



A Reconfigurable Multi-agent Society for Transportation Scheduling and Dynamic Rescheduling

Gordon Vidaver
David Montana
Talib Hussain
Jeff Berliner

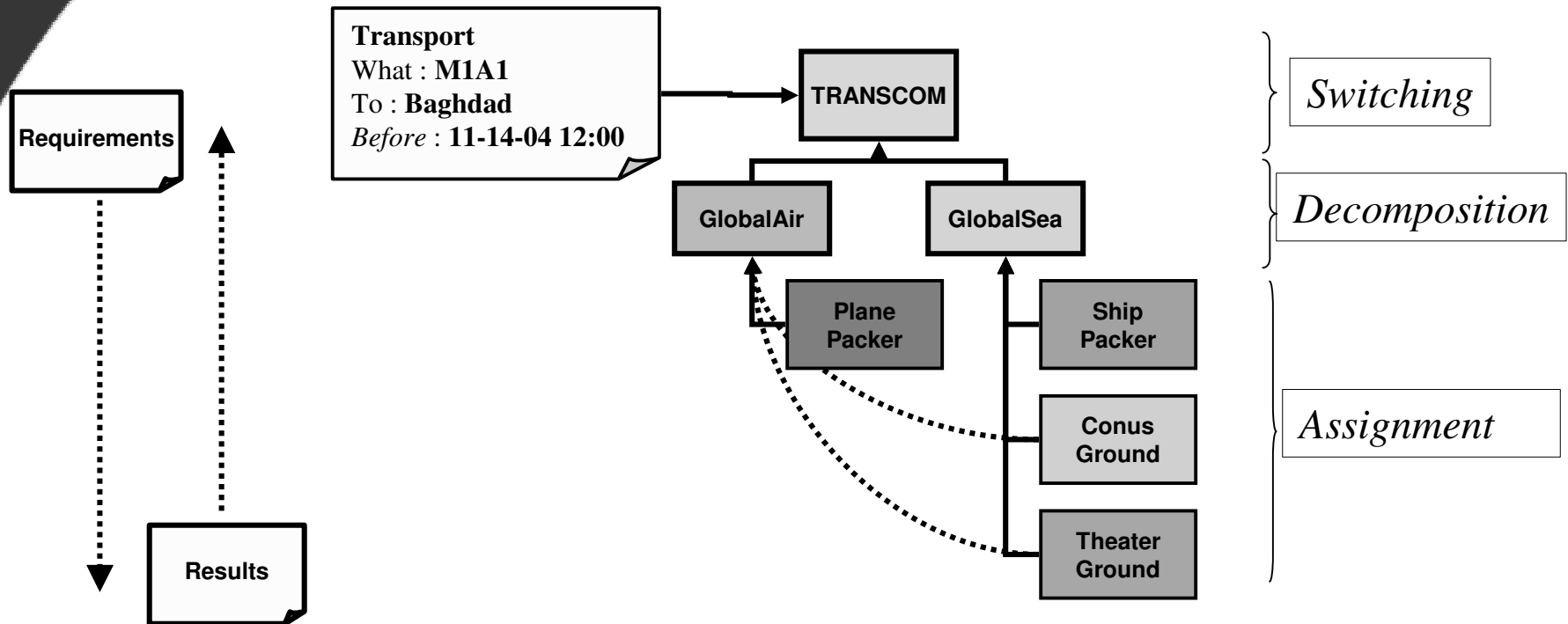
Tuesday, April 19th, 2005

BBN
TECHNOLOGIES

Introduction and Overview

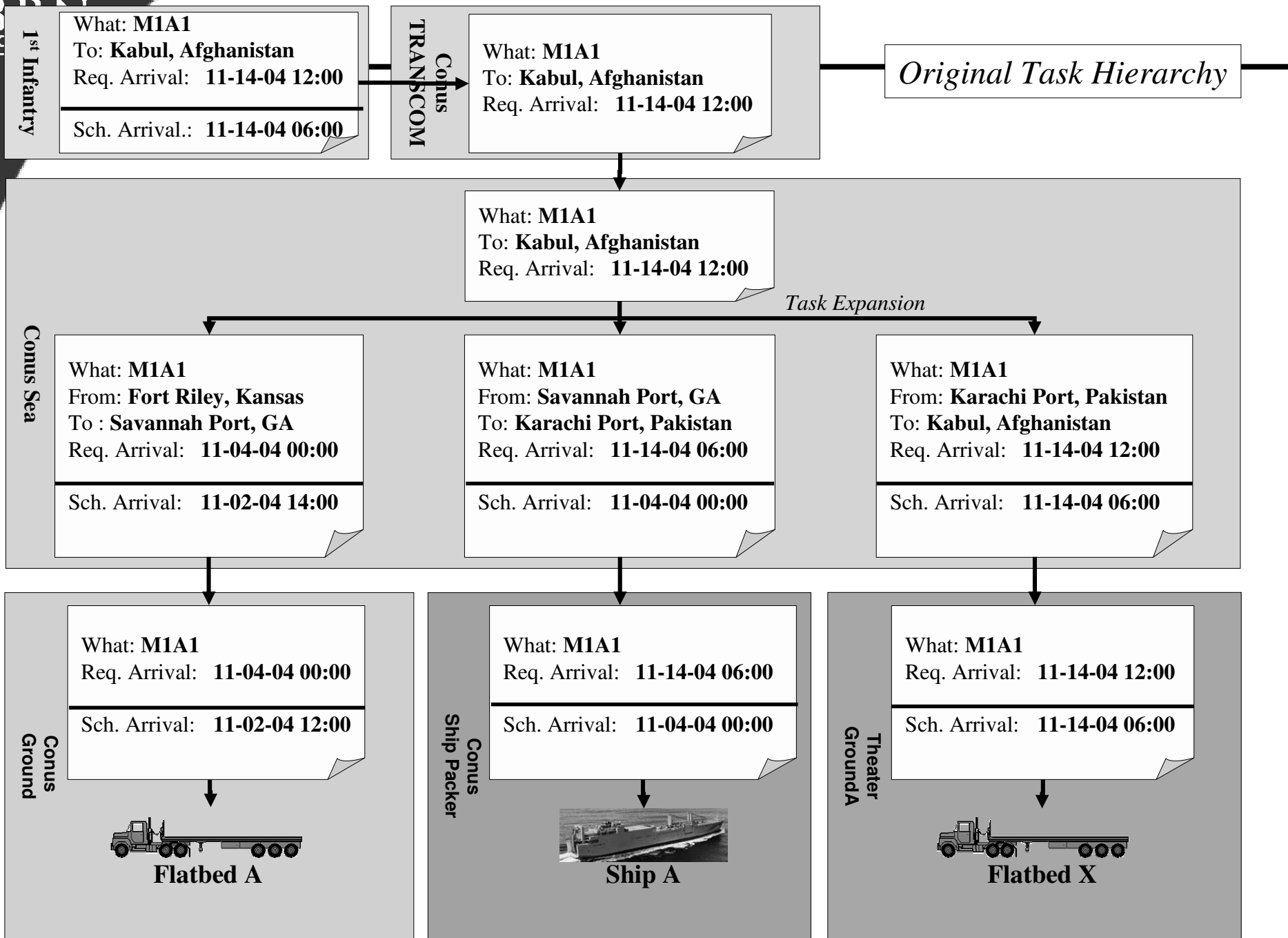
- Agent-based system to automate the process of scheduling most of the transportation assets for the U.S. military (Strategic Deployment)
 - Implemented with Cougaar Open Source agent framework
 - A fixed schedule based on static requirements is insufficient.
 - Instead, maintain the schedule as a dynamic entity in the face of
 - changing requirements
 - unreliable assets
 - unexpected events during execution.
- Dynamic rescheduling
 - Updating an existing schedule in response to changes in the scheduling resources and requirements
 - a special challenge in a multi-agent environment
 - propagate and coordinate changes between the different agents.
- Load management
 - The ability of the multi-agent society to change its structure in response to changing computational requirements.
 - Techniques to update the allocation of computational resources
 - Static approaches
 - Dynamic approaches, e.g. dynamic agent creation

