IST-214373 ArtistDesign
Network of Excellence
on Design for Embedded Systems

Activity Progress Report for Year 3

Resource-aware Operating Systems

Cluster:

Operating Systems and Networks

Activity Leader:

Prof. Giorgio Buttazzo (Scuola Superiore S. Anna)
http://retis.sssup.it/~giorgio/

Policy Objective (abstract)

The objective of this activity is to show how current operating systems can be designed to support emerging real-time applications that exhibit a high degree of complexity and operate in highly dynamic environments where resource demands can change unpredictably. Adaptive resource management mechanisms will be investigated both in uniprocessor and multicore architectures. The impact on operating system standards (like RT-POSIX and OSEK) will also be taken into account.

Versions

number	comment	date
1.0	First version delivered to the reviewers	February 4th 2010

Table of Contents

1. Overview of the Activity

1.1 ArtistDesign participants and their role within the Activity

Cluster Leader: Giorgio Buttazzo – Scuola Superiore S. Anna (Italy)
Role: Activity coordinator, kernel maintenance, development of robotic applications.

Team Leader: Luis Almeida – University of Aveiro (Portugal)
Role: networking platform, development of distributed applications.

Team Leader: Gerhard Fohler – University of Kaiserslauten (Germany)
Role: video streaming applications, scheduling.

Team Leader: Michael Gonzalez Harbour – University of Cantabria (Spain)
Role: definition of the POSIX operating system interface.

Team Leader: Alan Burns – University of York (UK)
Role: feasibility analysis of fixed priority real-time systems.

Team Leader: Eduardo Tovar – Polytechnic Institute of Porto (Portugal)
Role: distributed applications and QoS over heterogeneous networks.

-- Changes wrt Y2 deliverable --

No changes with respect to Year 2.

1.2 Affiliated participants and their role within the Activity

Team Leader: Tullio Facchinetti – University of Pavia (Italy)
Role: embedded real-time systems and robotics applications.

Team Leader: Paolo Gai – Evidence s.r.l. (Italy)
Role: real-time kernels and operating systems standards.

Team Leader: Hermann Haertig – Universtity of Dresden (Germany)
Role: microkernel architectures and virtualization techniques

Team Leader: Liesbeth Steffens – NXP (The Netherlands)
Role: real-time kernels for media processing applications.

Team Leader: Pau Marti – Universitat Politècnica de Catalunya (Italy)
Role: control applications and schedulability of event-driven control systems.

Team Leader: Alejandro Alonso – Technical University of Madrid (Spain)
Role: QoS resource management, high integrity systems

Team Leader: Marisol García Valls – University Carlos III of Madrid (Spain)
Role: QoS-based resource management, real-time support in middleware and distributed systems.

Team Leader: Alfons Crespo –Technical University of Valencia (Spain)
Role: real-time memory management, virtualization of real-time kernels

Team Leader: Salvatore Scafidi – Windriver (Italy)
Role: real-time operating systems

Team Leader: Stylianos Mamagkakis – IMEC Leuven (Switzerland)

Role: Energy-aware scheduling and memory management for dynamic applications

Team Leader: Antonio Bersani – Microchip Technology (Italy)
Role: Hardware components and I/O devices for real-time applications

-- Changes wrt Y2 deliverable --

The role of University Carlos III of Madrid has been updated.

1.3 Starting Date, and Expected Ending Date

Starting date: January 1st, 2008

Ending date. December 31st, 2011.

-- Changes wrt Y2 deliverable --

The ending date has been extended by one year.

