Jan 2004

Efficient DDM for HLA Federations on Myrinet Clusters

Unità di Roma 1

Andrea Santoro

Dipartimento di Informatica e Sistemistica
Università di Roma "*La Sapienza*"

Roadmap

- Context
  - Simulator integration: High Level Architecture (HLA)
  - HLA middleware: Run-Time Infrastructure (RTI)
  - RTI subsystem: Data Distribution Management (DDM)
- Our research
  - DDM Issues
  - Efficient low level implementations

Birth of a Simulator

- **In ancient times**
  Someone writes it. From scratch.

- **Today**
  Simulation models growing in complexity and details
  - **Old good way**: Someone writes it. From scratch.
    Drawback: Long and expensive
  - **Cheat**: use code written for other simulations.

High Level Architecture

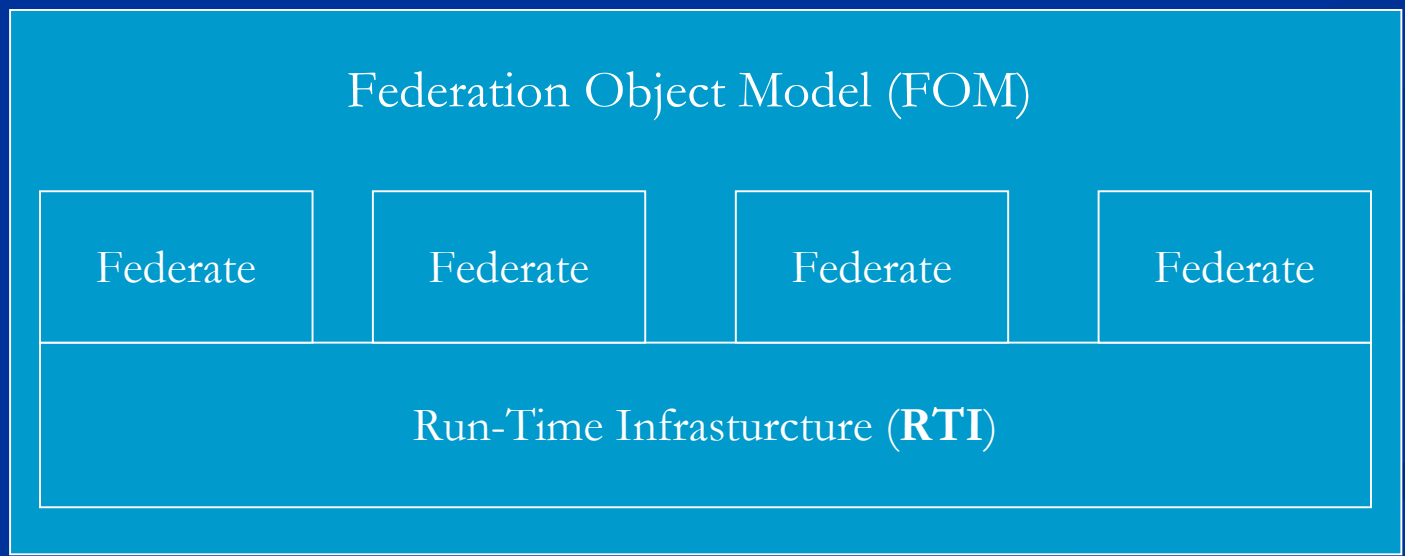
What is it?

- An Architecture Specification to "federate" Heterogeneous Simulators

- Definitions
  - Heterogeneous simulators: *Federates*
  - Simulation obtained by combination of heterogeneous simulators: *Federation*

Federation Structure

- Federates
- Static Description of Federation
- Middleware between Federates



Run-Time Infrastructure

- **Characteristics**
  - Middleware to interconnect individual federates
  - Interface specification in IEEE publications.
  - Interface specifically designed to interconnect simulation code.

  - Object oriented Paradigm:
    - Each federate seen as an object.
    - Federates treated as if they are managing simulation objects

  - ***Only the interface defined. No implementation described***

Research target: Design to implement efficient network communication

Communication between federates

- Communication Interface based on publish/subscribe paradigm

Each federate:

- Publishes the object ***instances*** it wants other federates to know.
- Subscribes to ***types*** of published objects

