Integrating middleware for timely reconfiguration of distributed soft real-time systems with Ada DSA

Marisol García-Valls mvalls@it.uc3m.es
Felipe Ibáñez-Vázquez fibanez@pa.uc3m.es

Distributed Real-Time Systems Laboratory (DREQUIEM)

Universidad Carlos III de Madrid

Universidad
Carlos III de Madrid

Ada-Europe 2012. Stockholm, June 11-15, 2012

Outline

- Objective of this work
- iLAND middleware
  - What is iLAND?
  - Architecture
- Adjusting to Ada distribution specifics
- Distribution philosophy of Ada DSA
- Ada DSA integration
- Conclusions and future work

Objective of this work

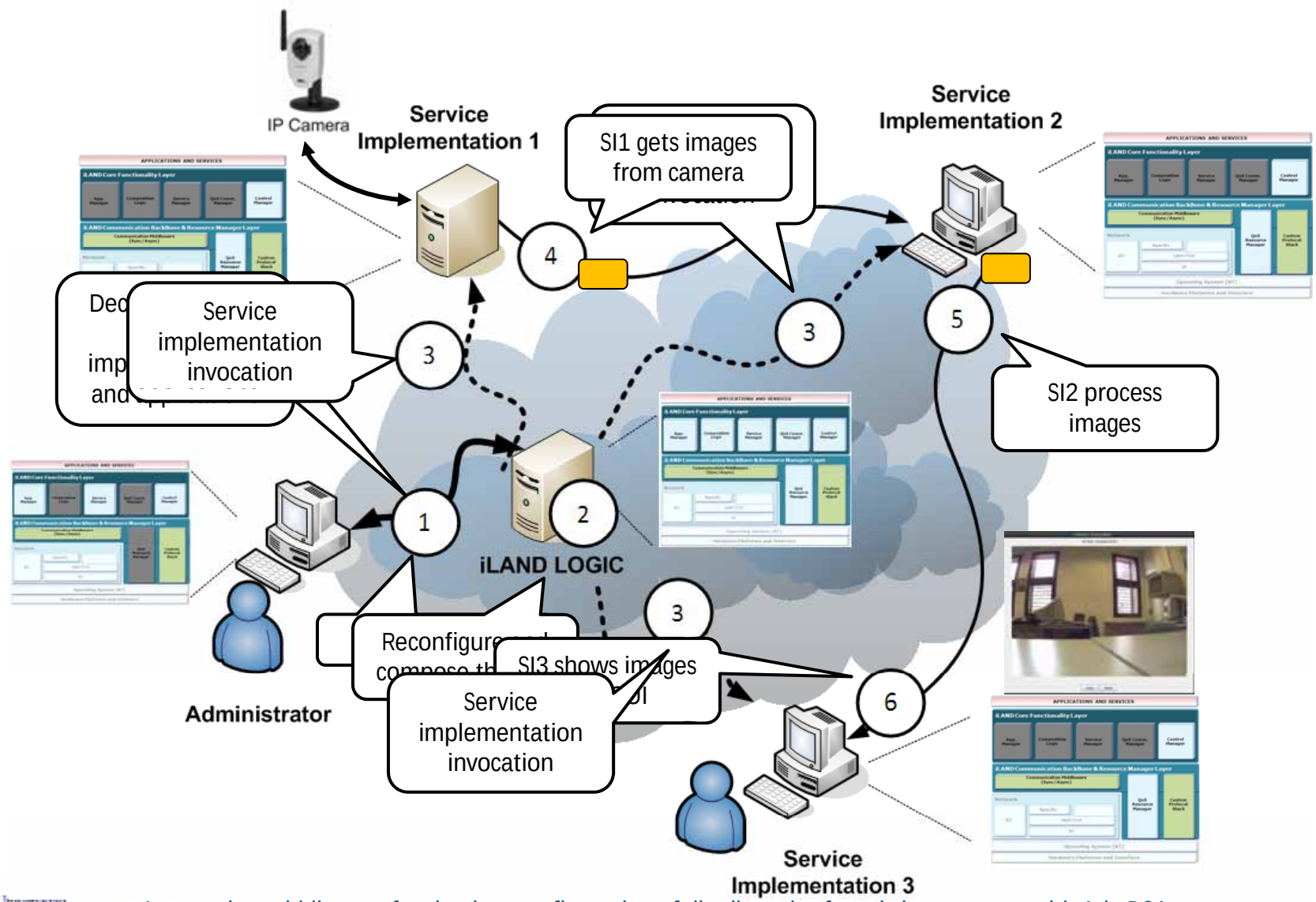
- **Improve** the **temporal predictability** of **iLAND**, a soft-real time reconfigurable middleware, through the integration **with Ada DSA**.
  - Interoperability achieved by the definition of an Ada Common Bridge for iLAND.
  - Proof of concept in PolyORB/QNX environment.