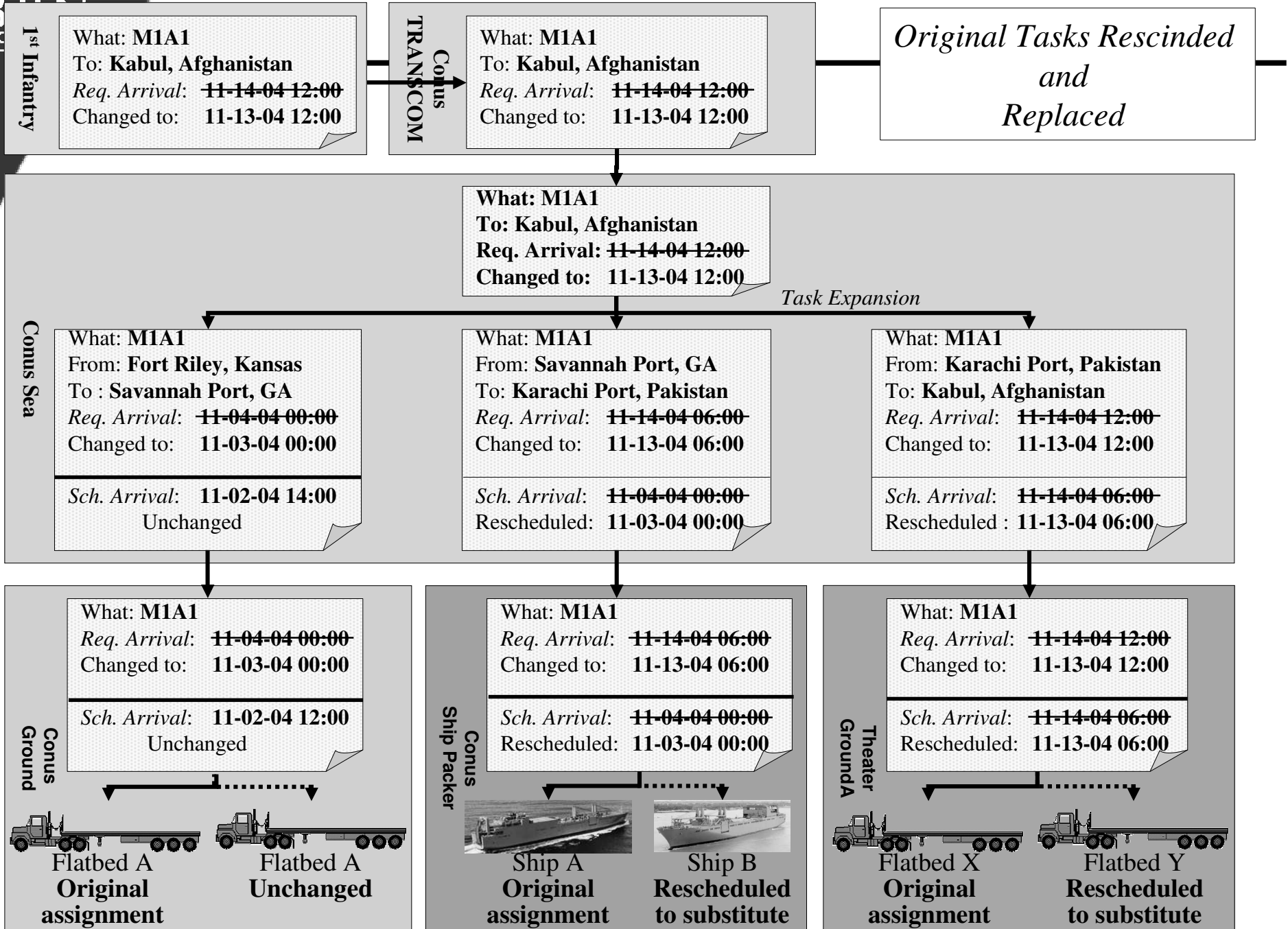
Seven agent Transportation Scheduling Community



