Real Time Support in Middleware

Marisol García Valls

Drequiem Lab. http://www.it.uc3m.es/drequiem

Departamento de Ingeniería Telemática
Universidad Carlos III de Madrid
mvalls@it.uc3m.es

Index of presentation

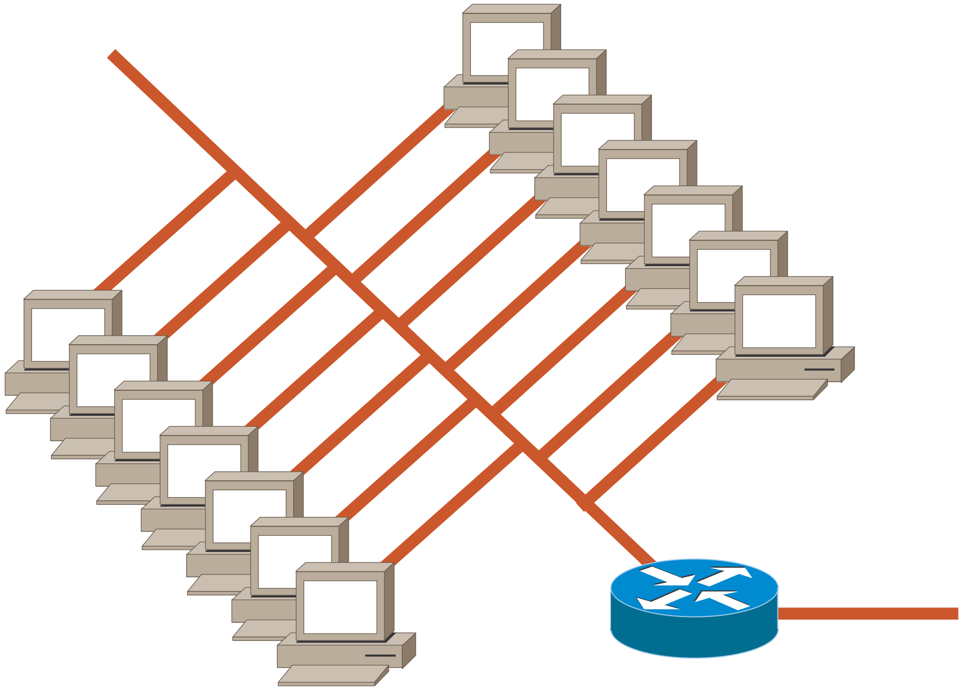
- Our lab
- Our view of networks for reconfigurable embedded systems
- Middleware layer
  - QoS-based reconfigurability of embedded systems
  - Dynamic composition of service-based real-time applications
  - Language-based middleware solutions
- Conclusions

Our lab: Drequiem Lab

Drequiem Lab at ***Universidad Carlos III de Madrid***

- Marisol GARCÍA VALLS (Head of Lab)
- Prof. Carlos DELGADO KLOOS (Gast Group)
- Pablo BASANTA VAL
- Iria ESTÉVEZ AYRES
- 5 Master Thesis

Our View



Node:

- Processing capacity

- Memory

- RT-communication, power, etc.

Functional Reconfiguration:

- Code paths

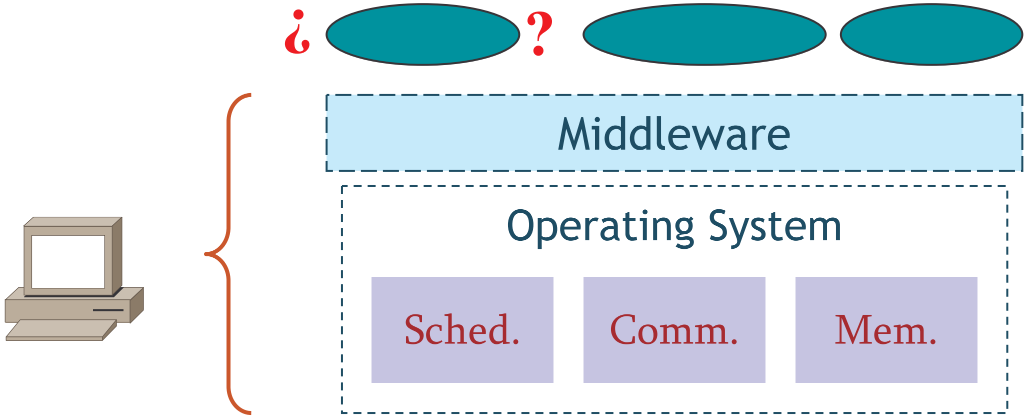
- Service profiles, etc.