What is iLAND?

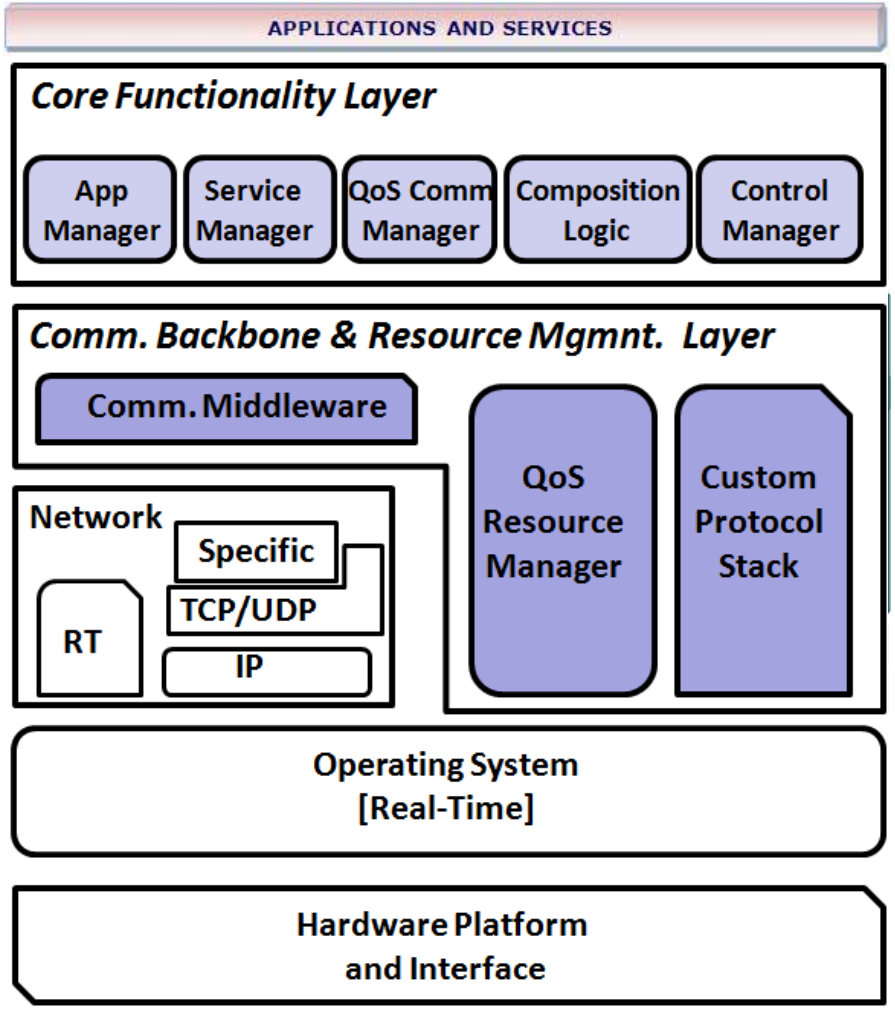
- M**I**dd**L**ew**A**re for Deterministic Dynamically Reconfigurable **N**etworke**D** Embedded Systems
  - Enhanced middleware to reconfigure in bounded time soft real-time applications
    - Relies on the usage of different middleware technologies.
    - Provides high level services to traditional middleware.
    - Reconfiguration and QoS dynamic composition of services in bounded time.
    - Service Oriented Architecture (SOA).
    - Transparency and interoperability between different devices and technologies.

iLAND

Example environment with an application

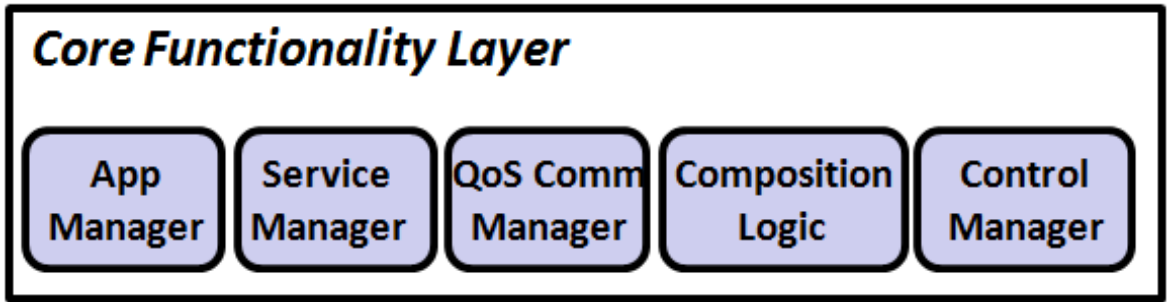


iLAND
Architecture



iLAND Architecture

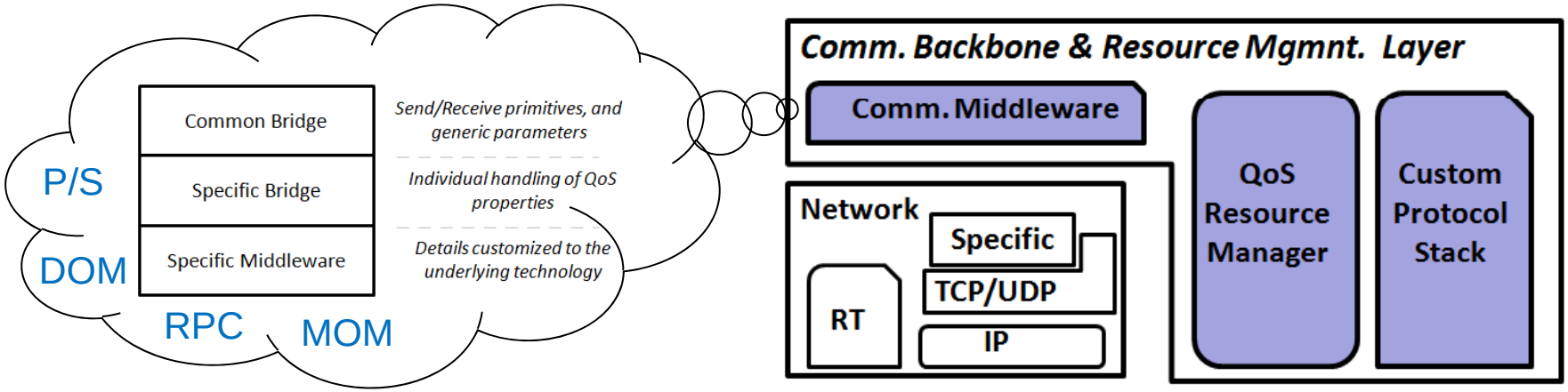
Core Functionality Layer



- *Service Manager*
  - Creation, deletion and updating of services.
- *Application Manager*
  - Creation, deletion and updating of applications.
- *Composition Logic*
  - Compose services in a limited time based on QoS parameters.
- *Control Manager*
  - High-level reconfiguration.
- *Communication QoS manager*
  - QoS communications.

iLAND Architecture

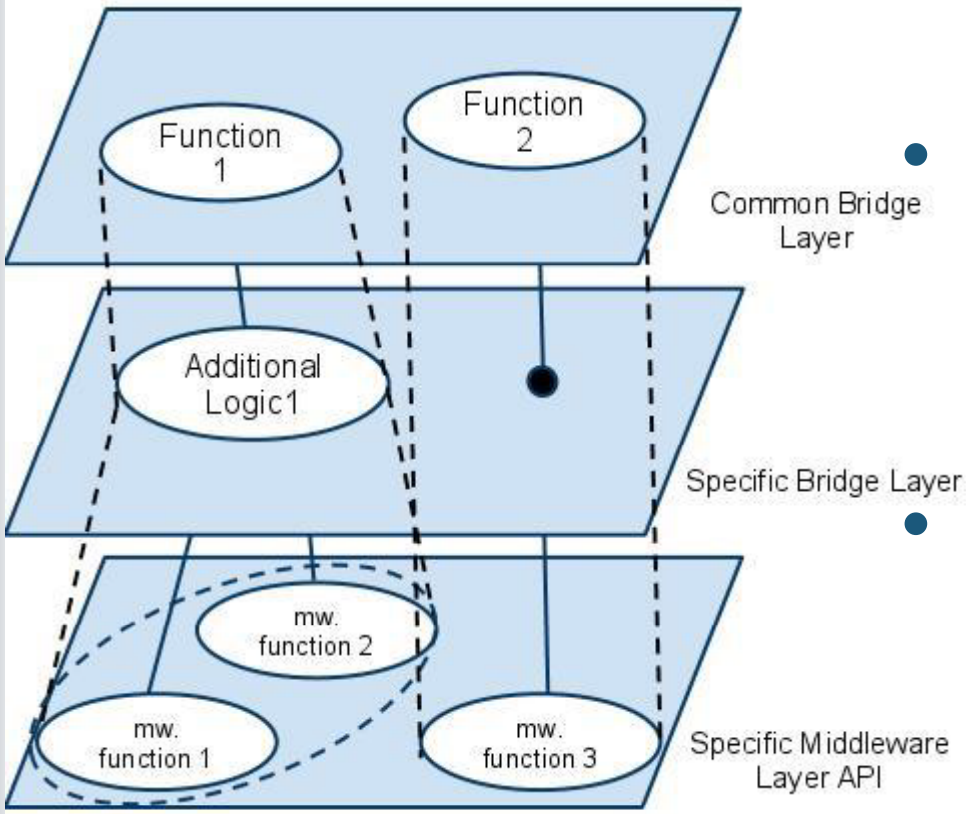
Communication Backbone Layer



- *QoS Resource Manager*
  - Operating System Resources
    - Thread management.
    - Schedulability analysis.
    - Resource reservation.
    - Monitoring.
- Communication resources
  - Holistic approaches – *Custom Protocol Stack*
  - Middleware approaches – *Communication Middleware*
    - Enable transparency of the different middleware technologies with the upper levels.

iLAND Architecture

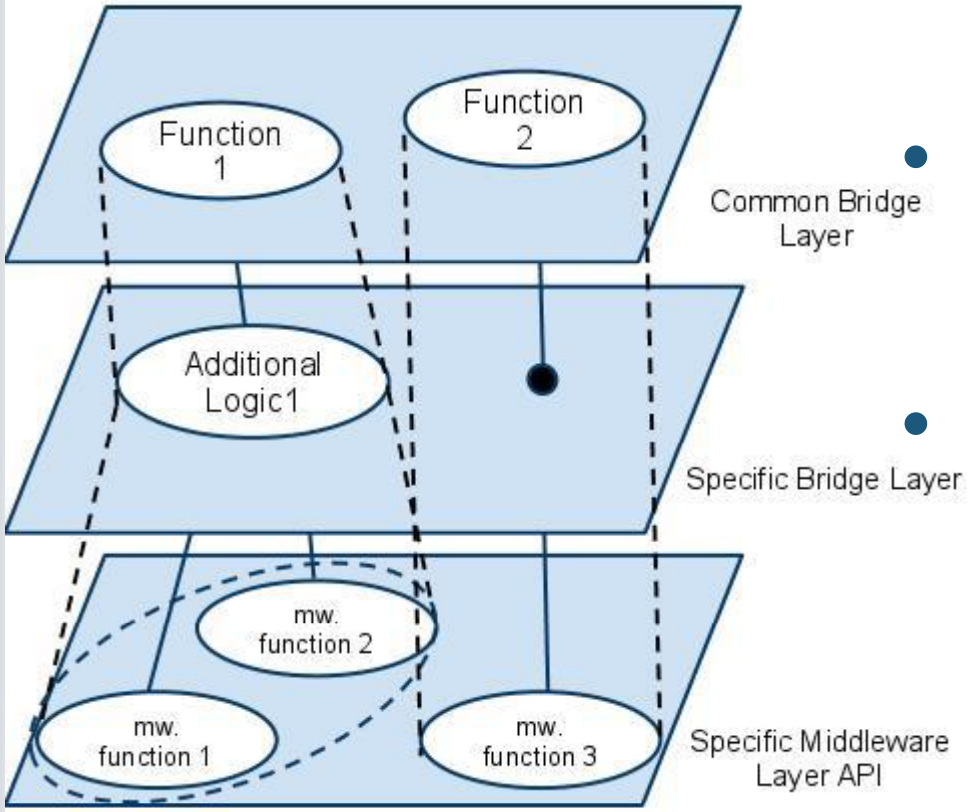
Common Bridge Architecture



- Common Bridge interface
  - Common API for application and other middleware components
  - Simple and technology independent
- Specific Bridge Layer
  - Transforms the provided data to independent data types of the *common bridge*
  - Creates the internal structures involved in the communication
  - Additional logic
- Specific Middleware Layer
  - Technology dependant part to map and/or adapt communication paradigms
  - The communication requirements influence the technology selection

iLAND Architecture

Adjusting to Ada distribution specifics

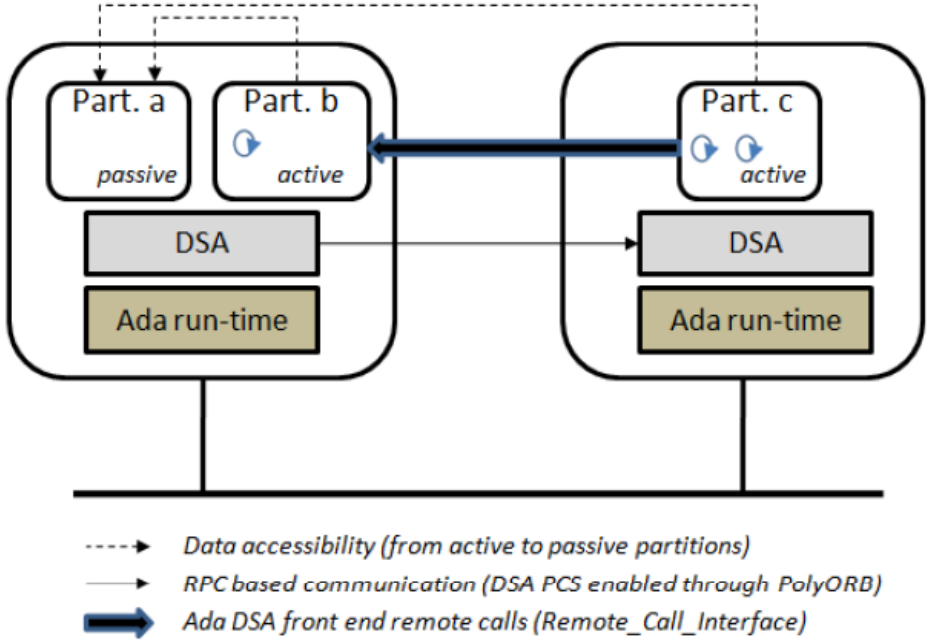


- Common Bridge interface
  - Synchronous communication
  - Asynchronous communication thanks to DSA Asynchronous pragma
- Specific Bridge Layer
  - C: POSIX socket
  - Ada DSA: GNAT.Sockets library
  - Simplified binding
- Specific Middleware Layer
  - Client–server model
    - Ada DSA partitions
    - Remote_Call_Interface pragma
  - PolyORB's PCS
    - Calling stubs
    - Receiving stubs

Distribution philosophy of Ada DSA

Brief review

- Pragmas based on the classical distribution paradigms
- Abstract interaction between partitions thanks to the PCS
  - Distributed shared data
  - Remote Procedure Calls (RPC)



Ada DSA integration

Selection of technologies

- AdaCore The GNAT Pro Company **PolyORB**
  - Ada 95 Distributed System Annex
  - Suitable for software development in distributed applications because it is licensed by AdaCore
  - Free version available
- QNX
  - Real-Time Operating System
  - PolyORB could be integrated with QNX

Ada DSA integration

PolyORB/DSA over QNX: Cross compiler

- AdaCore does **not support** PolyORB over QNX
- **Needed** a **Cross compiler** of the **DSA** on a Linux host **for QNX** targets **using PolyORB**
  - **Developed** in collaboration between Universidad Carlos III de Madrid and the Warsaw University of Technology
- **Precompiled packages** have been created **ready to be linked**
  - **Used** in Ubuntu 10.04 LTS and QNX 6.5.0

Ada DSA integration

PolyORB/DSA over QNX: Cross compiler

- No integrated development environments for Ada on QNX.
- Native compilation vs. cross compilation.
- Para-virtualization.
- Prerequisites
  - **Cross development** QNX 6.5.0 Linux host **tools.**
  - **GNAT** 2010 **compiler** for Linux.

Ada DSA integration

PolyORB/DSA over QNX: Ada cross compiler

- **Source:**
  - **Core Development Tools** from **QNX**.
- **Bound** the **sources:**
  - Provides specialization of the **Ada runtime** for the **QNX** platform.
  - Fixes **static library support** for the cross **QNX** build tools.
  - Fixes support for the **`GNAT.Sockets` under QNX.**
  - Spawning of a child processes in QNX.
- **Built** the **cross compiler:**
  - Native Ada compiler.

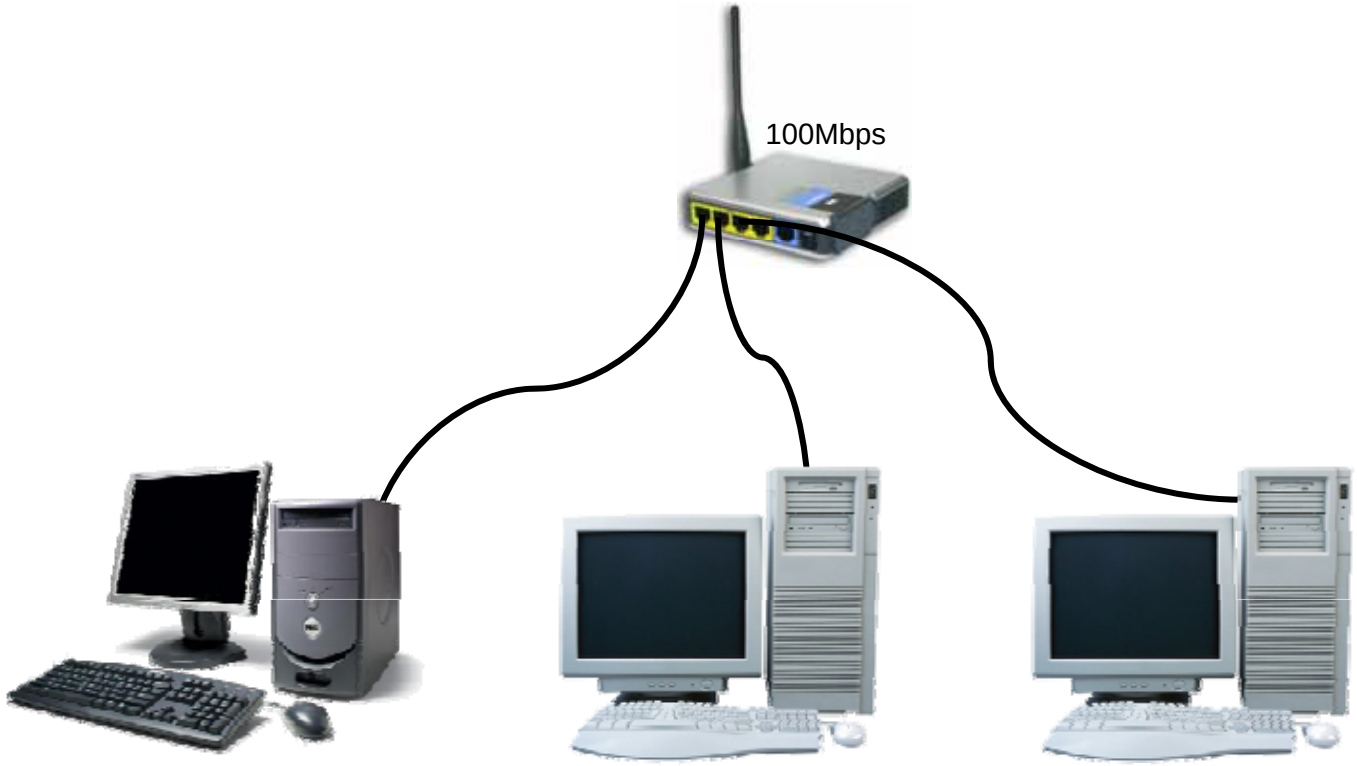
Ada DSA integration

PolyORB/DSA over QNX: PolyORB cross tools

- **Source:**
  - PolyORB compatible with the GCC sources released by the QNX (Release 139901).
    - Fully functional in respect of the DSA personality.
- **Bound** the **sources:**
  - Style errors (Errors and warnings).
- **Built** the PolyORB **cross tool**.
  - Native Ada compiler → PolyORB gnatdist.
  - Cross Ada compiler → PolyORB name server.