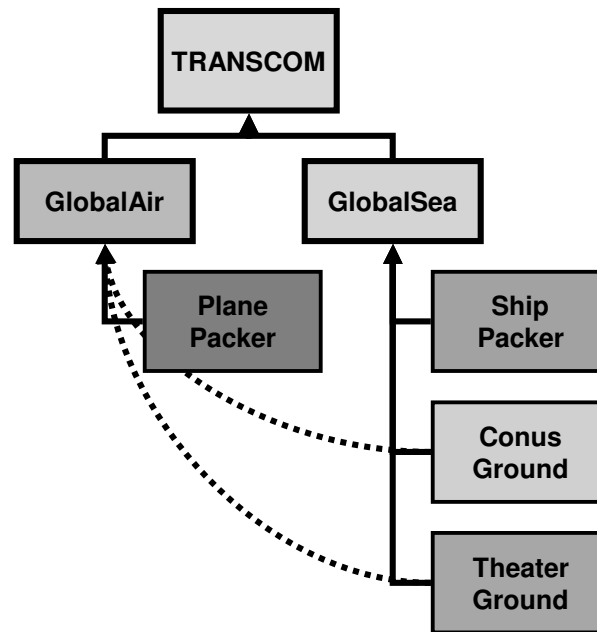
- Part of a larger society which generates transportation requests
- All supplies, personnel, and land vehicles in the plan must be transported from their original locations in the continental U.S. (CONUS), to air or sea ports, across the ocean, and then across land to their final destinations.

- 1) Choose best agent to handle high-level task
TRANSCOM acts as a “switch” for tasks – choose sea vs air
- 2) Break down high-level task into manageable sub-tasks
–Global Air/Global Sea orchestrate the planning of the legs of the journey
- 3) Solve encapsulated scheduling problem
–Leaf agents schedule these legs against physical assets (trucks, planes, ships).