Problems:

- Portability (VM) (arch).

- SW deployment (arch).

- Communications, etc.

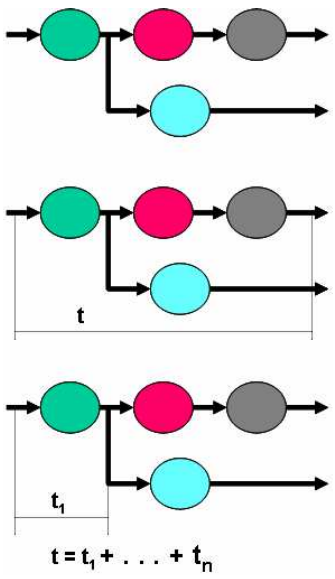


¿ ?

Middleware

Operating System

Sched. Comm. Mem.

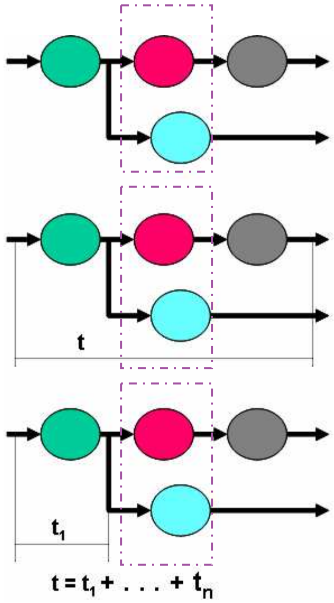


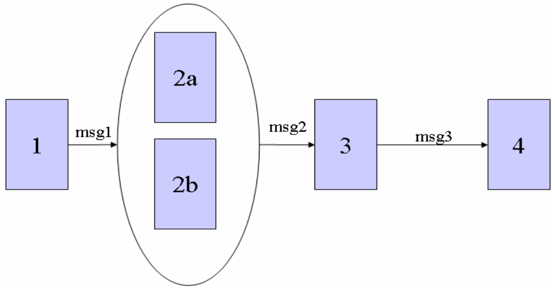
Applications

- Real-time requirements
- Networked

} message delivery and processing bounds (soft)

- Based on services
- Dynamically reconfigurable: QoS profiles
- Portability requirements: mobile (portable) code with VMs or appropriate interface abstractions





Support at middleware level

- When refering to middleware in real-time systems, the traditional focuse is on:
  - Performance
  - Network protocols
  - Real-time language interfaces
  - Non real-time centered designs of software architectures

Our focus

- Real-Time Support in Middleware

  * Dynamic composition of service-based real-time applications

  * QoS-based dynamic application management and reconfiguration

    1. Minimal infrastructure middleware

    2. Appropriate language interfaces (Object Oriented)

    3. Appropriate characterisation of services/applications

    4. Integration of real-time network protocols to assure real-time communication

Some problems and solutions

- Manage QoS of applications
  - Trading-off resources for the quality of the output
  - Algorithms for maximising output quality
  - Architectures of QoS manager entities

HOLA-QoS

- Basic infrastructure middleware (language-based middleware)
  - Memory management (in RTSJ)
  - Combine it with scheduling
  - Extensions to the language
  - Reimplementation of RMI with enhanced real-time support

Drequiemi: RT-RMI

- Dynamic composition and reconfiguration of service-based applications
  - Service composition algorithms
  - Service and application characterization
  - Appropriate architectures to include all necessary processes (discovery, download, etc.)

