Integral support for embedded, distributed open real-time systems (THREAD)
TIN2005-08665-C03

Juan Antonio de la Puente *
Universidad Politécnica de Madrid (UPM)

J. Javier Gutiérrez †
Universidad de Cantabria (UC)

Ismael Ripoll ‡
Universidad Politécnica de Valencia (UPV)

Abstract

THREAD is a research project aimed at providing an integral support for the development of embedded real-time systems. The support elements that are being developed in the project include a family of inter-operable execution platforms, their inter-connection mechanisms, and the applicable architecture and design methodologies. Some new-generation application domains for this kind of systems are also being explored.

This kind of integral support deals with all the abstraction levels of embedded computer platforms, from the operating system and networks, through the communications and quality-of-service (QoS) management middleware, up to the application level. The expected results include configurable, open platforms enabling inter-operability between processors, operating systems and programming languages, middleware for distributed embedded systems with real-time and quality-of-service requirements, as well as methodological guidelines and development tools.

1 Project objectives

The global aim of the project is to develop integral support elements for embedded, distributed real-time systems, including a family of inter-operable platforms, interconnection mechanisms, architecture and design methodologies, and application domains for this category of systems. The kind of integral support which is envisaged in the project deals with all system levels, from hardware, operating systems, and networks, through distribution middleware, to the application level.

The aims of the sub-projects are, in summary:

- **UPM**: distribution mechanisms for critical real-time systems; mechanisms for resource and QoS management; model-based development methods; new communication and service paradigms for embedded systems.

- **UC**: negotiation-based OS mechanisms and networks; communications middleware; schedulability analysis for new distributed platforms; component-based development environments.

- **UPV**: nanokernel for RTOS inter-operability; real-time storage systems; RT-Linux communications and middleware.

The work plan is organized into four work packages:

1. **Platforms.** The aim of this work package is to develop a family of inter-operable platforms for real-time embedded systems. The main development lines are:
• **Operating system mechanisms**: nanokernel for RTOS support (tasks 1.4–1.6); resource control (1.1); high-integrity RT kernels (1.2); embedded file systems (1.7); programming language issues (1.11).

• **Communications and middleware**: networks and communication protocols (tasks 1.8); Java-RMI extensions for high-integrity and QoS-oriented systems (1.3); configurable middleware for real-time systems (1.9); QoS-oriented middleware for soft real-time systems (1.10).

• **New generation platforms**: P4P platforms (task 1.12).

2. **Methods and tools**. The aim of this work package is to make progress in analysis methods and development methods and tools for embedded real-time systems. It covers the following lines:

• **Temporal analysis methods**: extension of analysis methods for distributed real-time systems (task 2.2) and hierarchically scheduled systems (2.3); QoS analysis methods for networks (2.4).

• **Execution environments**: Architectural paradigms (task 2.1); execution framework with QoS guarantee (2.5); open platform development environment (2.6); MDA for high-integrity systems (2.7).

3. **Applications**. This package is aimed at studying the application of the concepts, methods and tools developed in the previous work packages in order to make a preliminary assessment of their validity in the real world.

It includes demonstrations in the application areas of industrial control (task 3.1), aerospace and transport systems (3.2), mobile services and personal environments (3.3 and 3.4), remote assistance and e-learning (3.5 and 3.6).

4. **Project management and dissemination**. This work package includes project management activities and dissemination through publications, web sites, teaching, and others.

2 **Level of success**

2.1 **Platforms for real-time embedded systems**

2.1.1 **Operating systems mechanisms**

**Nano-kernel for RTOS support.** The main development in this area is XtratuM, a HAL (Hardware Abstraction Layer), or nano-kernel designed to meet real-time requirements. It enables the underlying hardware to be shared among several OS, one of which is a standard (non-realtime) Linux. It has been developed as a replacement for the original RTLinux nano-kernel mechanism, circumventing the patent problems related to it.

The latest public release of XtratuM (1.0) contains inter-domain memory protection by using the segmentation mechanism of the i386 architecture. MaRTE OS has been ported to XtratuM 1.0.