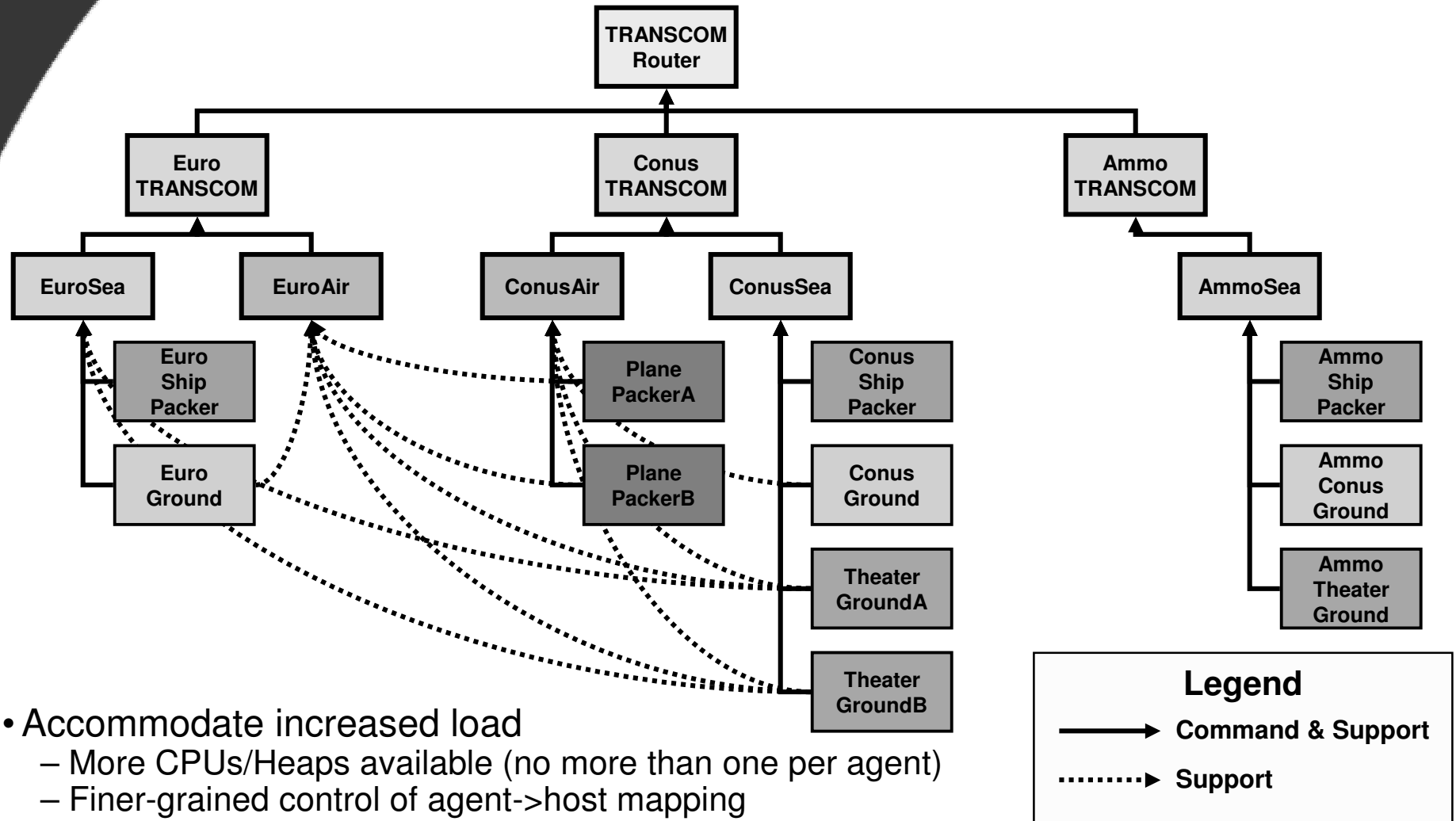




7 Agent Transportation Community



Expanded 20 Agent Transportation Community



- Accommodate increased load
 - More CPUs/Heaps available (no more than one per agent)
 - Finer-grained control of agent->host mapping
- Some problems fully separable
 - Ammunition travels on special ammunition ships
- Split an agent into two or more agents without a natural division
 - Theater Ground A and B each handle half of the tasks and resources for ground transportation to the destination.