CoSeRT

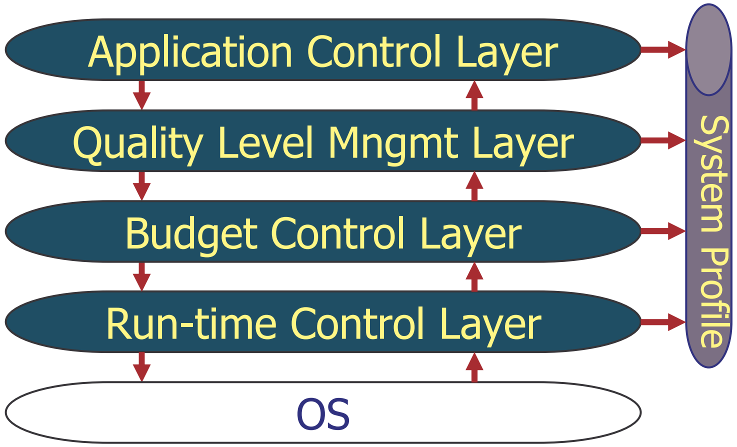
- Integration with real-time communications
  - Unified framework for the above
  - Integration of network protocols (RT&RMI-FTT; CoSeRT&FTT; and CoSeRT&RT-RMI&FTT)

QoS management

- HOLA-QoS Architecture

  (Homogeneous Open Layered Architecture)

  - Centralised (and Multiprocessor)
  - Applications based on tasks with different profiles
  - Objective: maximise quality of Multimedia applications on-line
  - Dificulty: application characterisation, run-time mode management
  - Implementation: TriMedia Processor on pSoSystem



Application Control Layer

Quality Level Mngmt Layer

Budget Control Layer

Run-time Control Layer

OS

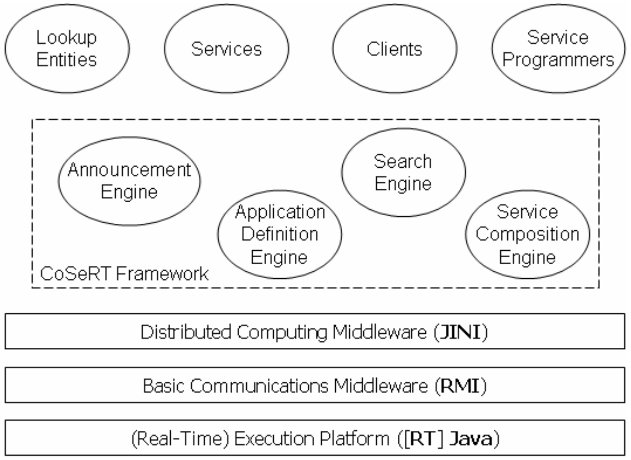
System Profile

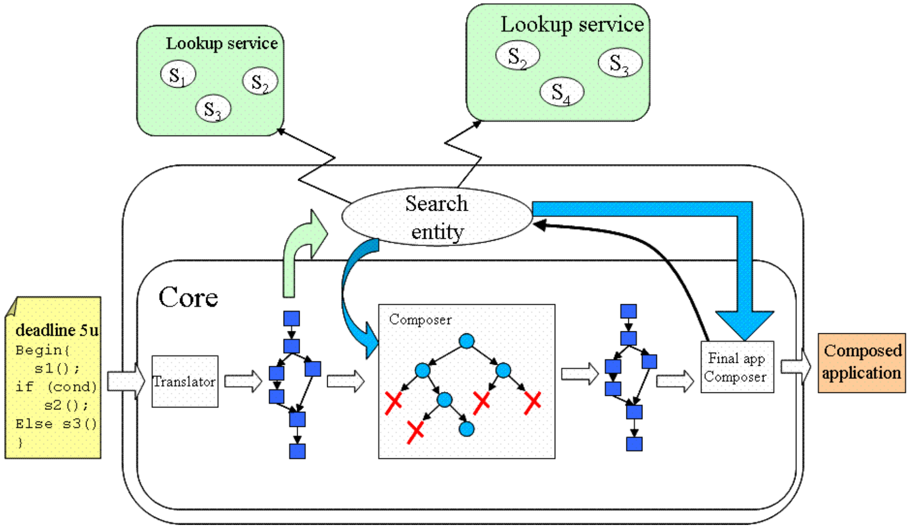
Dynamic composition of service-based applications