A new version of XtratuM (2.0) is under development. The objective of this new version is to provide full virtualisation for all the domains, making it possible to implement complex safety critical systems on the same computer platform as conventional applications running on general purpose OS, ensuring temporal and spatial isolation between them.

**RT-Linux redesign (partikle)** A complete re-implementation of RTLinux/GPL has been carried out, with the intent of solving both legal and internal design problems. The new RTOS, called Partikle, is compatible with the RTLinux API. Partikle is organised as a kernel, which is compiled to produce a kernel image, and a set of libraries and include files to be used by the applications.

Partikle services are provided via a single system call entry. It can be compiled to be executed on a bare machine, a XtratuM domain, or as a regular Linux process. A port of the kernel to the ARMv9 processor has also been done.

There are plans to replace RTLinux/GPL code with XtratuM/Partikle.
Resource control in real-time kernels. Mechanisms to protect shared resources have been designed and defined. In particular, a protected critical section mechanism has been designed, which ensures that the execution-time consumed in the critical sections does not exceed a specified budget. This mechanism has been implemented using the advanced timing control facilities recently added to MaRTE OS.

The execution-time monitoring mechanisms incorporated in the new Ada 2005 standard have been implemented in the current version of the ORK kernel (see below). These mechanisms can be used to ensure that a code segment does not consume processor time beyond a specified budget. Execution-time budgets can also be set to groups of Ada tasks, providing a useful mechanism for implementing execution-time servers and other hierarchical scheduling models.

High-integrity RT kernels. ORK (Open Ravenscar real-time Kernel) is a specialized kernel for the Ada Ravenscar tasking profile, supporting the implementation of high-integrity real-time systems in Ada. The original version of the kernel has been updated in order to support the full Ada 2005 standard, including new scheduling methods and other real-time features. The new version of the kernel provides a solid support for developing advanced real-time systems in Ada, with enhanced determinism and protection against potential timing errors.

Embedded file systems. Compact Flash and hard disk IDE drivers for the MaRTE OS have been implemented. A FAT16 file system has been implemented on top of these drivers. A new file system for Compact Flash memories, based on the TLSF (Two Level Segregated Fit) allocator, has also been designed.

The IDE driver for RTLinux has been re-designed and re-implemented to be portable and with DMA support. Currently, the temporal behaviour of the disks is being studied in order to be able to do scheduling analysis and request queue ordering.

Programming language issues. Several Java virtual machines have been evaluated with the idea to port one of them to MaRTE OS gaining the interoperability with the Java language. Finally, OVM was selected without conclusive results.

2.1.2 Communications and middleware

Networks and communication protocols. The communication protocol RT-EP running over MaRTE OS has been enhanced to provide it with a new generic interface that will be common to other protocols, mainly: we have added multicast and broadcast, synchronized send is available, we have implemented a distributed mutex mechanism with our own algorithm, sporadic server mechanisms has been implemented, and finally we have design a compatibility layer to POSIX. The protocol is being used to support the the new distributed contract mechanism developed in FRESCOR project. Based on RT-EP, AR-EP has been developed for high integrity systems. A driver of a wireless board for MaRTE OS has also been developed.

AFDX is a communication protocol designed for hard real-time communications in avionics systems. It was selected for testing RMI-HRT (see below), successfully used in an industrial experiment.

Java-RMI extensions. Two versions of RMI for distributed real-time systems have been developed: RMI-HRT and RMI-QoS. RMI-HRT, for hard real-time and safety-critical systems, is based on a high-integrity subset of RTSJ, and has been implemented on JamaicaVM using an AFDX network.

RMI-QoS, for QoS-oriented soft real-time systems, relies on assigning CPU and network budgets to communicating threads. It has been implemented on top of JamaicaVM and a TCP/IP stack. It has been ported to Linux/PC and Linux/iPAQ.

Configurable middleware for real-time systems. PolyORB (a middleware package that supports CORBA and the Ada Distributed Systems Annex) has been ported to MaRTE OS with RT-EP. The result continues being inter-operable with other ORB’s. The distributed transaction mechanism designed for RT-GLADE has been integrated into this platform, in order to provide support for the new flexible scheduling mechanisms available in the FRESCOR project.
QoS-oriented middleware for soft real-time systems. MaRTE OS and SHARK have been integrated in HOLA-QoS (a middleware for managing resources and QoS in a system with several running applications) in order to use their contract-based resource management techniques. The first prototypes have shown good initial performance.