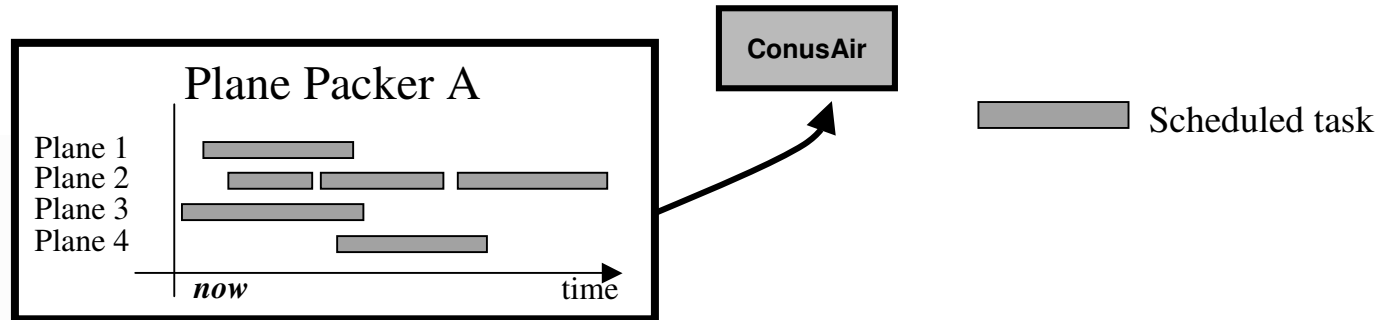
Legend

—> Command & Support

.....> Support

Dynamic Agent Creation and Resource Transfer: Plane Packer

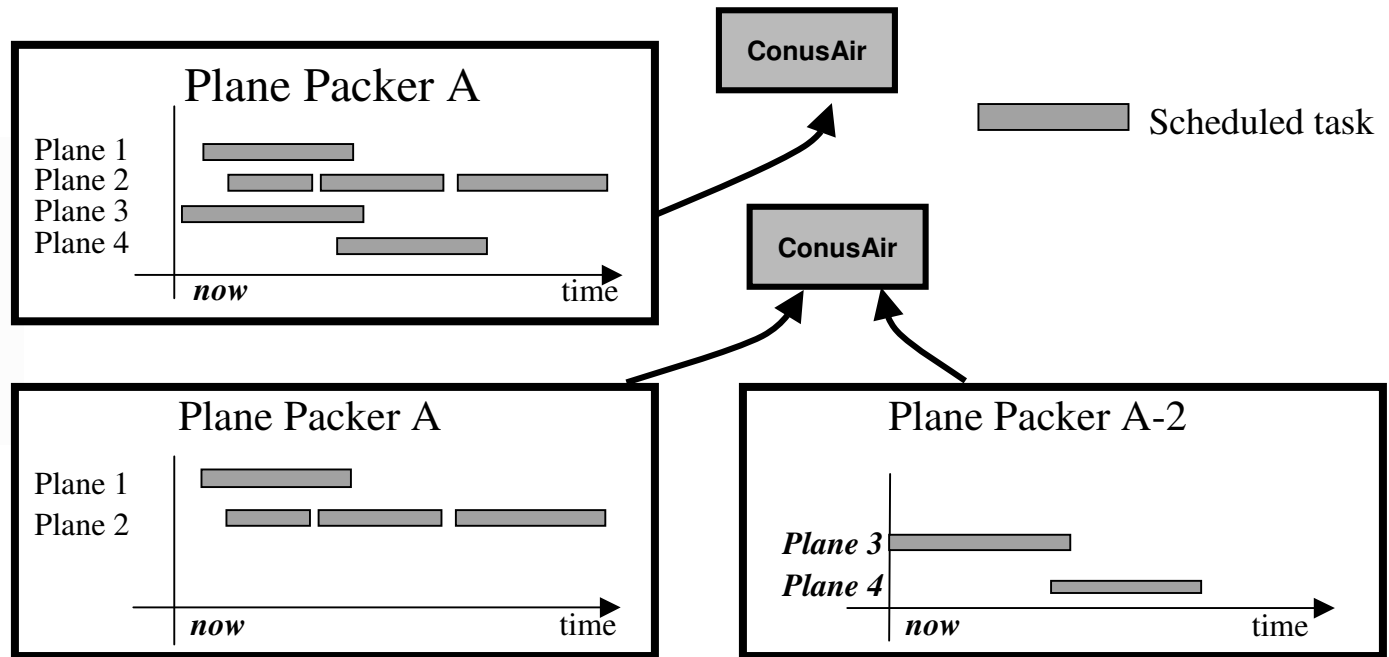
Adaptive load balancing automatically adjusts mapping of computational resources to agents as requirements change.



- Dynamic Agent Creation Steps
 - **Falling-behind sensor:** Detect agent not processing requests in a timely manner.

Dynamic Agent Creation and Resource Transfer: Plane Packer

Adaptive load balancing automatically adjusts mapping of computational resources to agents as requirements change.



- **Dynamic Agent Creation Steps**

- **Falling-behind sensor:** Detect agent not processing requests in a timely manner.
- **Spawn a new agent:** On a new (or underutilized) CPU.
- **Transfer half the physical assets:** From the overloaded agent to the new agent
 - i.e. *Plane 3* and *Plane 4*
- **New agent reports for service to CONUS Air.**

- **Result**

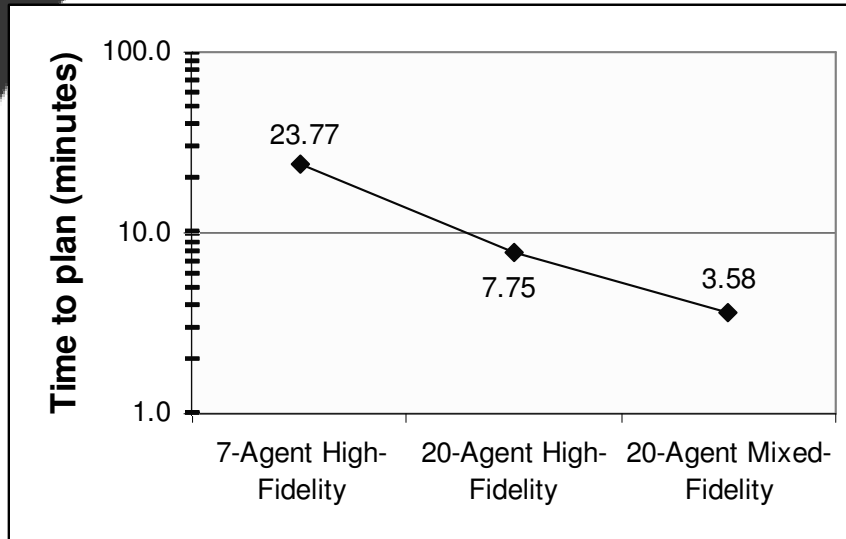
- **Superior can now do round-robin assignment**