- CoSeRT

  (Composition of Service-based Real-Time Applications)

  - Distributed applications based on services
  - Objective: dynamic composition and reconfiguration
  - Off-line service discovery
  - Implementation: Jini/RMI/JVM


Lookup Entities
Services
Clients
Service Programmers
Announcement Engine
Application Definition Engine
Search Engine
Service Composition Engine
CoSeRT Framework
Distributed Computing Middleware (JINI)
Basic Communications Middleware (RMI)
(Real-Time) Execution Platform ([RT] Java)

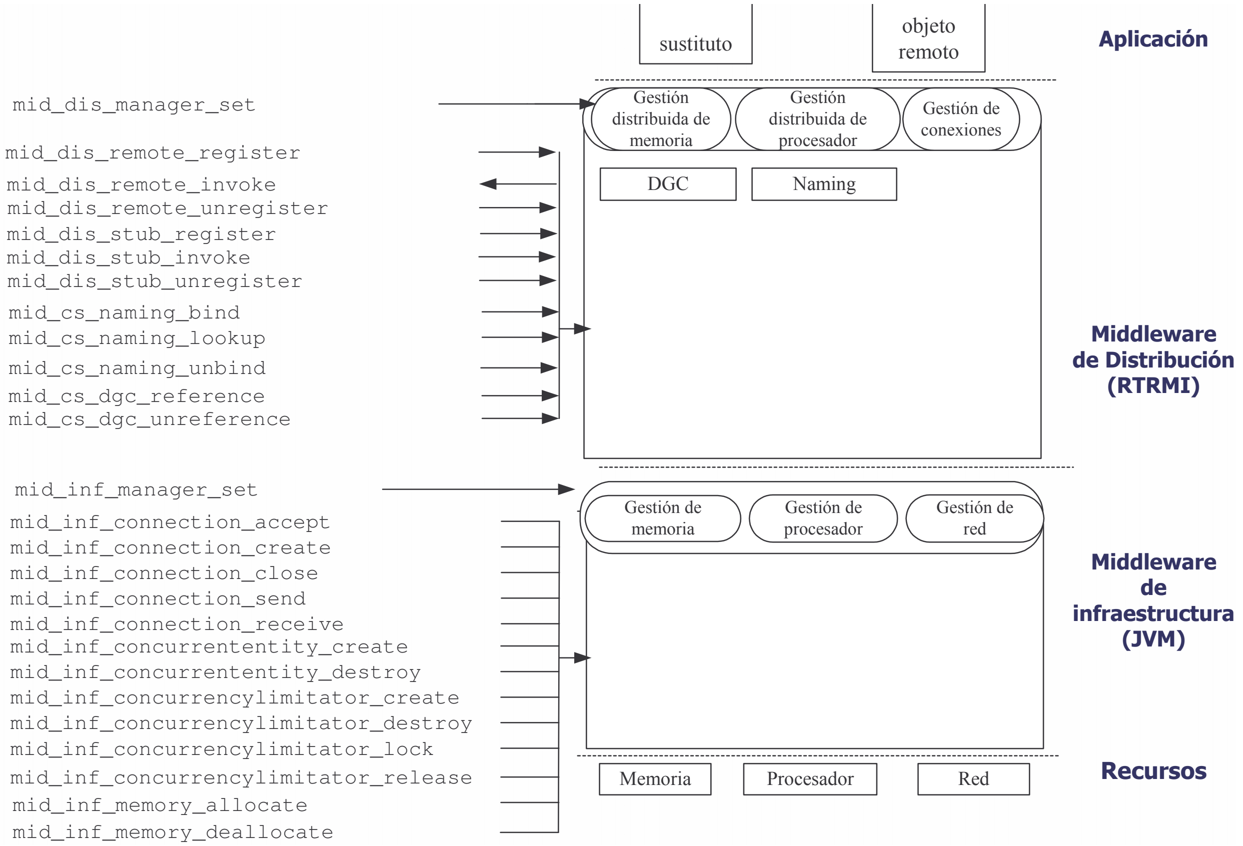


```
Begin{
  s1();
if (cond)
   s2();
Else s3()
}
```

