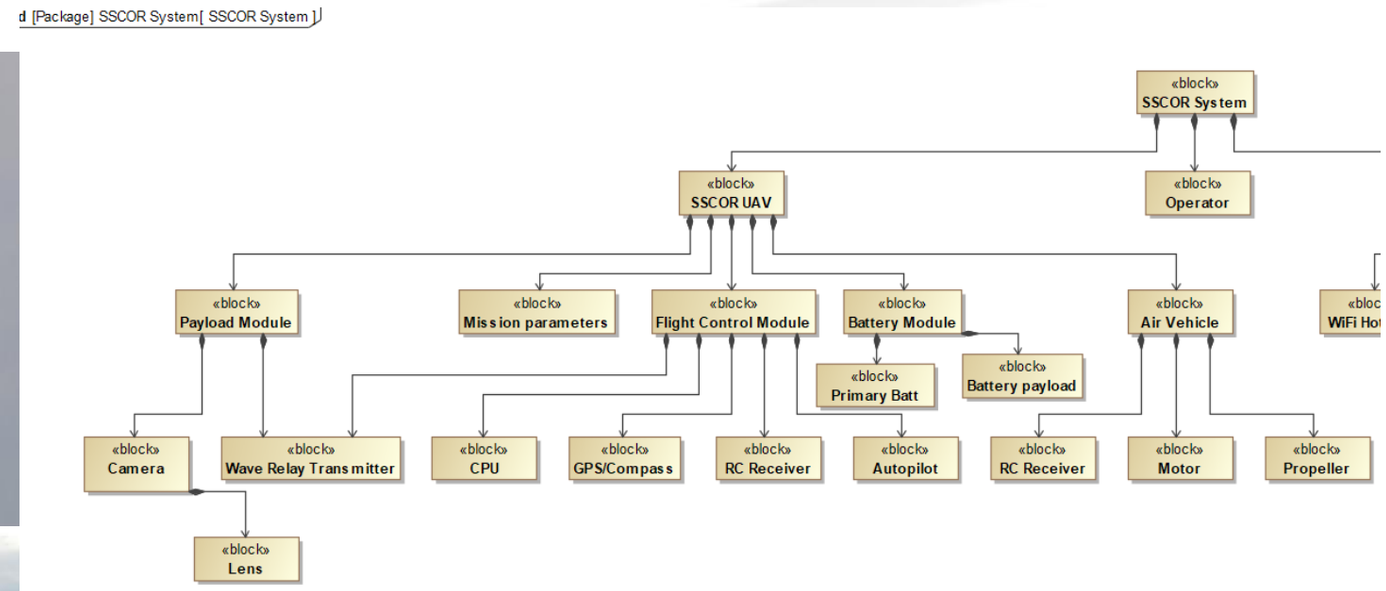
Research Employing the SUAS Reference Architecture

- Determining detectable and exploitable aspects of rogue SUAS
  - Architectural definition of common design approaches
  - Used to assess vulnerabilities of potential threat systems
- Improving the reliability of SUAS
  - Reliability analysis traceable to architectural designs and component choices
- Developing early, relative cost estimates from conceptual architecture definition
  - Parse SysML models to provide inputs for COSYSMO cost estimating tool
  - Parsed features include scenarios, requirements, functions, blocks and interfaces

Future Directions

- Demonstrate multi-domain application for ASRA
- Additional parametric analysis for SUAS RA
  - Performance calculations
  - Greater use of requirements verification where possible
  - Expand integration of MATLAB and other analysis tools
- Include design drivers stemming from DoD policy
  - Airworthiness
  - Flight test and system safety
  - Cyber security
- Developing Cubesat Reference Architecture
  - To serve similar need within AFIT's cubesat prototyping environment
  - Investigate application of ASRA to cubesat autonomy

Questions?


d [Package] SSCOR System[SSCOR System]
«block» SSCOR System
«block» SSCOR UAV
«block» Operator
«block» Payload Module
«block» Mission parameters
«block» Flight Control Module
«block» Battery Module
«block» Air Vehicle
«block» Primary Batt
«block» Battery payload
«block» Camera
«block» Wave Relay Transmitter
«block» CPU
«block» GPS/Compass
«block» RC Receiver
«block» Autopilot
«block» RC Receiver
«block» Motor
«block» Propeller
«block» Lens

SERC 2019 RESEARCH REVIEW

NOVEMBER 18-19, 2019
FHI 360 | WASHINGTON DC

MONDAY - NOVEMBER 18
SYSTEMS THINKING WORKSHOP
SERC DOCTORAL STUDENTS FORUM

TUESDAY - NOVEMBER 19
SERC SPONSOR RESEARCH REVIEW

REGISTER TODAY
▸ https://sercuarc.org/research-reviews/register2019/

NOV 18 | 9:30 AM – 6:00 PM

SYSTEMS THINKING WORKSHOP

SystemiTool is a systems thinking method and tool for mapping complex systems using "Systemigrams." This workshop will introduce the newly updated SystemiTool, as well as a look back over the past 30 years at systems thinking and Systemigrams. *NOTE: Registration for this workshop will be limited to the first 40 seats. Workshop attendees should bring their personal laptop in order to explore the full aspect of this tool.*

SERC DOCTORAL STUDENTS FORM

The SERC Doctoral Students Forum (SDSF) provides an opportunity for SERC Doctoral Fellows and other doctoral students conducting highly relevant, systems engineering-related research at any of the SERC collaborating universities to present their research in an open forum.

Day 1 will come to a close with an Evening Reception beginning at 6:00 pm

NEW
PI 101
Principal Investigator 101

A chance to review Pre-Award and Post-Award functions for all Principal Investigators – New and Seasoned as well as impart tips and answer FAQs to hopefully make the process smoother and more painless.

SERC 2019 RESEARCH REVIEW

NOV 19 | 8:00 AM – 5:00 PM | SERC SPONSOR RESEARCH REVIEW

The SERC SPONSOR RESEARCH REVIEW program and sessions focus on the latest research results from SERC researchers aligned with emerging and critical sponsor research needs.

PROGRAM TRACKS INCLUDE:

- MISSION ENGINEERING
- DIGITAL ENGINEERING TRANSFORMATION
- AI / AUTONOMY & SYSTEMS ENGINEERING
- SYSTEM SECURITY ENGINEERING
- SYSTEM ENGINEERING FOR VELOCITY
- EDUCATION & HUMAN CAPITAL DEVELOPMENT

This two-day event unites the government, industry, and university systems engineering research community in order to share research progress and discuss the most challenging systems engineering issues facing the Department of Defense (DOD) as well as other federal departments and agencies.

SERC TALKS

Thank you for joining us!

Please check back on the SERC website for today's recording and future SERC Talks information!

Subscribe and follow SERC on our social channels: