



# **An Exposé of Autonomous Agents in Command and Control Planning**

**Presented by:**

**Christopher Matthews**

**Claw.Matthews@us.army.mil**

**U.S. Army RDECOM CERDEC**

**Fort Monmouth, NJ**



**RDECOM**



# Talking Points

- **Army Program**
- **Mission need**
- **Autonomous agents**
  - **Motivation**
  - **Employment**
- **Agent development framework**
- **Prototype – Maneuver Sustainment Planner**
- **Design & implementation of agents**
- **Summary**





# Motivation

- **Logistics Command & Control (LogC2) Advanced Technology Demonstration (ATD)**
  - Integrate logistics and maneuver planning
  - Faster OPTEMPO & reduced logistics footprint
  - Shorten Combat Service Support (CSS) planning times
  
- **Achieved through the research, development and transition of:**
  - Collaborative, cross-functional planning tools
  - Optimization tools for increased maneuver sustainment efficiency
  - Adaptive, predictive consumption models and demand generation functionality
  - Near real-time running-estimate decision aid software
  - Dynamic re-planning and execution-monitoring software capability





# Agent Development Framework

## Cognitive Agent Architecture (Cougaar)

- Darpa initiative
  - Advanced Logistics Program (ALP) (FY96 – FY01)
  - UltraLog Program (FY01 – FY04)
- Features
  - Distributed, large-scale workflow engine
  - Open source Java software
  - Multi-Resolutional Logical Data Model (LDM)
  - Built-in dynamic re-planning & execution-monitoring capability
  - Asynchronous communication protocol
  - Classic publish/subscribe blackboard
  - Domain independent



**RDECOM**



# ALP Goals

## ■ Technical Goals:

- Distributed agent architecture research
- Distributed information management research
- Real-time information fusion research

## ■ Functional Goals:

- Automated logistics plan generation
- Real-time logistics situation assessment
- End-to-end movement control
- End-to-end rapid supply



**RDECOM**



# UltraLog Goals

## Expanding the ALP vision:

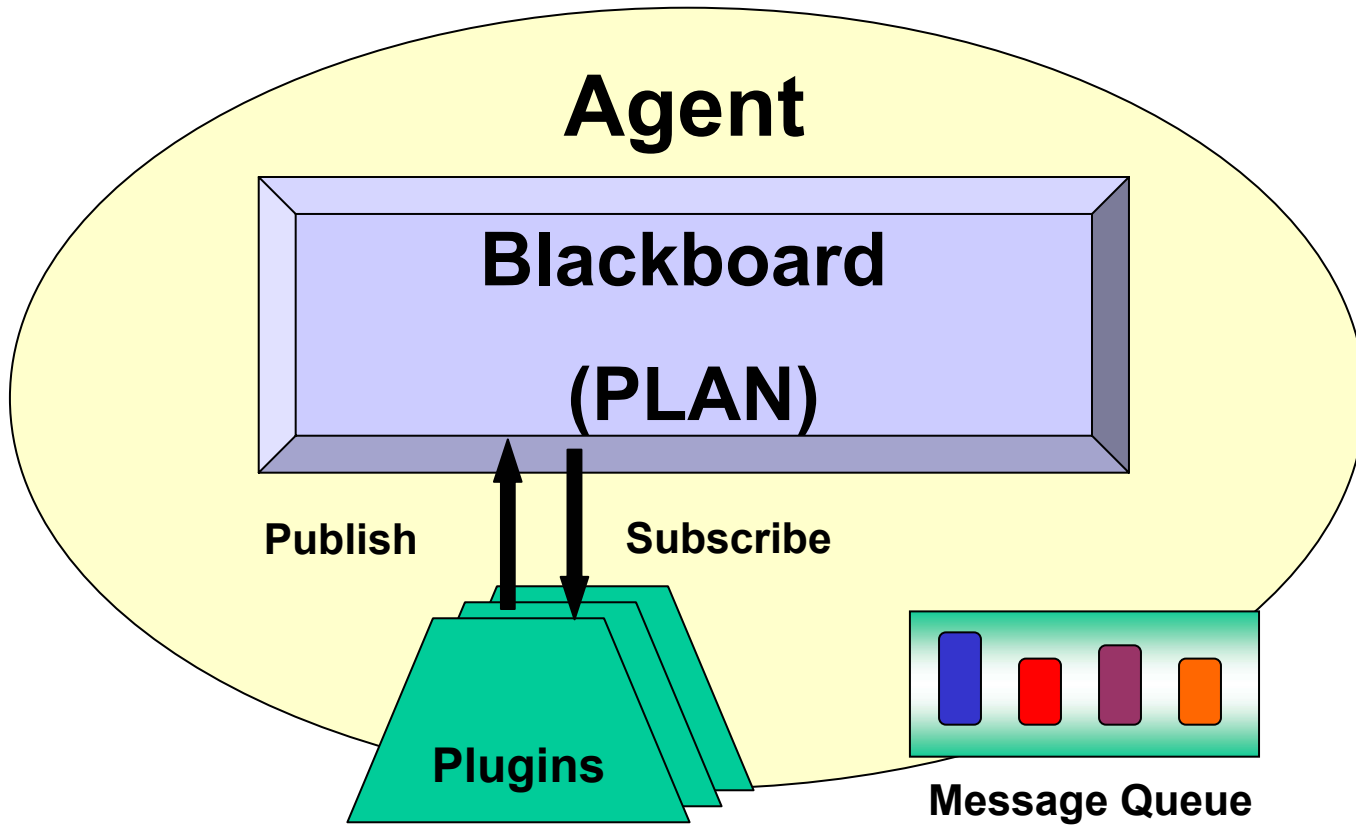
- **Military logistics domain**
- **Enhance the Cougar framework**
  - **Security – trusted systems under information warfare attacks**
  - **Scalability – stability for large, distributed network of agents**
  - **Robustness – high state of survivability in chaotic environments**
  - **System integration – combining all of above to achieve desired systemic effects**



**RDECOM**

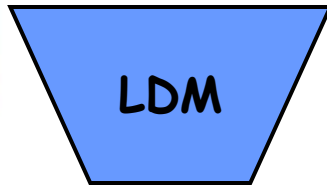


# Cougaar – Agent basics

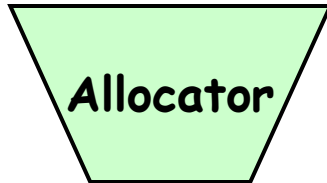




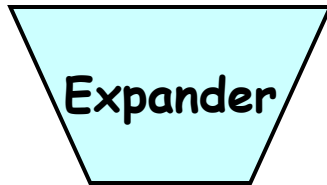
# Cougaar - Plugins



Populating society with data from external systems.



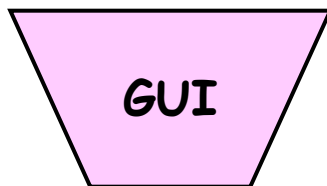
Allocates tasks to other agents/assets.



Decomposes tasks into more manageable Subtasks.



Evaluates the projected and actual results of allocated tasks



Provides the web-based front end for viewing activity within society.

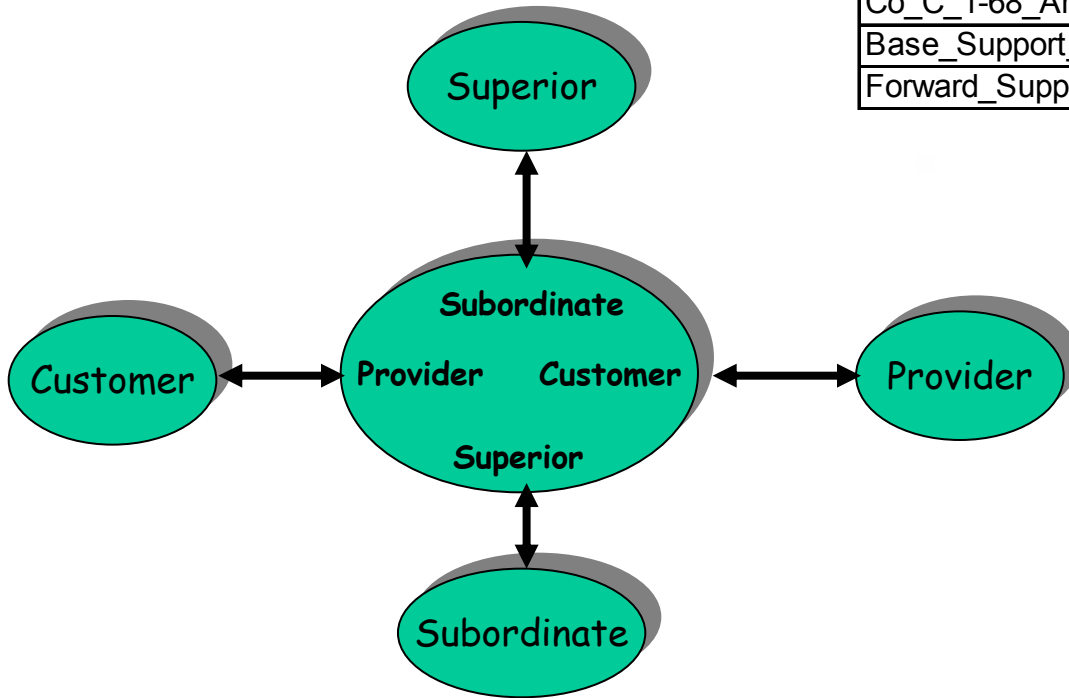






# Cougaar – Agent behavior

## Agent Relationships



Agent	Role	Role
Bn_1-8_Infantry	MobileRefuelCustomer	CombatUnit
Co_A_1-8_Infantry	MobileRefuelCustomer	CombatUnit
Co_B_1-8_Infantry	MobileRefuelCustomer	CombatUnit
Co_C_1-8_Infantry	MobileRefuelCustomer	CombatUnit
Bn_1-68_Armor	MobileRefuelCustomer	CombatUnit
Co_A_1-68_Armor	MobileRefuelCustomer	CombatUnit
Co_B_1-68_Armor	MobileRefuelCustomer	CombatUnit
Co_C_1-68_Armor	MobileRefuelCustomer	CombatUnit
Base_Support_Company	MobileRefuelProvider	LogisticsUnit
Forward_Support_Company	MobileRefuelProvider	LogisticsUnit

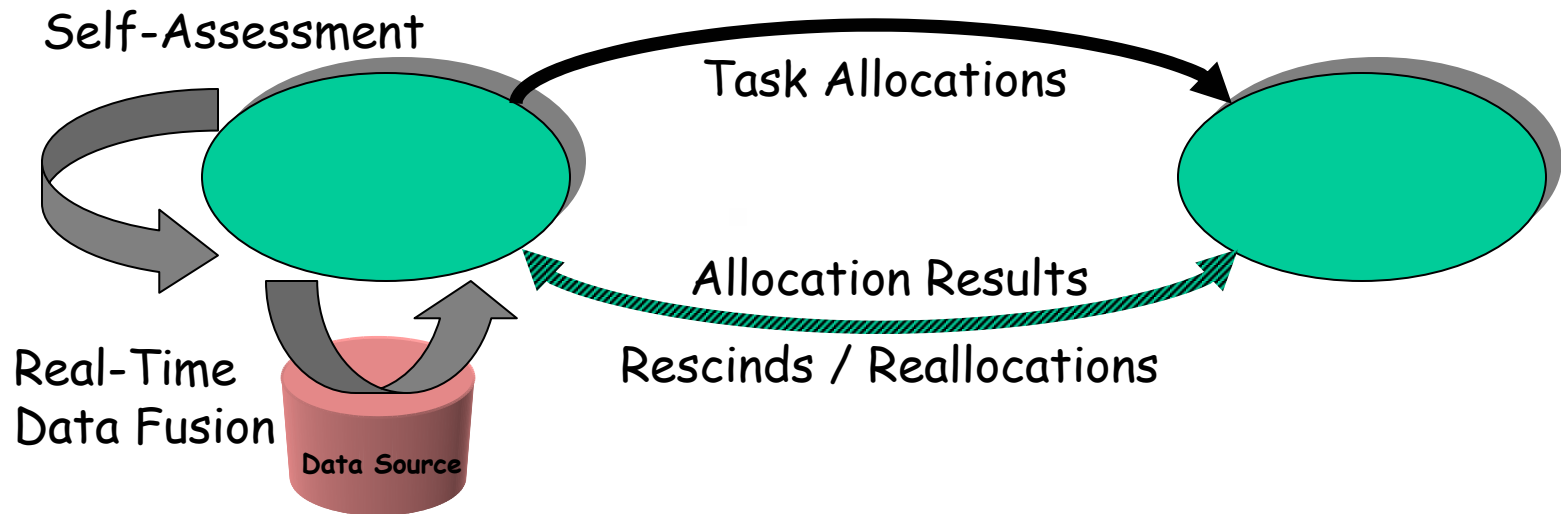
## Agent Roles





# Cougaar – Re-planning Concept

## Dynamic Re-planning & Execution Monitoring





# Cougar – Logical Data Model (LDM)

## ■ Requirements

- Support over 6 million items
- Support efficient transport and distribution of objects
- Support modification and extension during execution

## ■ Principles

- Based on the properties of objects and not what they are
- Represent all the properties of assets needed to reason about them
  - Over a range of granularities
  - Supporting their time-varying nature
  - Referring to specialized properties of assets

## ■ Implementation

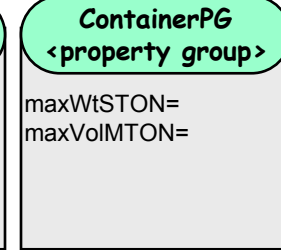
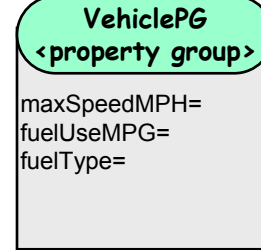
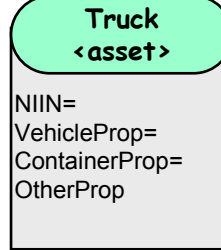
- Use prototypes and delegation to reduce classes needed
- Prototype classes determine the required properties of all instances
- Related properties are collected in Property Groups
- Asset instances delegate properties to their Prototype instances



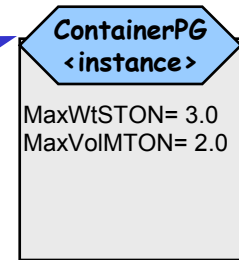
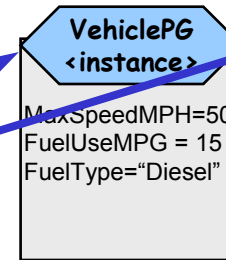
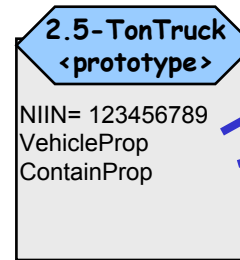
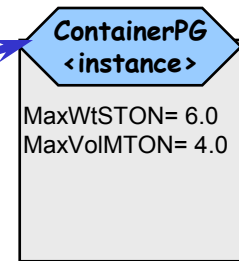
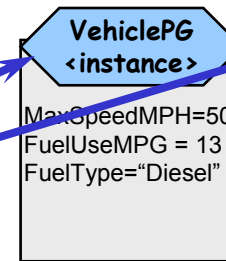
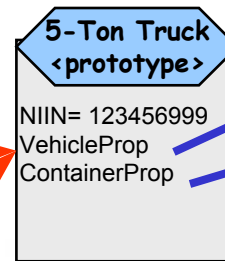
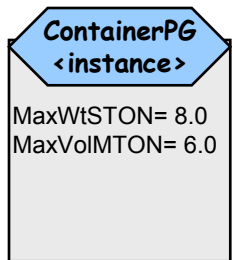


# Cougar – LDM examples

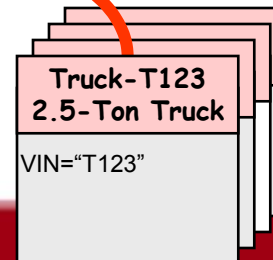
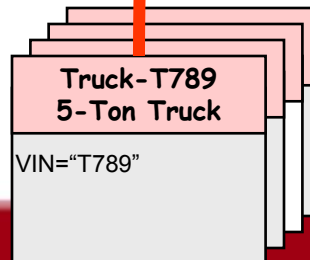
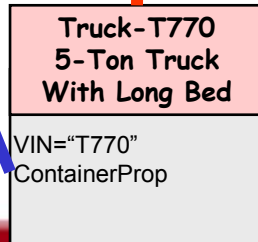
## LDM Classes



## Asset Prototypes and Property Groups



## Asset Instances





# Maneuver Sustainment Planner (MSP)

## Goals

- **Develop proof-of-concept prototype**
  - Integrate logistic-planning impacts into the maneuver planning process
  - Develop a *detailed* logistics plan to support maneuver operations
  - Model maneuver activities and generates expected logistics demand as a function of platform, posture, and optempo
  - Provides dynamic re-planning & execution-monitoring capability
- **Evaluate Cougar and agent benefits for C2 planning**



**RDECOM**



# MSP – Why Cougar?

- **Core planning capabilities included**
  - **Dynamic replanning/execution monitoring**
  - **Resource management (asset scheduling)**
- **Rapid software development**
- **Leverage existing logistics software components**
- **Agents map elegantly to military force structures**
- **Digitization of reusable, intricate, and highly complex business models**
- **Easy to introduce external data into agent society**



**RDECOM**



# MSP - Approach

- **Functional analysis & design**
  - Agent Enumeration
  - Role/Relationship Analysis
  - Plugin Enumeration
  - Task Grammar
  - Asset/Property Requirements Analysis
  - Execution Monitoring and Dynamic Replanning Analysis
- **External system interfaces**
  - Databases
  - Maneuver Command & Control (MC2) application
  - MSP Plan Viewer (GUI)
- **Cougaar *Plugin* development**

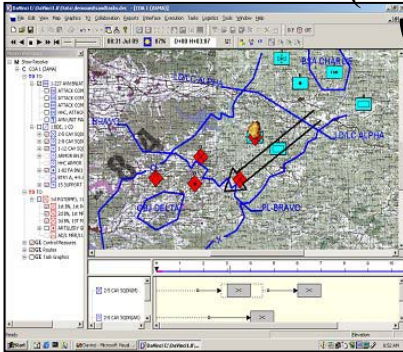




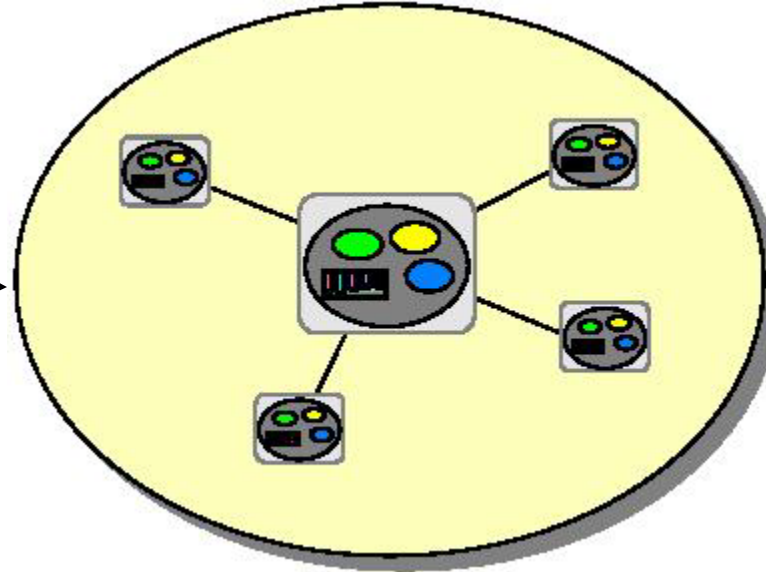
# MSP – Preliminary System Overview

## Maneuver Sustainment Planner (MSP)

Maneuver Command & Control  
Application  
(MC2)

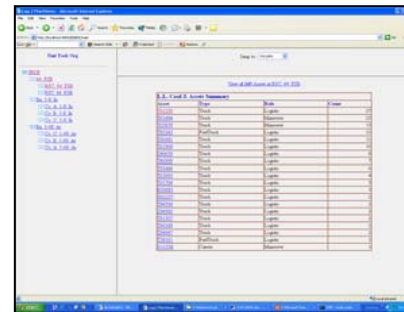


Digitized  
Maneuver  
Plan



Feedback Loop  
(Feasibility, Alerts,  
& Recommendations)

Detailed Logistics Plan



**RDECOM**





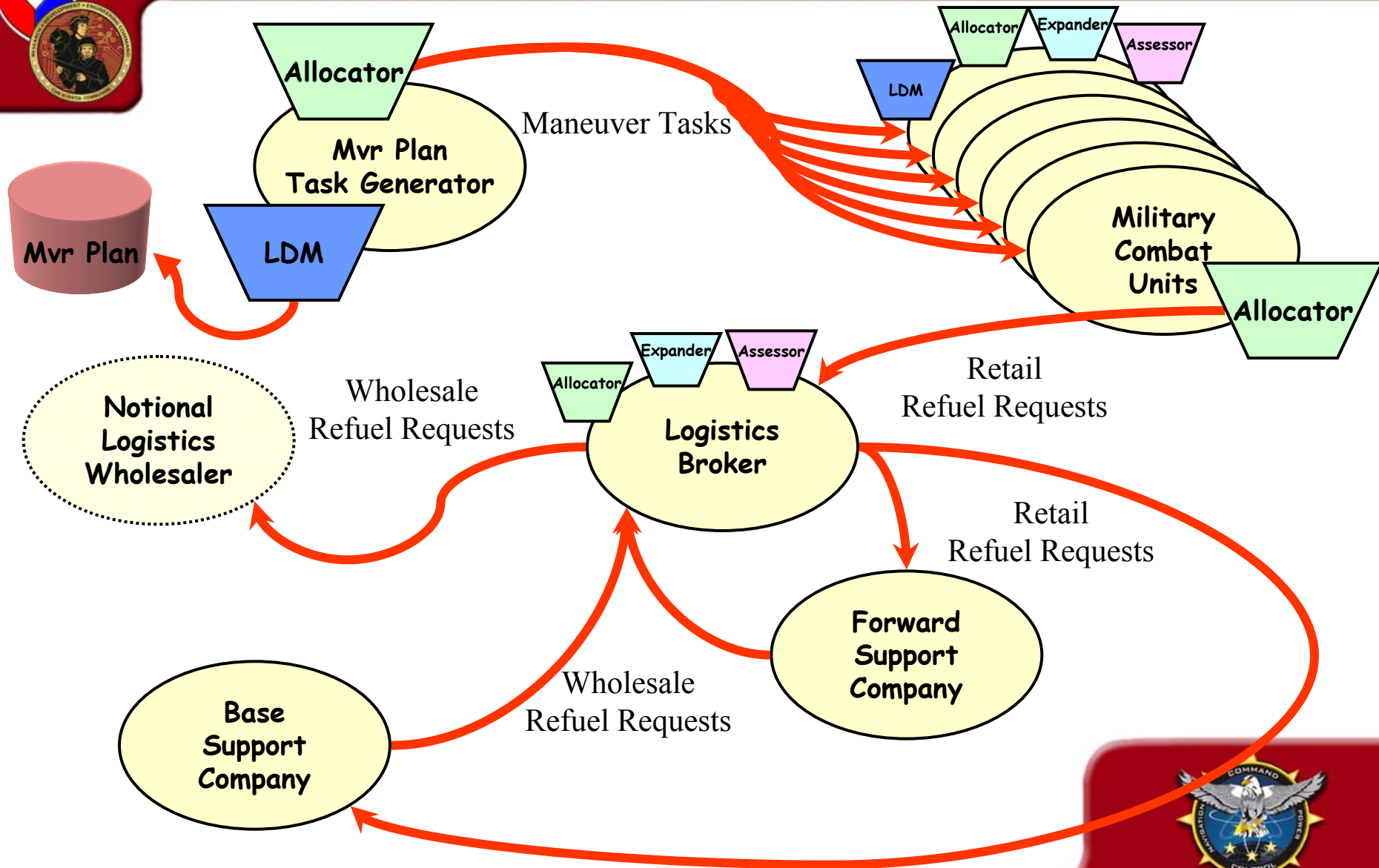
# MSP – Constraints

- **No control over the maneuver plan (read only)**
  - **Maneuver Command & Control (MC2) system**
    - **Stove-piped system**
    - **Rigid, closed plan representation**
    - **Large, unwieldy XML plan data**
- **Atypical Cougar use**
  - **Driven by MC2**
  - **Short-lived vs. 24x7**
- **Demand generation (simulate consumption)**
  - **CASCOM Equipment Usage Profiles (EUP)**
    - **Equipment type**
    - **Optempo**
  - **Same as MC2**





# MSP – Agent Community







# In Summary

## ■ Pros

- Rapid software development
  - Provides logical roadmap for application design & development
- Ideal for military planning systems
- Domain independent
- Open source software
- Core planning & information management functionality
- Well documented architecture & developer guides

## ■ Cons

- Large overhead
- Bandwidth intensive
- Steep learning curve
- Poorly documented software (sparse Javadocs)
- Frequent Architecture upgrades





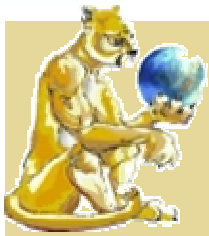
# Backup Slides



**RDECOM**

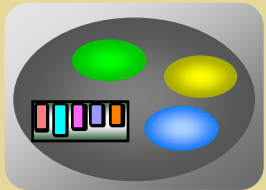


# Cougaar + UltraLog

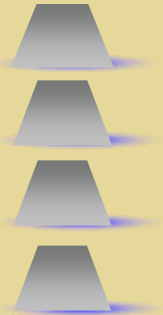


## Cougaar

Generic Agent



Generic Plugins



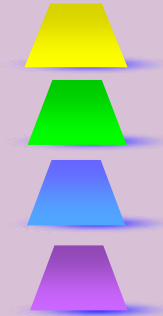
- Basic building blocks
- Easy to specialize
- Domain independent

+



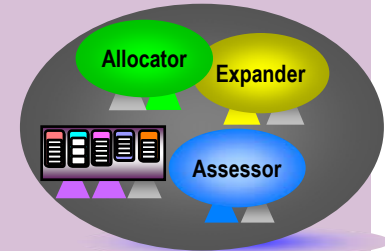
## UltraLog Society

Specific Plugins



=

Domain Agent



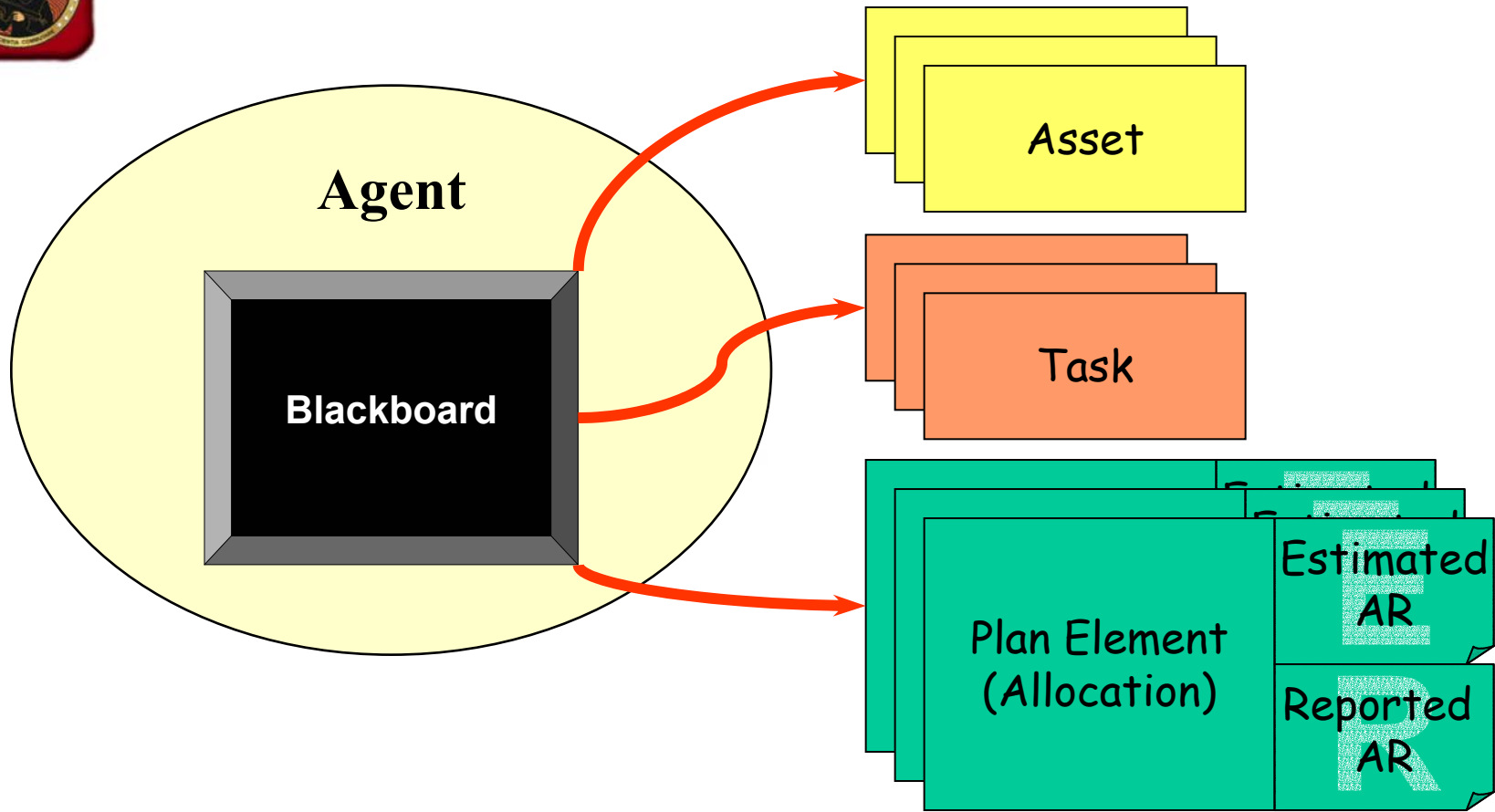
- Military specific processes
- Interfaces to military systems
- Specific to Logistics Domain



**RDECOM**

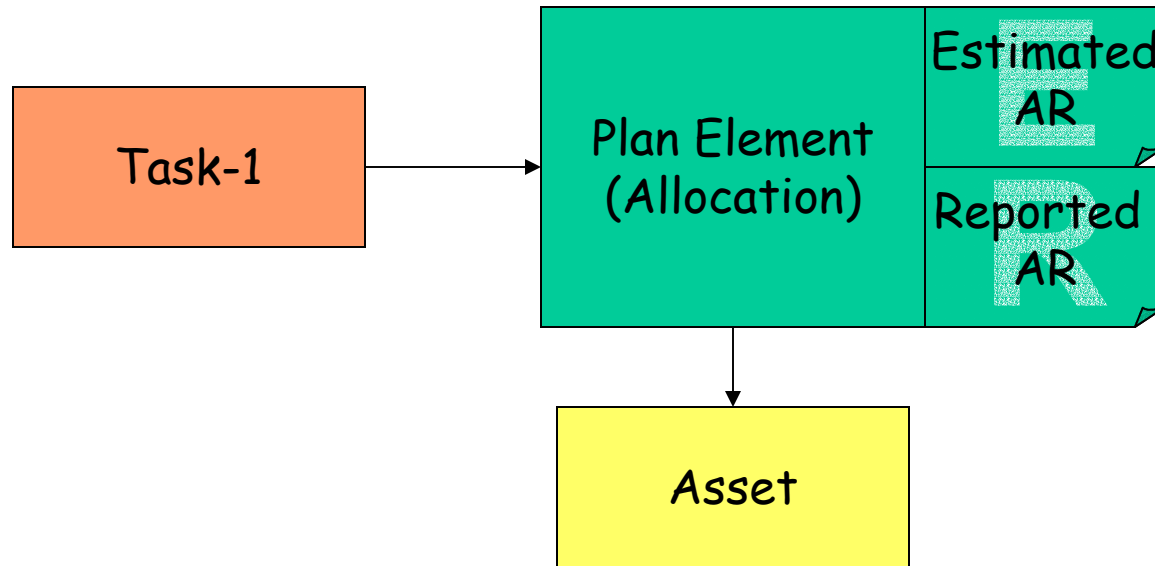


# Cougaar – The Distributed Plan





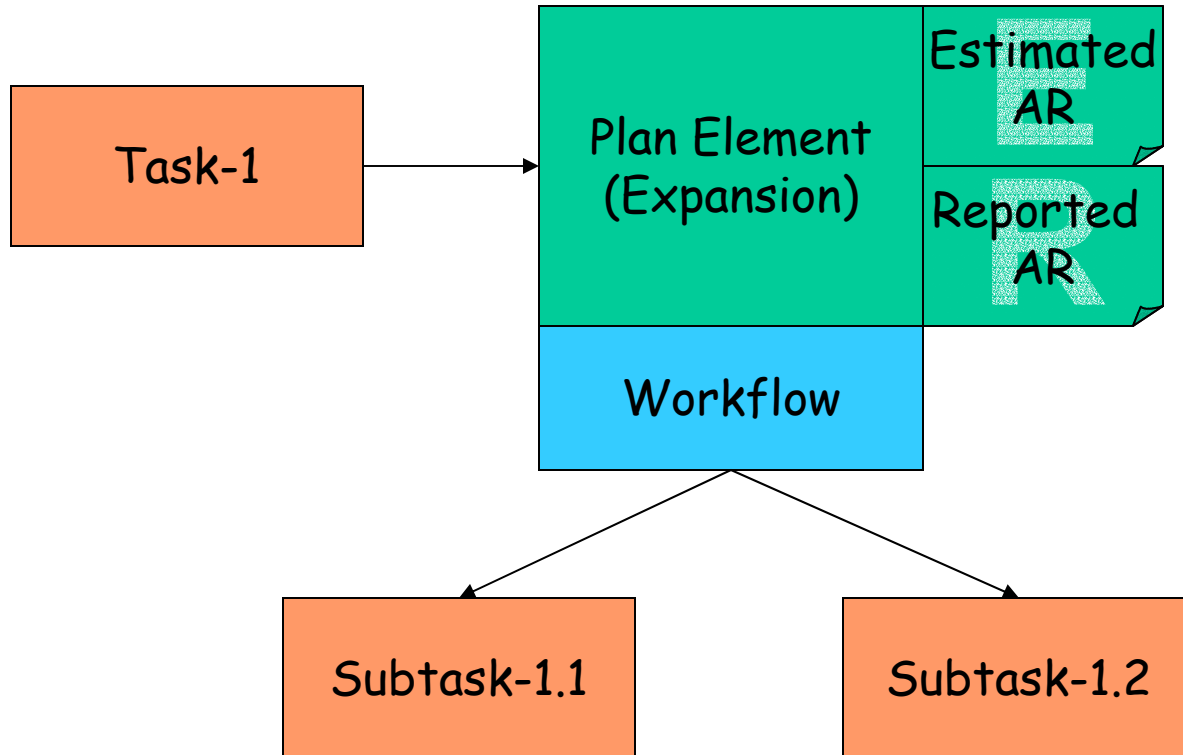
# Plan Element - Allocation





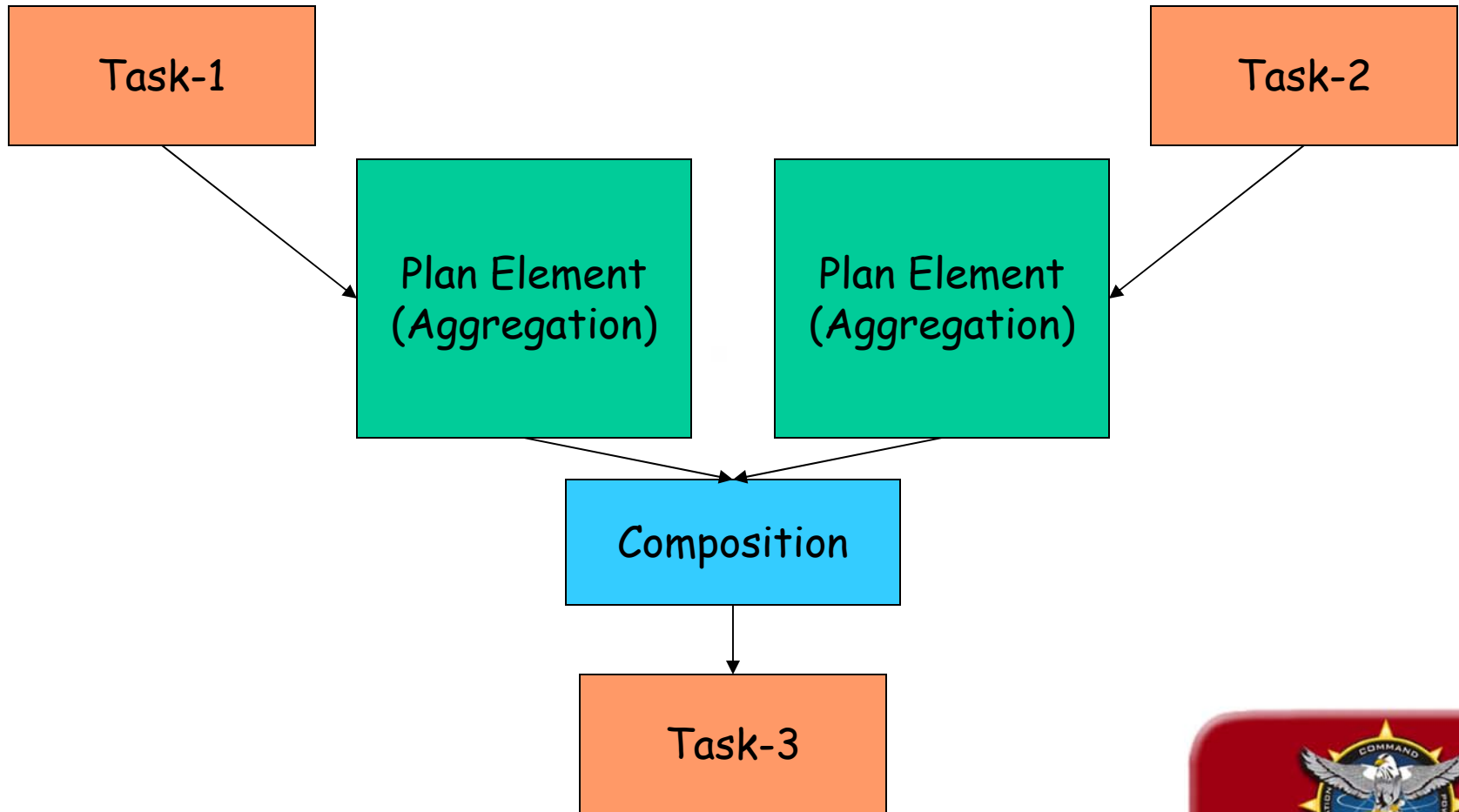


# Plan Element - Expansion



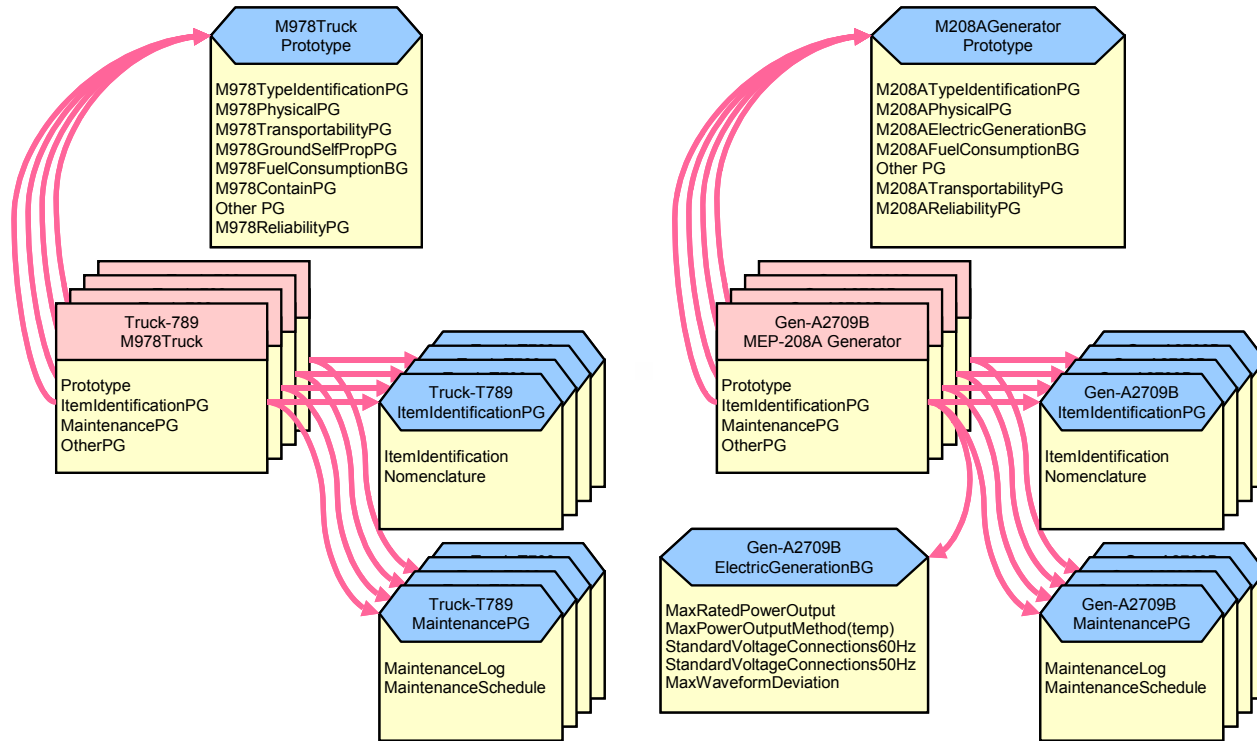


# Plan Element - Aggregation





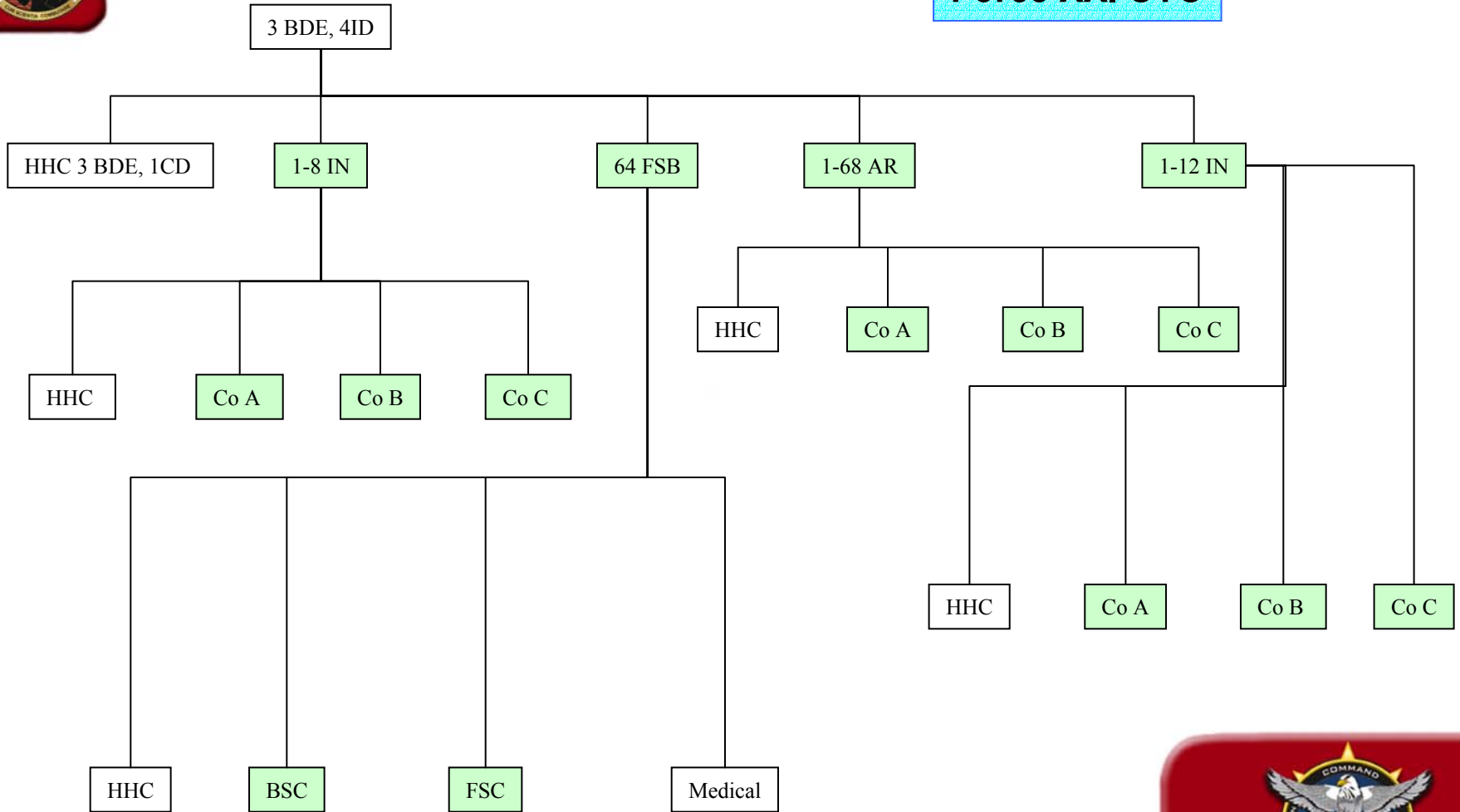
# Cougaar – LDM Asset





# MSP – Military Agents

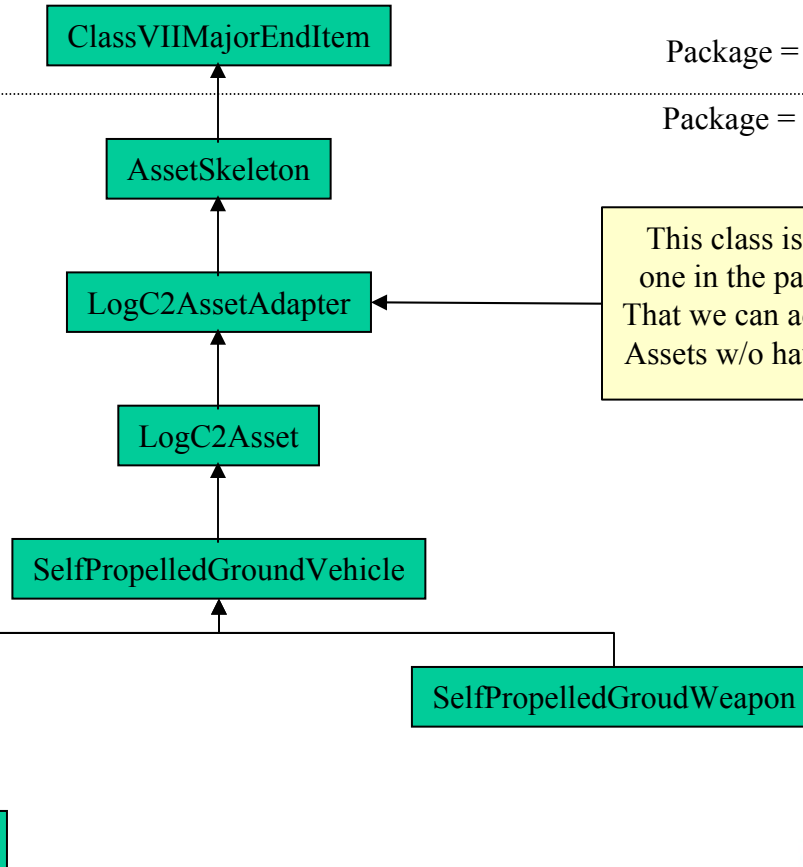
Force XXI UTO



RDECOM



# MSP – Asset Class Hierarchy



Package = org.cougar.glm.ldm.asset

Package = logc2.assets

All Assets in the logc2.assets package are machine generated from the logc2props.def and the logc2assets.def files. All ground vehicle Assets will be of one of the types depicted here.

This class is hand generated (the only one in the package.) This is needed so That we can add functionality later to all Assets w/o having to mod lots of classes.

Open to view LogC2 Vehicle List	
---------------------------------	---





# References

- Barger, Mark, & Wong, Jason. (2004). *Cougaar Training Slides*.
- BBN Technologies. (Version 10.0). (2003). *Cougaar Architecture Document*.
- BBN Technologies. (Version 10.0). (2003). *Cougaar Developer's Guide*.
- Berliner, Jeffrey, Thome, Michael, & Cerys, Daniel. (2003). *Multi-Resolutional Knowledge Representation Using Prototypes and Properties*.
- “Cougaar Open Source Web Site”, <http://www.cougaar.org>.
- “UltraLog Web Site”, DARPA, <http://www.ultralog.net/>.