Potential Communication Overhead

- A federate can subscribe only to an object type

Traffic of useless messages in the RTI

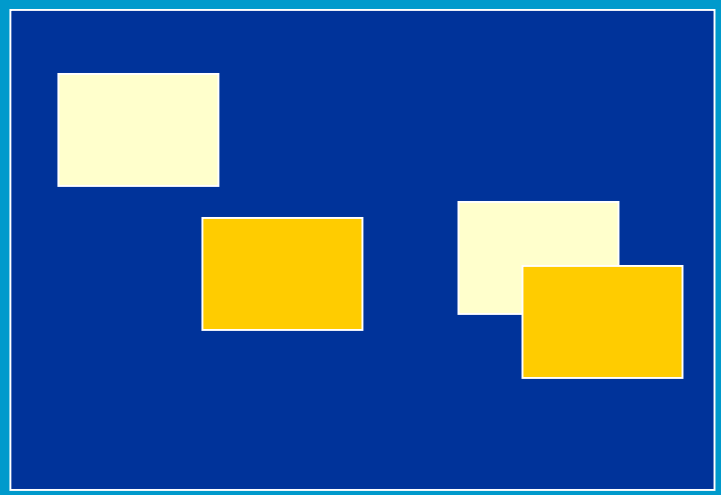
Data Distribution Management

Extension of the communication interface to tackle communication overhead

Same interface with **advanced filtering mechanisms** of messages

Simulation takes place into a simulated spatial environment

- Published object associated with "update" regions
- Federates subscribes to "subscription" regions for each object type



Each object sends updates to a federate iff

- Federate subscribed to the object's type (***same as classic RTI***)