Real-time languages: RTSJ

πρωιτιδα
Nombre
de capa

- Distributed Real-Time Java
  - Architecture for Real-Time RMI (based on SUN RMI)



sustituto

objeto remoto

Aplicación

mid_dis_manager_set

mid_dis_remote_register

mid_dis_remote_invoke

mid_dis_remote_unregister

mid_dis_stub_register

mid_dis_stub_invoke

mid_dis_stub_unregister

mid_cs_naming_bind

mid_cs_naming_lookup

mid_cs_naming_unbind

mid_cs_dgc_reference

mid_cs_dgc_unreference

Gestión distribuida de memoria

Gestión distribuida de procesador

Gestión de conexiones

DGC

Naming

Middleware de Distribución (RTRMI)

mid_inf_manager_set

mid_inf_connection_accept

mid_inf_connection_create

mid_inf_connection_close

mid_inf_connection_send

mid_inf_connection_receive

mid_inf_concurrententity_create

mid_inf_concurrententity_destroy

mid_inf_concurrencylimitator_create

mid_inf_concurrencylimitator_destroy

mid_inf_concurrencylimitator_lock

mid_inf_concurrencylimitator_release

mid_inf_memory_allocate

mid_inf_memory_deallocate

Gestión de memoria

Gestión de procesador

Gestión de red

Middleware de infraestructura (JVM)