Ada DSA integration

Ada DSA environment



Host: Ubuntu 10.04 LTS amd64
IP address: 163.117.141.A

Target: QNX 6.5.0
CPU: AMD Athlon XP 1600+
RAM: 256MB
IP address: 163.117.141.B
Name server: po_names
Partition: client_partition

Target: QNX 6.5.0
CPU: Intel P4 1.8GHz
RAM: 256MB
IP address: 163.117.141.C
Partition: server_partition

Conclusions

- **iLAND** middleware provides **flexibility** and **timely reconfiguration** for **soft RT** distributed systems.
- **Ada** technology backbone allows to **increase** the temporal **determinism** of **iLAND**.
  - Higher efficiency and control over local execution and remote communications
- **Ada DSA – iLAND** middleware interoperable **architecture defined**.
- **Cross compiler** of PolyORB/DSA over QNX **developed**.
  - Available source code at GitHub.

Future work

- Integration of Ada DSA as a core middleware technology in iLAND
  - Soon available source code of the iLAND Reference Implementation created by UC3M at SourceForge


GOBIERNO
DE ESPAÑA
MINISTERIO
DE INDUSTRIA, TURISMO
Y COMERCIO
plan
avanza2

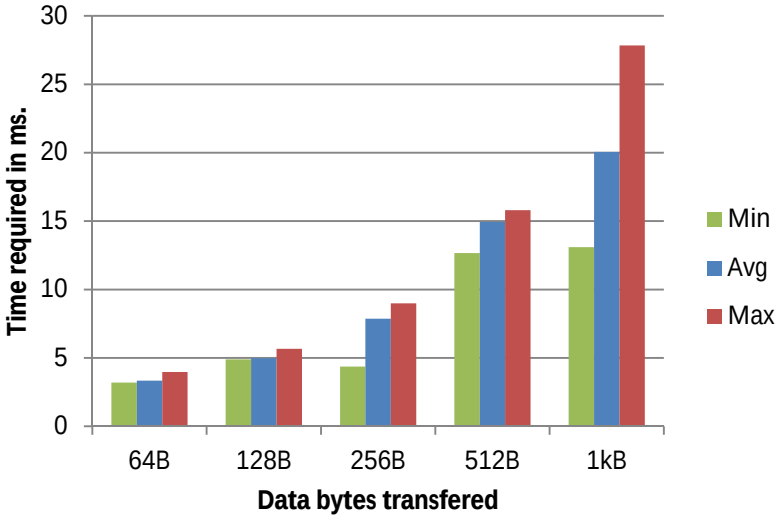

ARTEMIS
LAND

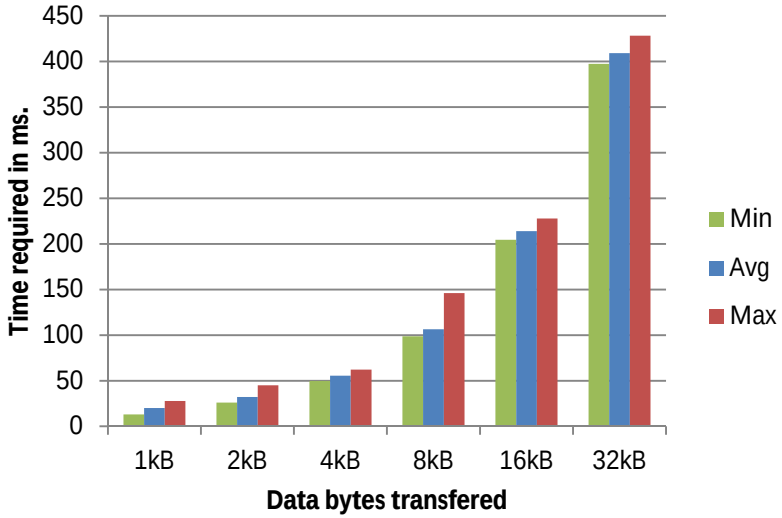
Thank you


Universidad
Carlos III de Madrid

Ada DSA integration

Performance when using PolyORB





- Many elements can be configured in order to increase the throughput
  - Build options
  - Tasking policies
  - Transport parameters