2.1.3 New generation platforms

Work in this area has been centred on P4P platforms, particularly on applying the federated identity management approach from the Liberty standard on the m-learning domain. A circle of trust on a virtual campus has been used for prototyping purposes. The basic technologies used include GPRS, J2ME and JXTA. An extension to the Apache Tomcat including Liberty single sign-on in a transparent way has been developed. This work has been extended for full J2EE servers, including JBOSS, as well as the integration of Liberty identity providers with AAA servers such as FreeRadius.

2.2 Methods and tools

2.2.1 Temporal analysis methods

Analysis methods for distributed real-time systems. Theoretical developments on deadline assignment and the combination of deadline-based and priority-based scheduling in distributed systems have also been achieved. The new analysis techniques are currently been implemented in the MAST toolset in order to assess their validity for practical usage.

Work is being carried out on the analysis of RPCs in distributed systems, as an adaptation of offset-based analysis.

A response-time analysis method has been developed for applications that use RMI-HRT on AFDX networks. The design of RMI-HRT was carefully done in order to being able to analysis the response time of the threads involved in a communication. In addition, a model for calculating the delivery time of messages on an AFDX network has been proposed.

Analysis methods for hierarchically scheduled systems. A technique for response time analysis has been developed for systems with two-level hierarchical scheduling with a mix of EDF and priority-based schedulers. Work has also been done on the application of hierarchical scheduling methods to ensuring time isolation in partitioned systems with high-integrity requirements.

QoS analysis methods for networks. A model based on histogram and stochastic processes has been developed in order to analyse execution and transmission delays in networks from a statistical viewpoint. This model provides more accurate results than previous worst-case approaches, which often suffer from pessimism.

2.2.2 Execution environments

Architectural paradigms. Work on this topic has been postponed as it has been found interesting to base it on results from other tasks. For the moment, a taxonomy of embedded system architectures is under development. Additionally, an architectural framework for distributed critical systems, based on ORK and PolyORB, is being defined.

Open platform development environment. An extension for real-time of the standard D&C (Deployment and Configuration of Component-Based Distributed Application Specification) by the OMG has been developed at different levels: the description of the component interface, the description of the component implementation, the description of the mechanism to package the components, the platform description, and the description of the deployment. This extension provides a formalization of the available information for designing component based real-time applications. The proposed APIs constitute the base for the inter-operation of design and deployment tools for this kind of applications. This technology is also being used to develop new techniques based on CCM and Ada.
Model-driven technology for high-integrity systems  
Modelling language enhancements have been developed in order to enable safety aspects to be defined and validated. Work has been done on integrating general development methods based on UML, and safety verification languages based on FMECA (Failure Mode and Effect Analysis) and FTA (Fault Tree Analysis). The tools and artifacts developed are: an UML profile for representing safety concepts in software architectures, model transformations for evaluating models developed in Objecteering and UML2, as well as FTA and FMECA; definition of metrics for architectural evaluation, and tools for evaluating metrics applicable to Objecteering models.

2.3 Applications
Most of the tasks in the applications work package are planned to start in the second half of the project, after results from the previous work packages have been obtained. Therefore, only results in the following research area have been obtained up to now:

Mobile services and personal environments. The initial goals of this task have been adapted twice, in order to add information gathering activities about wireless sensor networks, and to include the analysis of using these networks into services oriented architectures. A survey of the state of the art in WSN has been produced, an initial set of basic technologies has been selected (tinyOS, mica2; IEEE802.15.4-ZigBee; uPnP and OSGi gateways), and a network architecture model has been produced.

The work on the definition of a reference architecture for distributed embedded systems, based on the usage of OSGi enabled network nodes, has been continued. The industrial-wide OSGi standard specification provides a services packaging model and a central registry which allows for dynamic deployment, update and retirement of services implementations; provides distributed management services, and a set of standardized basic services including Jini and uPnP support. The standard has been initially devised for its usage in home and car domains, but it is being used in mobile platforms, server platforms (including Spring and JBoss microcontainer architectures).