1.4 Policy Objective

The main objective of this activity is to investigate how current real-time operating systems have to be extended or modified to support emerging real-time embedded systems characterized by a high degree of complexity, highly variable resource requirements and parallel processing such as multicores. Most embedded systems are often characterized by scarce resources, in terms of processing power, memory, space, weight, energy, and cost. Hence, another objective is to investigate kernel mechanisms that can efficiently manage the available resources, taking multiple constraints into account, whilst guaranteeing isolation properties. Also, to support dynamic applications with variable resource requirements or to cope with unpredictable resource availability, feedback control techniques for resource management at the operating system and application level will be investigated. The impact on operating system standards (like RT-POSIX and OSEK) will also be taken into account. In fact, developing real-time applications and components using an interface compliant to a standard will promote portability to other compliant platforms and will challenge the current standard to be extended to better meet the needs of advanced applications with flexible resource requirements. We realize though that in specific application domains, significant performance advantages can be realized by optimizing software across layers, for example exploiting specific behaviour of a medium access control protocol. This is often the case in operating systems for sensor network platforms such as TinyOS or NanoRK. Such cross-layer design does not necessary contradicts operating system standards, but they do require other interfaces.

-- Changes wrt Y2 deliverable --

No changes with respect to Year 2.

1.5 Background

Although there is a large variety of real-time operating systems (RTOSs) varying in sizes, level of provided services, and efficiency, there are some common elements that can be found in most of them:

- An RTOS usually provides support for concurrent programming via processes or threads or both. Processes usually provide protection through separate address spaces, while threads can cooperate more easily by sharing the same address space, but with no protection.
- Real-time scheduling services are provided because this is one of the keys to obtaining a predictable timing behaviour. Most current RTOS's provide the notion of a scheduling priority, usually fixed, as for the moment there are few systems providing deadline-driven or other dynamic-priority scheduling.
- Although some RTOS designed for high-integrity applications use non preemptive scheduling, most support preemption because it leads to smaller latencies and a higher degree of utilization of the resources.
- The OS has to support predictable synchronization mechanisms, both for events or signal and wait services, as well as for mutual exclusion. In the later case some way of preventing priority inversion is required because otherwise very improbable but also very long delays may occur.
- The OS has to provide time management services with sufficient precision and resolution to make it possible for the application to meet its timing requirements.
- OS behaviour should be predictable, and so metrics of the response time bounds of the services that are used in real-time loops should be clearly given by the RTOS manufacturer or obtained by the application developer. These metrics include the interrupt latency (i.e., time from interrupt to task run), the worst case execution time of the system calls used in real-time loops, and the maximum time during which interrupts are masked or disabled by the OS and by any driver.

An RTOS is generally chosen not only for its real-time characteristics, but also for the middleware that is integrated in the RTOS, such as file system, communication stack, for its portability to different platforms (i.e., the board support packages that are provided), and for the associated cross-development environment.
A commercial RTOS is usually marketed as the run-time component of an embedded development platform, which also includes a comprehensive suite of (cross-) development tools and utilities and a range of communications options for the target connection to the host, in an Integrated Development Environment (IDE). Moreover, the vendor generally provides development support. For each successful open source RTOS there is also at least one commercial distributor that provides development tools and development support. For many embedded-systems companies, the availability of development tools and support is a major requirement for choosing a particular RTOS. The quality of the overall package deal, including service and pricing strategy is often decisive in choosing a particular RTOS.

-- Changes wrt Y2 deliverable --

No changes with respect to Year 2.

1.6 Technical Description: Joint Research

Multicore embedded real-time systems
Next generation RTOS must allow optimal off-line partitioning of the application source code on the different CPUs available on multicore heterogeneous systems, as well as on-line strategies for the run-time migration with the objective of guaranteeing optimal usage of the CPUs available, with real-time response as well as minimization of power consumption. The Cluster investigated a method for partitioning parallel applications on top of multicore platforms, optimizing bandwidth usage and tacking timing and precedence constratins into account.

Component-based operating systems
To optimize the use of resources and increase software portability on different platforms, it is highly desirable to compose the operating system using the functions strictly necessary for the application. To achieve this goal, it is crucial to design the operating system to be modular, so that each component can be independently developed from the others and can be replaced without changing the application. The Cluster was involved in the development of modular real-time systems and hierarchical scheduling techniques that allow isolating the temporal behavior of real-time components.