Varying Time Horizons to Manage Computational Load

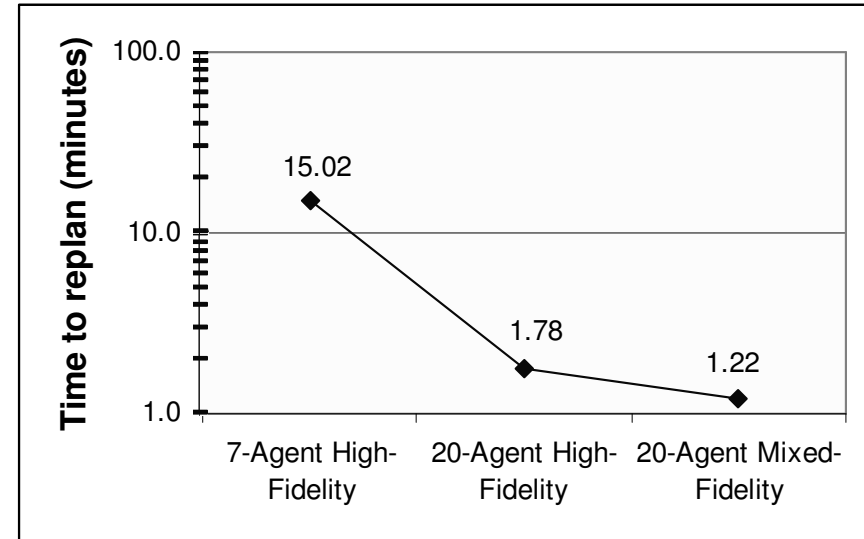
Today	Detailed Planning Individual Items	Coarse Planning Aggregate Items	Feasibility Estimate	
	High Computation	Medium Computation	Low Computation	
Now	T1: +6 Days	T2: +30 Days	+180 Days	

- **Mixed-fidelity planning**
 - Divide the future into three different levels of planning
 - Within a certain time horizon into the future, T1, individual items are included in a detailed schedule/plan.
 - Between time T1 and some more distant time in the future, T2, plans are created at a coarse level, based on large aggregates of items.
 - Beyond T2, only a feasibility estimate is made.
- Each scheduling agent concentrates its efforts on planning for the near term
- Requires the ability to perform dynamic rescheduling.
 - As coarse plans transform to detailed plans due to the passage of time or a lightened load, the agents must adapt the schedule accordingly using dynamic rescheduling

Experimental Results



Results for static schedule creation



Results for dynamic rescheduling

- Static schedule creation :
 - From United States and Europe to Afghanistan over a 180-day period
 - 22,118 equipment items and 34,058 people to move with 20 ships, 80 planes, 265 trucks
 - Seven agent vs 20 agent
 - High-fidelity planning over the entire 180 days vs performing mixed-fidelity planning
- Dynamic Rescheduling :
 - ~10% replan (1654 items and 3479 people) : Change arrival requirement from 10 to 15 days from the beginning of the mission.

- The problem of military transportation scheduling is inherently dynamic.
 - New and changing requirements, as well as unanticipated events, necessitate the continual revision of any schedule.
 - A multi-agent society presents a special set of challenges.

- Algorithmic challenges :
 - agents need to cooperate on their updates so as to maintain a globally good schedule
 - avoid uncontrolled dynamics in the society.

- Structural challenges :
 - As computational needs shift with changing requirements, the society needs to reassign computational resources

- Our techniques robustly address both types of challenges

- **BBN Technologies:**
 - <http://www.bbn.com>
- **Cougaar Agent Architecture:**
 - <http://www.cougaar.org>
- **Other Cougaar-related KIMAS'05 papers:**
 - “Watching Your Own Back: Self Managing Multi-Agent Systems”, M. Thome, T. Wright, et al
 - “Using QoS-Adaptive Coordination Artifacts to Increase Scalability of Communication in Distributed Multi-Agent Systems”, J. Zinky, S. Siracuse, et al
 - “Adaptive Optimization of Solution Time in a Distributed Multi-Agent System”, A. Fedyk, et al