2.4 Management and dissemination
Management activities have been carried out at the project level by means of coordination meetings (held three times a year) and frequent e-mail and telephone exchanges. Internal management of the sub-projects includes routine organizational matters and local meetings.

Dissemination activities include publications (see next section), adding new content to undergraduate and graduate level courses, and a web site for the project including extensive information about real-time systems [www.dit.upm.es/str](http://www.dit.upm.es/str).

3 Result indicators

3.1 UPM results
The main results of sub-project 1 are:

- RMI-HRT, an extension of Java RMI for high-integrity real-time systems.
- RMI-QoS, an extension of Java-RMI for QoS-driven real-time systems.
- UML profile for representation of safety concepts in software architectures, and related tools and metrics for Objecteering.

3.1.1 Training
- PhD students: José Antonio Pulido, Santiago Urueña, Daniel Tejera, Juan Pedro Silva, Daniel Berjón, Javier López Briones, José Luis Ruiz Revuelta, Rubén Trapero, Ruth Tolosa (until June 2006), José Luis Arciniegas (until September 2006), Manuel Santillán (until September 2006), Hugo Parada, Félix Cuadrado, Bonifacion García.
3.1.2 Publications

The UPM team has produced a total of 22 publications (1–22 in the references section). See also www.dit.upm.es/str/papers/. 

3.1.3 Software products

- **ORK & GNATforLEON** has been adopted as the basis for the ASSERT virtual machine, a configurable platform for on-board aerospace systems. Thales-Alenia Space and EADS Astrium are using it in space-related pilot projects.

- **AR-TP with Ethernet driver** has been developed in Ada following the Ravenscar restrictions and is fully integrated with the PC version of ORK. This software together with PolyORB was used to build a preliminary prototype of the ASSERT virtual machine and it is currently used to test communication paradigms for high-integrity distributed systems with hard real-time requirements.

- **RMI-HRT** was the first mechanism for communication in high integrity distributed real-time systems in Java. It includes a novel approach to the response time analysis of systems based on RMI-HRT on top of AFDX networks.

- **RMI-QoS**, and also RMI-HRT have been used in the context of the HIJA project. The industrial partners stated the requirements for these type of mechanisms. Then, they were successfully integrated and tested on industrial applications.

3.1.4 Technology transfer

The ORK kernel is being used by Thales-Alenia Space and EADS Astrium in the framework of the ASSERT project (see below). Ada Core is distributing it as part of its GNAT Pro for LEON compilation system.

RMI-HRT and RMI-QOS have been used in the context of the HIJA project. The industrial partners stated the requirements for these type of mechanisms. Then, they were successfully integrated and tested on industrial applications.

Developments in the area of distributed real-time systems were used in the PROFIT project COM-PUGRID, in collaboration with ELIOP.

Work on wireless sensor networks has led to collaboration with Telvent and Telefónica I+D. The work on Liberty extensions has got two awards from Ericsson (EEM Patent Office). There is also an active collaboration with Telefónica I+D in this field through several international and national projects.

3.1.5 International projects

- **ASSERT** — Automated proof based System and Software Engineering for Real-Time (FP6-IST IP 004033).

- **Modelware** — MODELing solution for softWARE systems. (FP6-IST IP 511731).

- **HIJA** — High-Integrity Java (FP6-IST STREP 511718)

- **MORE** — Network-centric Middleware for Group communication and Resource Sharing across Heterogeneous Embedded Systems (FP6-IST STREP 032939).

- **ARTIST2** — Network of Excellence on Embedded Systems Design (FP6-IST NoE 004527).

- **PABIOS** (EUREKA-CELITIC).

- **OPUCE** (FP6-IST IP)

- **ITEA-COSI**
• ITEA-OSIRIS
• ITEA-PASSEPARTOUT
• ITEA-SERIOUS
• ITEA-MOSIS

3.1.6 Collaboration with other groups

Close cooperation has been kept with real-time research groups at the Universities of Zaragoza, Bilbao, Politécnica de Cartagena, Vigo and Rey Juan Carlos, as well as MCC and ESI.