Deadline Scheduling on Linux
Kernel support for deadline-based scheduling algorithms has been developed in Linux. The contribution to the cluster has been twofold: On one hand, a new POSIX-like scheduling policy, called SCHED_DEADLINE, was implemented as a novel scheduling class, like the POSIX-specified ones already present in the kernel; on the other hand, the existing bandwidth control mechanism (called throttling), was extended using a Constant Bandwidth Server.

Real-Time techniques for Service oriented Architectures (SoA)
Resource provisioning in SoA is a challenge, especially when the system consists of hundreds of nodes and real-time constraints are required. We extended a technique called “Advance Reservation” that is used in GRID architectures to cope with QoS requirements and dynamic admission control which is required by very dynamic SoA. The Cluster adopted EDF and resource reservations as a base scheduler. On top of it, two admission control methods have been implemented: a deterministic admission control, which is safe but can waste resources, and a probabilistic admission control, which greatly improves on resource utilization.

-- Changes wrt Y2 deliverable --

More effort has been put to include deadline scheduling in the Linux kernel, extend resource reservations to multiprocessor systems, build an abstraction layer for multicore platforms and investigate hierarchical scheduling for component-based design.

1.7 Work achieved in Year 1 *(Jan-Dec 2008)*

- **Pisa**: Partitioning a real-time application in a multi-core architecture, with the objective of providing the resource reservation abstraction to achieve temporal protection and to allocate a fraction of the available resources to a given application. Unfortunately, extending the resource reservation paradigm to multicore architectures is not trivial, since resource allocation must be considered together with the problems of exploiting the intrinsic parallelism of certain applications.

- **Pisa**: Another issue we started to investigate in this first year is the evaluation of the interactions of cache memories with scheduling. In particular, preemptive algorithms tend to destroy the cache content of the preempted activity, so increasing the number of cache misses, causing an increase of task computation times. To better evaluate such dependencies we decided to perform a set of simulation experiments aimed at measuring such effects in different scenarios and working conditions.
- **Pavia**: We started investigating the possibility of executing different operating systems on a single multicore platform, by assigning a different core to each operating system, as a special kind of virtualization technique that does not rely on lower-layer components. The short-term goal is to run a real-time operating system along with a general-purpose one in a completely transparent fashion, while the subsequent research will address the cross effects between the operating systems due to shared resources, like memory, bus, peripherals and cache.
- **Pavia**: Specific work on embedded and robotics applications has been started. In this domain the research directions focuses on the development of applications based on small micro-controllers to assess the benefits of real-time computing for the predictability and the overall performance and the application.
- **Catalonia**: We also started to investigate the schedulability analysis of a specific type of control applications named event-driven control applications. The importance of these type of controllers is that they can provide the same control performance than standard periodic controllers while minimizing resource utilization.

-- No changes wrt Y2 deliverable --

This section was already presented in the Y1 deliverable, in sections 1.7 and 3.1.

1.8 Work achieved in Year 2 (Jan-Dec 2009)

Resource reservations on multi-core platforms
Extending the resource reservation framework from uniprocessor to multicore systems is not trivial and requires new theoretical bases. In this year, the problem of partitioning parallel real-time applications on top of multicore platforms has been investigated for exploiting the available parallelism offered by modern architectures.

Deadline Scheduling on Linux
Extending the Linux kernel to support novel real-time scheduling algorithms is essential to expose the most relevant research results achieved within small research communities to the entire world of software developers. A great effort has been done in this Cluster to introduce the Earliest Deadline First (EDF) scheduler as a scheduling class in Linux, to allow a more efficient support of real-time applications and enable the implementation of advanced resource reservations techniques.

Power-Aware Library
The Scuola Superiore Sant'Anna of Pisa started a work to implement a kernel library to manage power and temperature constraints in real-time embedded systems.

ERIKA on MPARM
The Scuola Superiore Sant'Anna of Pisa ported the Erika real-time kernel on the MPARM simulator developed by the University of Bologna.

-- No changes wrt Y2 deliverable --

This section was already presented in the Y2 deliverable, in sections 1.8 and 3.1.

1.9 Problems Tackled in Year 3 (Jan-Dec 2010)

Resource reservations on multi-core platforms
The work on multicore resource reservation has been extended by providing a methodology that can partition a parallel application on a set of uniprocessor reservations with minimal bandwidth requirements.

Deadline Scheduling on Linux
A great effort has been done in this Cluster to introduce resource reservation and deadline-based scheduling (EDF) in the Linux operating system. The implementation is now complete and allows a more efficient support of real-time applications, enabling the implementation of advanced resource reservations techniques with adaptive bandwidth management.

Power-Aware Resource Management
The work started by the Scuola Superiore Sant'Anna of Pisa on providing kernel support for managing power and temperature constraints has been completed. A C library for the Erika kernel has been developed to manage real-time applications with power and temperature requirements. A uniform approach has been adopted to handle different types of resources, such as processors, memories, buses, wireless transceivers, and I/O devices.

Architecture simulator for real-time applications
The multi-processor cycle-accurate architectural simulator (MPARM) developed at the University of Bologna has been extended to run real-time concurrent applications on the Erika kernel developed by Evidence and the Scuola Superiore Sant'Anna of Pisa. The simulator has been used to perform a set of experiments of non-preempitive scheduling, cache-aware scheduling, and power-aware strategies.

-- The above is new material, not present in the Y2 deliverable --

2. Summary of Activity Progress in Year 3 *(Jan-Dec 2010)*

2.1 Technical Achievements

Partitioning real-time applications on multi-core platforms (Pisa)
The RETIS group of Pisa investigated a method for partitioning parallel applications on top of multicore platforms, optimizing bandwidth usage and taking timing and precedence constraints into account. The method generalizes the resource reservation abstraction to multicore systems and allows allocating a fraction of the available resources to a given application. The tool is available from: http://mrr.sssup.it/mrr/

Elastic Scheduling on the Erika kernel (Pisa)
The Erika kernel has been enhanced with the implementation of the elastic scheduling algorithm, which allows handling overload conditions on the task set by changing the task periods. The scheduler has been used in conjunction with a TDMA bus arbiter to allocate higher bandwidth to tasks that need to run with higher activation rates.

Power management library on the Erika kernel (Pisa)
A C library for the Erika kernel has been developed to manage real-time applications with power and temperature requirements. A uniform approach has been adopted to handle different types of resources, such as processors, memories, buses, wireless transceivers, and I/O devices.

Running Erika on the MPARM simulator (Pisa)
The porting of the Erika kernel on the MPARM emulator has been completed. Several real-time applications have been executed on the ARM7 architecture to test the performance of novel energy-aware scheduling algorithms.
ERIKA: http://erika.sssup.it/
MPARM: http://www-micrel.deis.unibo.it/sitonew/research/mparm.html

Remote programming of wireless nodes (Pisa)
A software support has been developed on the Erika kernel to allow remote programming and management of wireless nodes through the ZigBee standard. With this tool, it is possible to monitor and reprogram the nodes of a wireless network through a remote base station PC.

Hybrid FP/EDF scheduler on Linux (Pisa)
The hybrid FP/EDF scheduler developed in the context of the IRMOS European Project (http://www.irmosproject.eu/) is being maintained by Pisa. In particular, the scheduler has been ported to more recent kernel releases (2.6.35) and to the well-known PREEMPT_RT branch of the kernel. The same scheduler has been applied to the problem of temporal isolation of Virtual Machines co-scheduled on the same CPUs.

Execution of multiple real-time operating systems on a multicore architecture (Pavia)
This research activity focused on the development of a software methodology for executing more than one operating system (OS) on a multicore platform. The goal is to allocate one or more cores to one OS and establish a communication policy among different OSes. In particular, we studied the technical issues related to running the Linux operating system side by side with the S.Ha.R.K. Real-Time kernel on a multicore processor. It is possible to launch a S.Ha.R.K. application on a given core while running Linux on the main core.

Deadline Scheduling on Linux (Pisa, Evidence)
The Linux scheduling framework (kernel release 2.6.33) has been extended to include a resource reservation scheduler based on Earliest Deadline First (EDF) scheduling. A new scheduling class SCHED_DEADLINE has been implemented and proposed to the kernel community. The class adds the possibility of scheduling tasks using the Constant Bandwidth Server (CBS) under the EDF algorithm, without changing the behavior of tasks scheduled using the existing policies. Extensive experiments show that tasks executed under SCHED DEADLINE instead of SCHED FIFO can better meet their timing requirements. For example, when SCHED FIFO is used to schedule two instances of a GTK video player, only one instance of the player can meet its timing constraints (i.e., 25 frames per second). When SCHED DEADLINE is used, instead, both instances can reproduce the movie smoothly, and the performance of each instance is proportional to the share of CPU assigned through the real-time parameters.

A general description of EDF on Linux: http://retis.sssup.it/?q=node/35

SCHED_DEADLINE: http://www.evidence.eu.com/sched_deadline.html

The throttling reimplementation: http://feanor.sssup.it/~fabio/linux/edf-throttling/

OS structure for large heterogeneous NoC platforms (York)
Preliminary examinations of appropriate OS structures for large heterogeneous NoC platforms. Initial work examines I/O and data/code memory transfers and bottlenecks. Currently we are modelling typical memory transfers in order to derive appropriate policies. Also we are examining the restructuring of Unix style OSs to be more appropriate for NoCs.

Virtualisation for multicore architectures (York)
The University of York, in collaboration with LIRMM (France) worked on virtualisation and binary translation for multicore architectures. A first prototype in FPGA of a multicore architecture that has a core dedicated to binary translation was produced. A high level model of that architecture within lsi.noc framework is being produced to evaluate the potential benefit of dynamic code retargetting and migration for specific applications.

Execution time monitoring and interrupt handlers (Cantabria)
University of Cantabria has continued to work on the monitoring of the execution times of interrupt handlers, which is important to increase the accuracy of the mechanisms used to monitor the execution times of all tasks and detect situations in which the estimated worst-case execution time (WCET) is exceeded. The accuracy of the POSIX execution time clocks and timers can be enhanced significantly if they are not arbitrarily charged with the execution times of interrupt service handlers, as occurs in simple implementations. We have implemented support for measuring the execution time of interrupt handlers separately from the execution time of user threads. We have measured the overheads incurred by the implementation of the new services, and found that they are very low in comparison with the execution times of common interrupt handlers.

Task assignment for heterogeneous multiprocessors (Porto)
Multiprocessors implemented on a single chip (a.k.a multicores) are becoming ubiquitous. More and more designers realize the advantages (in terms of computational speed or power "consumption") in using multicore processors comprising different functional units or processor units specialized to operate at very high speed for the type of computation it is designed for. For example a graphics processor unit can perform graphics operations at a much higher speed than a general purpose processor. From the perspective of real-time scheduling, this implies that the execution time of a task depends on which processor core it is assigned to.

Previously known algorithms have high computational complexity but one can observe that many heterogeneous multiprocessors used in practice have only two types of processors.
Researchers at CISTER research unit at ISEP/IPP have therefore considered the special case of a heterogeneous multiprocessor with two types of processor and under this assumption they have developed a new algorithm for assigning real-time tasks to processors. What makes this new algorithm remarkable is that it offers a so-called resource augmentation bound similar to the prior state of art but in addition to that, in experimental performance evaluations, it is found to be able to succeed to assign more task sets than prior state of art and it runs more than 10000 times faster.
URL: http://www.cister.isep.ipp.pt/docs/

Provably Good Resource Sharing for Multiprocessors (Porto)
The number of processor cores in multicore processor on a single chip is increasing exponentially and this brings a tremendous amount of raw computing capacity available to designers. Many applications however are composed of tasks which share data or other resources and this can limit the parallelism in the application and consequently making it impossible to harness the full power of a multicore processor. The research literature makes available several protocols for allowing real-time tasks executing on a multiprocessor to share resources but unfortunately, it has been difficult to know how well they perform and in particular, it has been difficult to know if there is any task set for which existing resource sharing protocols perform very poorly (for example performing like a uniprocessor where it would be possible to utilize the entire parallism in the computing platform).
Researchers at CISTER research unit at ISEP/IPP have addressed this problem. Specifically, Researchers at CISTER research unit have developed a new resource sharing protocol for multiprocessors and a scheduling algorithm to be used in conjunction with this protocol. This protocol has *provably*good*performance* meaning that if it is possible for a set of tasks to meet deadlines then tasks meet deadlines as well if this resource sharing protocol is used (and its corresponding processor scheduling algorithm) assuming that the new protocol is given processors which are speeded up by a constant factor.
URL: http://www.cister.isep.ipp.pt/docs/

Least-Laxity-First Schedulability Analysis on Multiprocessor Platforms (Porto)
A "natural" approach for scheduling real-time tasks might be to select the task with the earliest deadline and then if the remaining execution time of a task is equal to its remaining time to its deadline then the task is given the highest priority. Another "natural" approach would be to, for each task, compute the remaining deadline minus the remaining execution time. The former is called Earliest-Deadline-First-with-Zero-Laxity (EDZL) and the latter is called Least-Laxity-First (LLF). Both of them utilize the so-called laxity (the remaining deadline minus the remaining execution time) and they are both optimal for uniprocessor scheduling. It is known that they are non-optimal for multiprocessor scheduling but they are often perceived to be "good" algorithms for multiprocessors. CISTER researchers have explored these algorithms and produced new schedulability tests for the use of these algorithms on multiprocessors.
URL: http://www.cister.isep.ipp.pt/docs/

Capacity Sharing and Stealing Server-based Scheduling (Porto)
As an increasing number of users run both real-time and traditional desktop applications in the same system the issue of how to provide an efficient resource utilisation in this highly dynamic, open, and shared environment becomes very important. Based upon a careful study of the ways in which unused reserved capacities can be more efficiently used to meet deadlines of tasks whose resource usage exceeds their reservations, CISTER researchers proposed the coexistence of the traditional isolated servers with a novel non-isolated type of servers, combining an efficient reclamation of residual capacities with a controlled isolation loss. The work undertaken unified the sharing and stealing strategy with reservations exchange within

bandwidth inheritance, in order to mitigate the cost of blocking on soft real-time tasks whose actual execution behaviour is only known by executing tasks until completion. While preserving the isolation principles of independent tasks, upon blocking, a task is allowed to be executed on more than its dedicated server, efficiently exchanging reserved capacities among servers to reduce the undesirable effects caused by inter-task blocking.
Considering the performed work, hard schedulability guarantees can be provided either for independent and inter-dependent task sets, even when hard and soft real-time tasks do share resources and exhibit precedence constraints in open real-time systems.
URL: http://www.cister.isep.ipp.pt/docs/

Platform for QoS Applications in Mobile Phones (Porto)
Since its public release, the Android operating System has captured the interest from companies, developers and the general audience. From that time up to now, this software platform has been constantly improved either in terms of features or supported hardware and, at the same time, extended to new types of devices different from the originally intended mobile ones.
However, there is a feature that has not been explored yet - its real-time capabilities. This work analyzed the suitability of Android in order to be used in Open Real-Time environments. By analysing the software platform, with the main focus on the virtual machine and its underlying operating system environments, we are able to point out Android's current limitations and, therefore, propose different directions in order to make Android suitable for these environments.
URL: http://www.cister.isep.ipp.pt/docs/

Temporal behavior of the protocol stack and impact of the OS (UnivPorto)
Experimental study and characterization of the impact of the operating system (different flavours of Linux and QNX Neutrino) and protocol stack (IP plus device-driver) in the timeliness of Ethernet communications [SASM10]. Among other phenomena, it was observed that the time a packet spends inside the protocol stack can reach one order of magnitude higher than the time it takes to be transmitted on the wire. Work mainly carried out within the iLAND (ARTEMIS) project.