AND

- the object's "update" region overlaps the federates "subscription" region

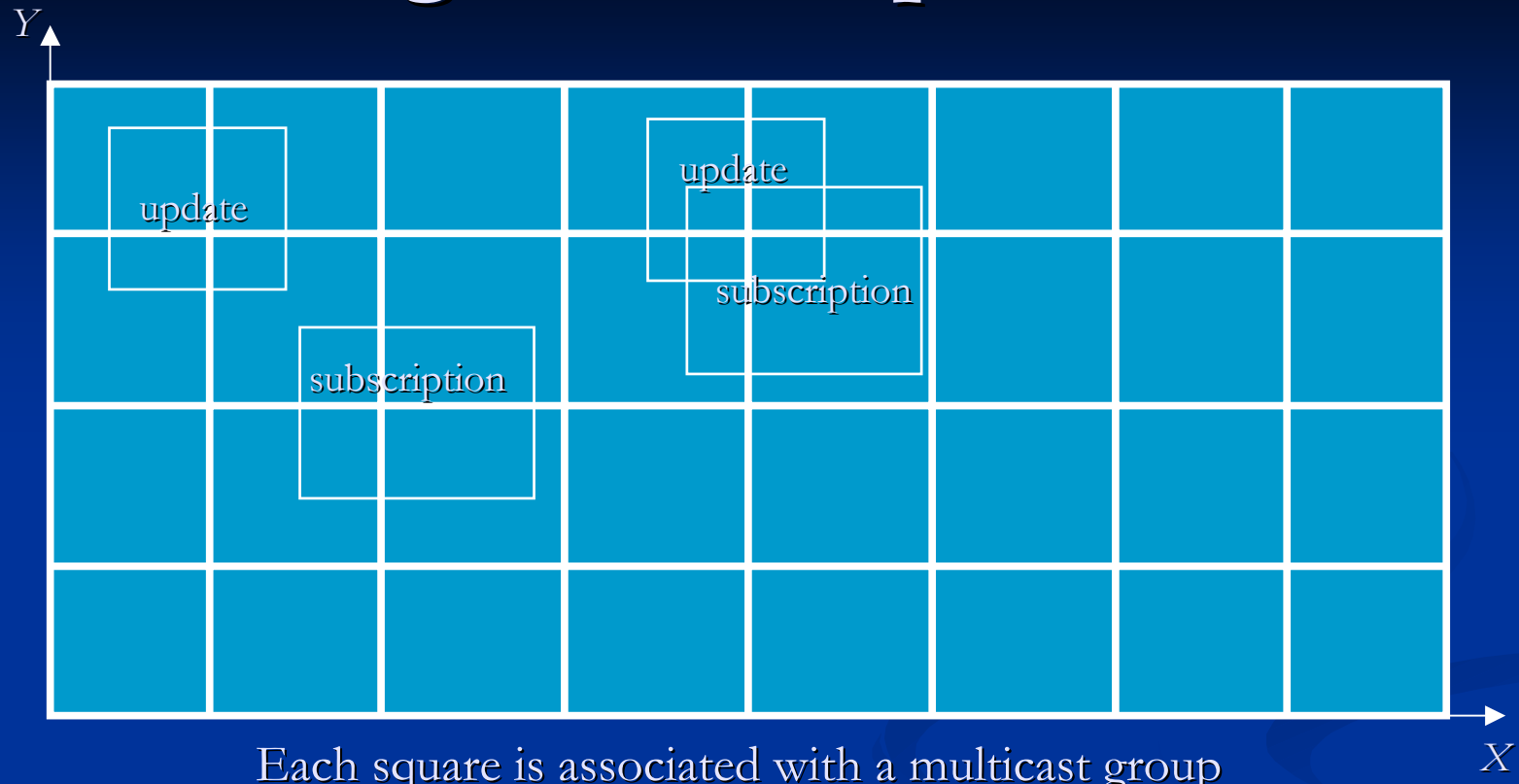
Region-based DDM Implementation

Straightforward implementation of the DDM specification

Anytime a region is modified DDM subsystem must compare it with all the opposite kind of regions to check wether any overlap takes place or is removed.