Memoria

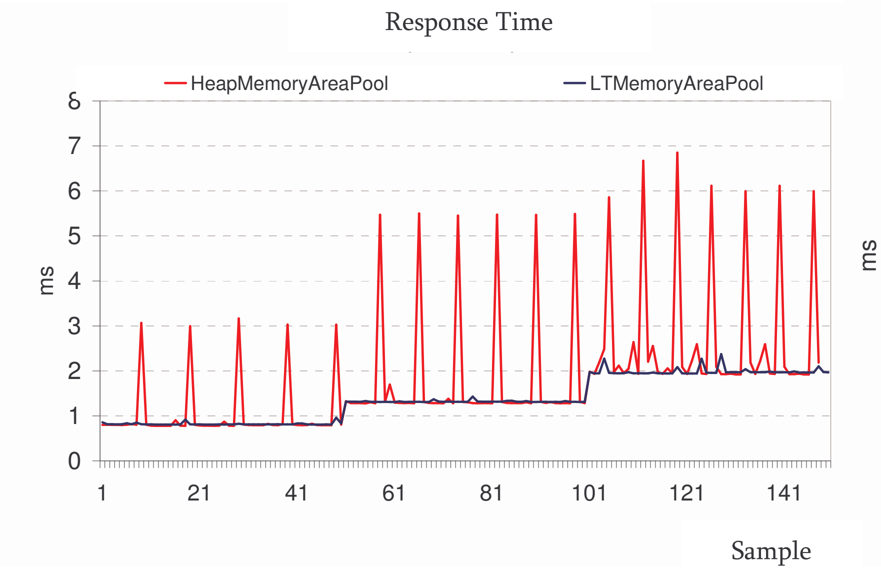
Procesador

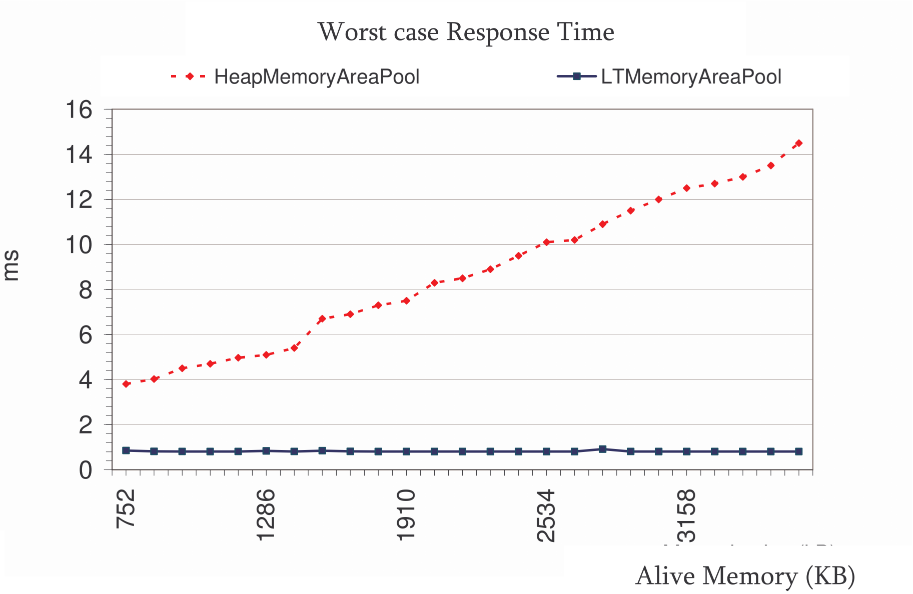
Red

Recursos

  - Extensions for Real-Time Specification for Java
    - AGCMemory, NoHeapRemoteObject, ExtendedPortal, RealTimeThread++

Real-time languages: improvements to middleware





Time

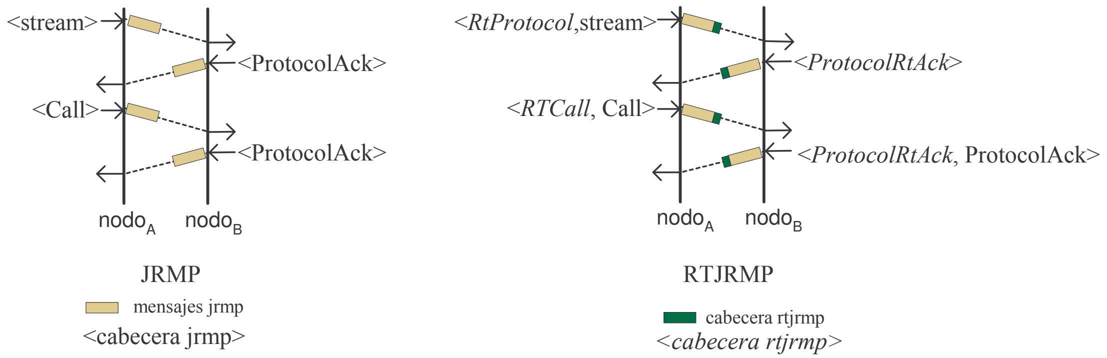
Garbage Colector	**peaks +5 ms**
Regions/Scopes	plain

Increase of memory usage

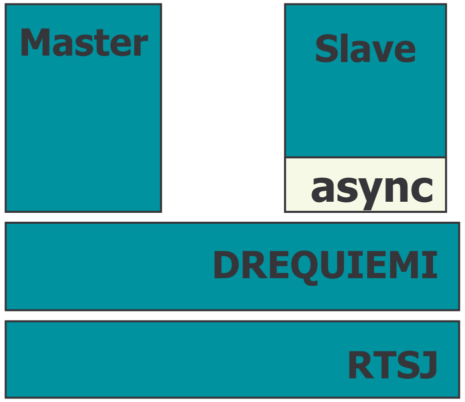
Garbage Colector	**linear (+ 3,1 ms/kb)**
Regions/Scopes	plain