In the framework of the ARTIS2 network, work has been done in collaboration with the real-time groups at York and Pisa Universities.

3.2 UC results

The main results of sub-project 2 can be summarized as follows:

• Control mechanisms for resources and a file system for MaRTE Os.
• Transaction mechanisms and management added to a RT-CORBA platform for Ada applications running on MaRTE OS with RT-EP.
• Schedulability analysis algorithms for hierarchical systems, and priority and deadlines assignment algorithms for distributed real-time systems with mixed scheduling.
• Real-time extension for the configuration and deployment specification of OMG component-based applications.

3.2.1 Training

• PhD students: Patricia López, Héctor Pérez, Sergio Martín, Miguel Tellería, Ángela del Barrio, Laura Barros, César Cuevas, Álvaro Cuesta
• Doctoral dissertations: Patricia López (planned before the end of the project)
• Undergraduate students: 9 graduation projects.

3.2.2 Publications

The UC team has produced a total of 19 publications (23–41 in the references section). See also http://www.ctr.unican.es.

3.2.3 Software products

• MaRTE OS: Operating system (http://marte.unican.es)
• A version of PolyORB with enhanced real-time capabilities.
• MAST: Modelling and analysis toolset (http://mast.unican.es)
• Plug-in for ECLIPSE to manage real-time components specification.
3.2.4 Technology transfer

- Collaboration with the *Centro Tecnolóxico de Componentes de Cantabria* and MATESCO (UC Department) for technical assessment on applying MaRTE OS to the control of an orientable antenna and the calculation of GPS-based orientation.

- Collaboration with *Ada Core* on integrating MaRTE OS as a basic platform of the GNAT compiler.

- Collaboration with the Commissariat à l’Énergie Atomique (CEA, Paris) on the elaboration of the OMG standard MARTE.

- Collaboration with ENSA (*Equipos Nucleares S.A.*) on robotic applications development.

- Participation in HESPERIA project (CENIT program granted by CDTI) [www.proyecto-hesperia.org](http://www.proyecto-hesperia.org).

3.2.5 International projects


3.2.6 Collaboration with other groups

- University of Mälardalen (Sweden), collaboration with PROGRESS project members [http://www.mrtc.mdh.se/progress/](http://www.mrtc.mdh.se/progress/).

- University of Aveiro (Portugal), to integrate their communications protocol FTT-Ethernet with the platform of this project.

- University of Zaragoza, RoPERT group (Robotics, Perception and Real-Time), to develop a real-time protocol over wireless networks.

3.3 UPV results

The main results from sub-project 3 are:


- Partikle: A new RTOS, intended to replace the original RTLinux to overcome its limitations.

- A stochastic model for studying the network transmission time of video streams.

- Study of rotational hard disk timing parameters in order to guarantee real-time performance.

3.3.1 Training

- PhD students: Vicente Brocal, Salvador Peiró.

- Doctoral dissertations: Miguel Masmano, Arnoldo Díaz.

- Post-doctoral students: Miguel Masmano, Audrey Marchand.

3.3.2 Publications

UPV has produced a total of 8 publications, which are listed as numbers 42–49 in the references section.
3.3.3 Software products

- XtratuM 1.0: [www.xtratum.org](http://www.xtratum.org)
- Partikle 1.0: [www.e-rtl.org](http://www.e-rtl.org)
- TLSF 2.2: [rtportal.upv.es/rtmalloc](http://rtportal.upv.es/rtmalloc)

3.3.4 Technology transfer

- Most of the software products developed has been included in the standard RTLinux/GPL distribution. The computer that hosts the rtlinux/gpl site [www.rtlinux-gpl.org](http://www.rtlinux-gpl.org) is run by the UPV group.
- Contacts with MoviRobotics [www.m-robotics.com](http://www.m-robotics.com) which are interested in using Partikle on their products.

3.3.5 International projects


3.3.6 Collaboration with other groups

1. Active collaboration with OpenTech [www.hofr.at/](http://www.hofr.at/) to jointly develop RTLinux/GPL.

References