Computation Overhead

Gridding DDM Implementation



Each square is associated with a multicast group

Problems

- Useless messages
- Duplicated Messages

➢ Network traffic
➢ Discrimination logic
➢ Difficult to model

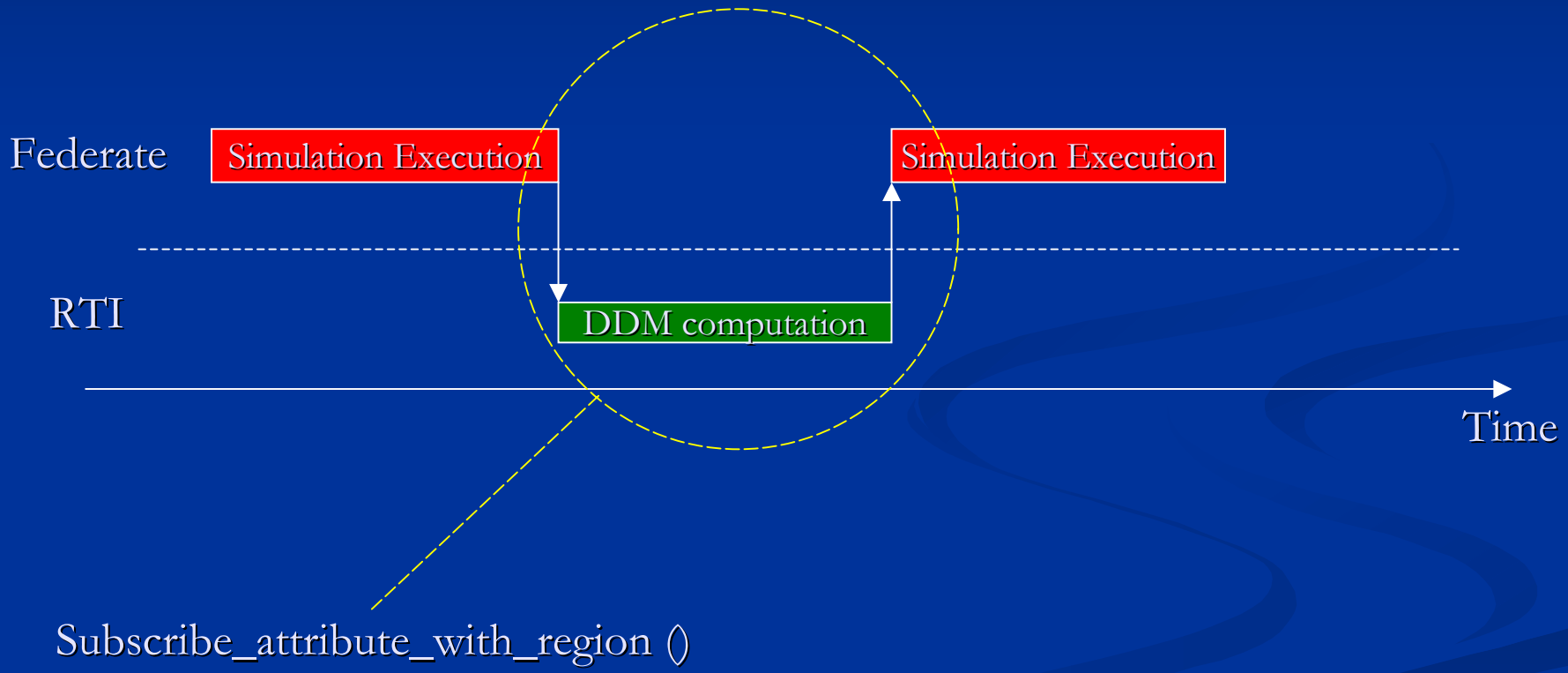
Our Research

- Increase Region-Based performance

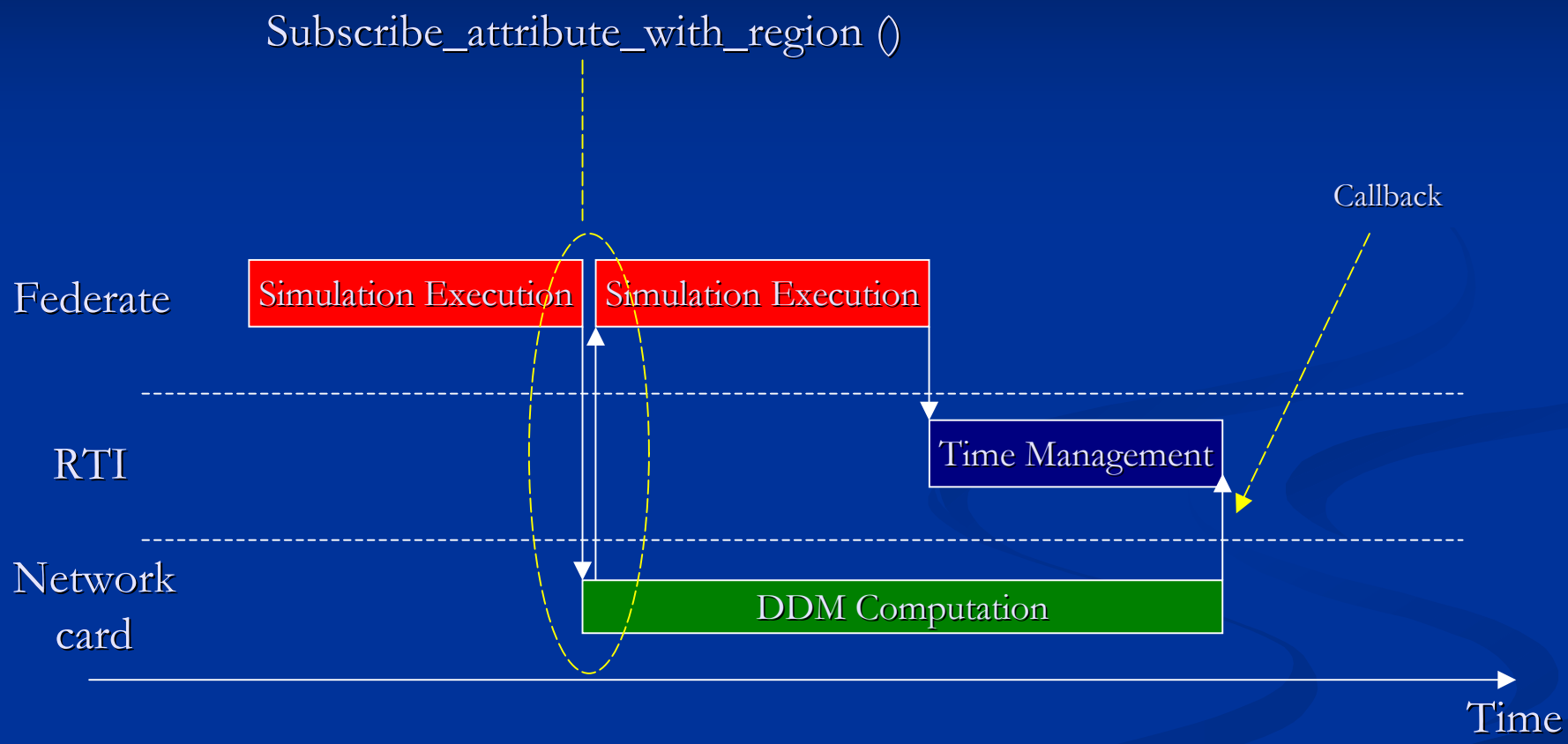
How?

- Execute DDM computation on the myrinet processor.
- Multicast Mechanism that avoids duplicated messages
- Move multicast to levels 2/1 in the ISO/OSI stack to avoid multiple message copies across the system bus.