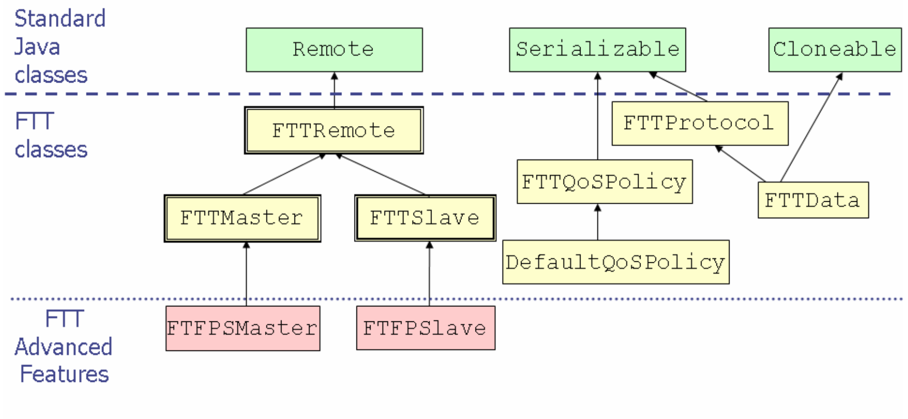
Integration of Real-Time Network Protocols

- New protocols (RTJRMP) to transfer real-time parameters.



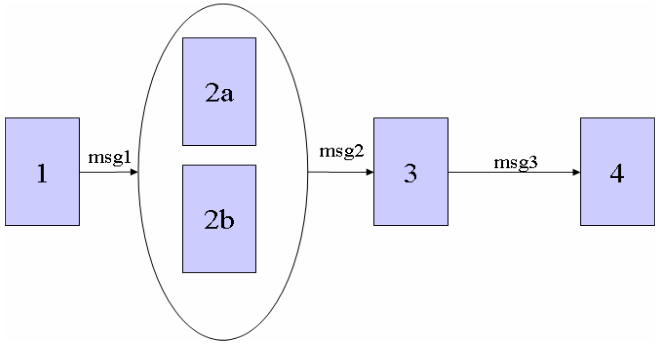
- Current approaches for DRTSJ are silent about clock synchronization

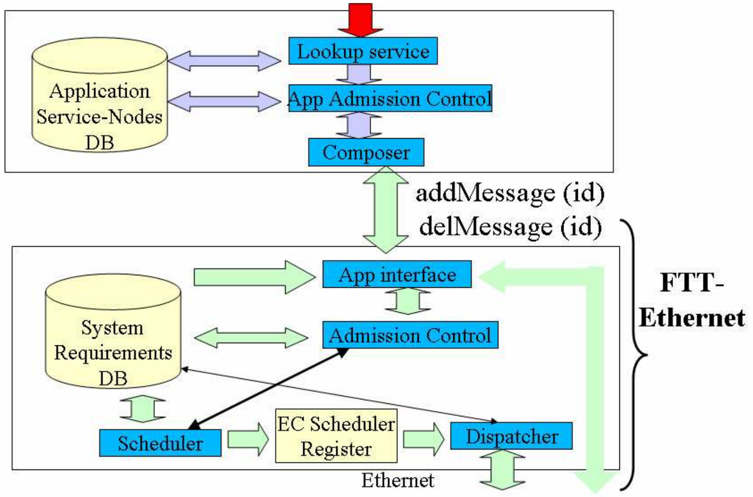


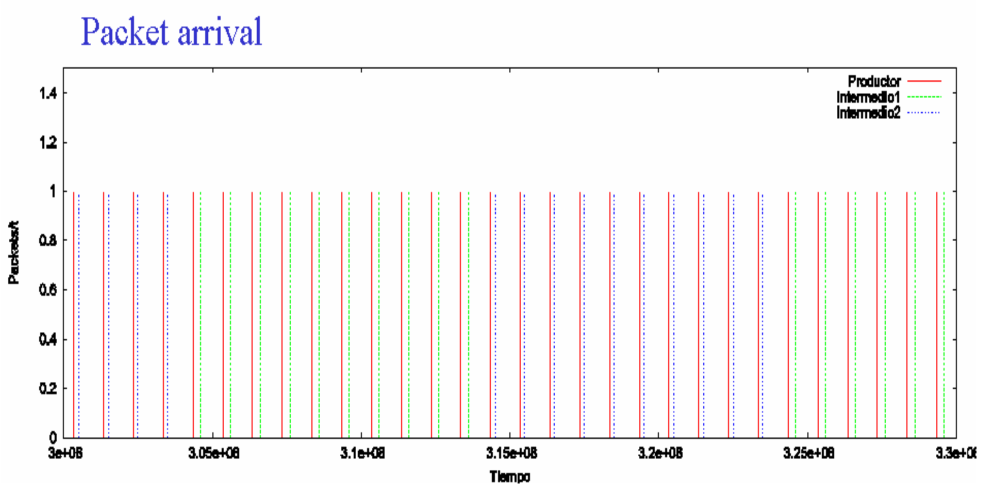


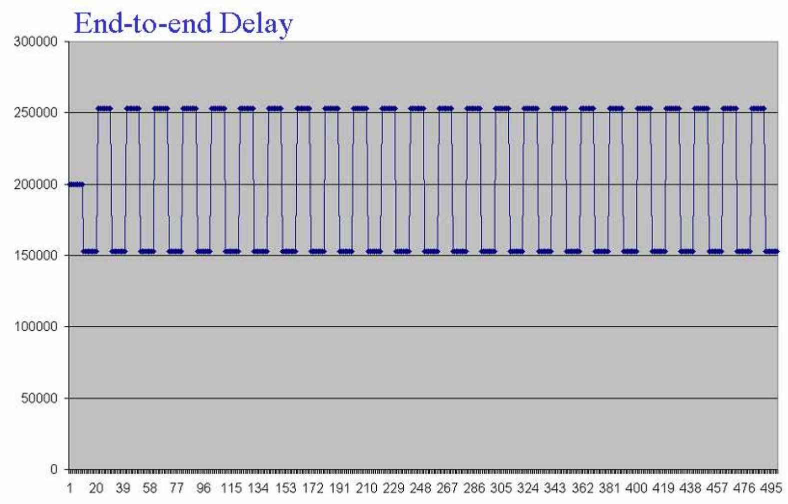
Real-time communicacion for dynamic composition

- Integration of CoSeRT with FTT






Packet arrival


End-to-end Delay

Conclusions

- Dynamic reconfiguration of distributed embedded systems requires
  - Middleware with true real-time support
  - Frameworks and architectures for dynamic composition of functionality

- Real-time support in middleware requires working at several levels

- At Drequiem Lab, **Real-Time Distributed Embedded Systems** (Infrastructure Middleware)

  - QoS management architectures
    - **HOLA-QoS**: QoS manager for Multimedia Embedded Systems

  - Dynamic composition and reconfiguration
    - **CoSeRT**: Architectural framework for dynamic composition

  - Language-based middleware: Real-Time Java (RMI)
    - Memory management in RTSJ; extensions to the language
    - **DREQUIEMI**: Reimplementation of RMI with enhanced real-time support

  - Integration with real-time communications
    - Integration of CoSeRT with FTT
    - Integration of FTT-RMI

Future lines

- Object Oriented Infrastructure Middleware built according to real-time concepts
- Memory management techniques combined with scheaduling in middleware (interfaces + implementation)
- Further integration of network protocols and adaptation to better suit the basic middleware infrastructure needs
- Development of QoS-based architectures and frameworks for dynamic composition and reconfiguration of service-based applications