Classic DDM Subsystem

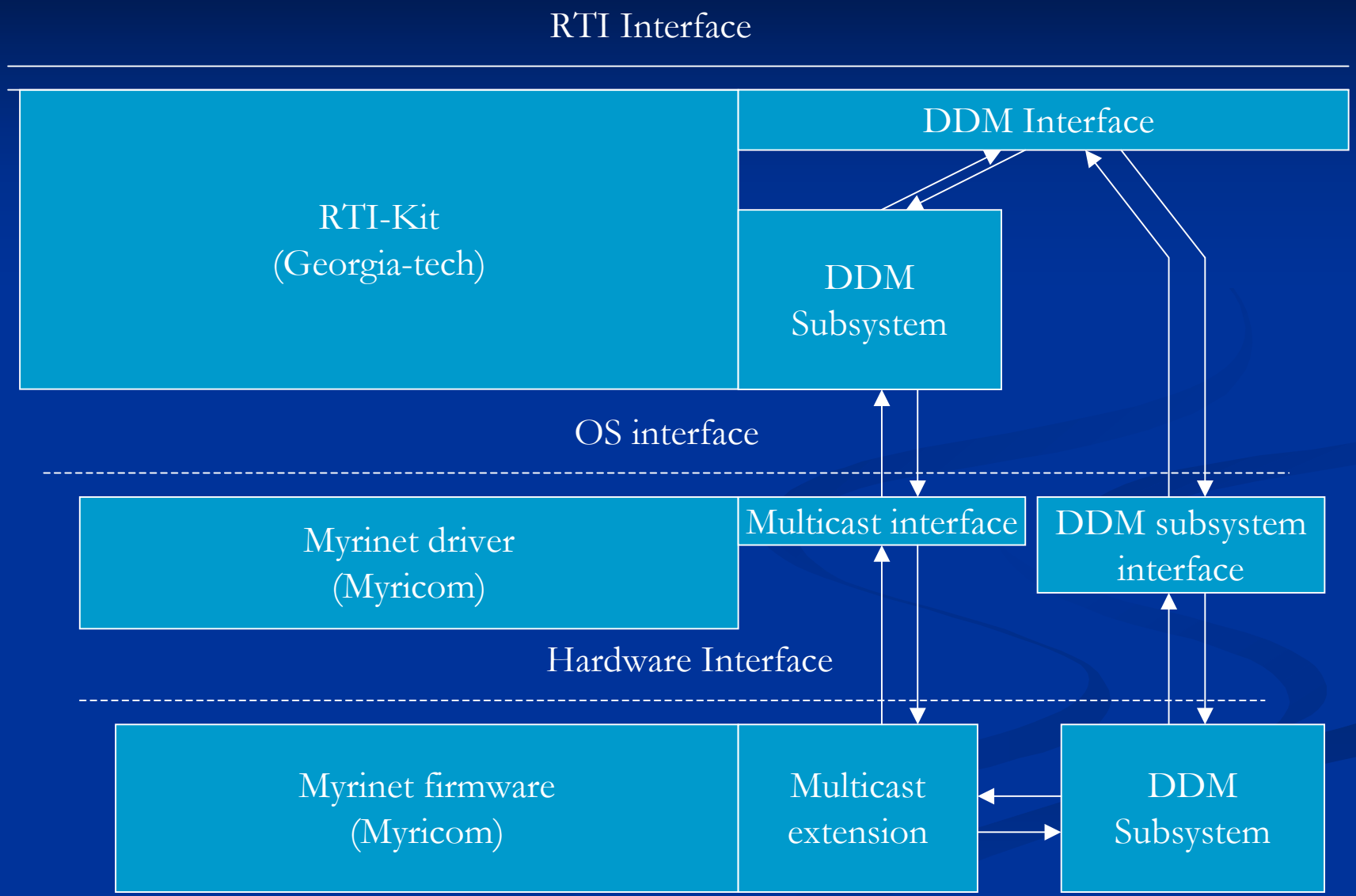


DDM Subsystem on myrinet processor



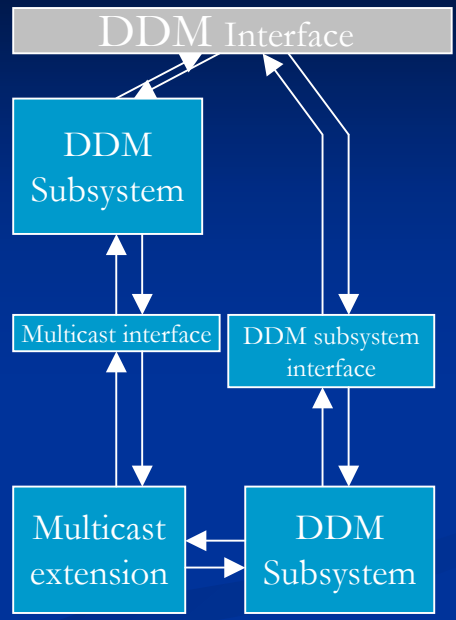
DDM Computation is executed together with simulation and RTI executions

Overall Software Structure



Where are we?

- Basic Implementation completed



Where do we go from here?

- Experiments between the two execution flows.
- Content-aware firmware to avoid control message internal routing