Education of embedded control system engineers. (UPC, SSSUP)
The traditional teaching approach to real-time system and to control system courses can be quite math intensive and abstract, thus failing to introduce students to the realities of embedded control system implementation. Moreover, the natural interaction and integration between these two disciplines is often neglected due to the traditional compartmentalized nature of science and engineering education. To overcome such limitations, a laboratory experiment to be integrated in the education of embedded control system engineers that flexibly combines these tow main disciplines has been developed [MVFCB10]. The experiment is a real-time control of a dynamical system in the form of an electronic circuit, supported by the Erika real-time kernel. The flexibility is achieved by describing a set of problems/observations that provide the spectrum of possible choices that instructors/students have and the work that has to be done to complete the experiment. This permits elaborating diverse assignments for the laboratory experiment with open problems, rather than providing tight guidelines, while providing the tools for assessing whether the students' design and implementation choices were correct. Finally, through the experiment, it is shown that the combination of both disciplines does not raise conflicts. It rather provides complementary approaches/views that help in the multidisciplinary learning process required in the embedded system education.

Implementation of Self-Triggered Networked Control Systems (UPC)
A self-triggered controller is characterized, in general, by a non-periodic sequence of job activations. And each job execution, apart from performing sampling, control algorithm computation and actuation, calculates the next job activation time as a function of the plant state. An implementation of self-triggered controllers in networked control systems has been performed [CMVLVFG10]. In particular, the processing platform is composed by computing nodes enabled with the Erika real-time kernel that exchange control data over the Controller Area Network. The implementation corroborates that self-triggered control can be used for minimizing bandwidth utilization while providing similar control performance than periodic controllers.

The SimpleRTK kernel (UPC)
The SimpleRTK, a minimal multi-tasking preemptive real-time kernel based on Microchip dsPIC33FJ256GP710 has been developed [GVM10]. It has been designed pursuing two goals. First, it aims at becoming a valuable software tool in practical assignments in undergraduate and graduate courses on networked and embedded control systems. Hence, it includes only basic services, avoiding excessive memory/time overhead and unnecessary complexity. Second, it offers simple but effective support to small control applications characterized by timing properties different than those exhibited by periodic controllers. In particular, it suits applications based on feedback scheduling approaches or on event-driven control methods, where tasks executions are non-periodic in the general case.

-- The above is new material, not present in the Y2 deliverable --

2.2 Individual Publications Resulting from these Achievements

Pisa

1. Tommaso Cucinotta, Luigi Palopoli, "QoS Control for Pipelines of Tasks Using Multiple Resources," IEEE Transactions on Computers, Vol. 53, No. 3, pp. 416--430, March 2010.
2. T. Cucinotta, D. Giani, D. Faggioli and F. Checconi, “Providing Performance Guarantees to Virtual Machines using Real-Time Scheduling,” in Proceedings of the 5th Workshop on Virtualization and High-Performance Cloud Computing (VHPC 2010), Ischia (Naples), Italy, August 2010.
3. T. Cucinotta, D. Faggioli, “An Exception Based Approach to Timing Constraints Violations in Real-Time and Multimedia Applications,” in Proceedings of the IEEE Symposium on Industrial Embedded Systems (SIES 2010), Trento, Italy, July 2010.
4. D. Faggioli, G. Lipari, T. Cucinotta, “The Multiprocessor BandWidth Inheritance Protocol,” in Proceedings of the 22nd Euromicro Conference on Real-Time Systems (ECRTS 2010), Bruxelles, Belgium, July 2010.
5. T. Cucinotta, F. Checconi, L. Abeni, L. Palopoli, “Self-tuning Schedulers for Legacy Real-Time Applications,” in Proceedings of the 5th ACM European Conference on Computer Systems (EuroSys 2010), Paris, France, April 2010.
6. N. Serreli, G. Lipari, E. Bini “The Demand Bound Function Interface of Distributed Sporadic Pipelines of Tasks Scheduled by EDF”, in Proceedings of ECRTS, Bruxelles, pp. 187-196, July 3-5, 2010.

7. N. Serreli, G. Lipari, E. Bini, “The Distributed Deadline Synchronization Protocol for Real-Time Systems Scheduled by EDF”, Proceedings of ETFA, Bilbao, 13-15 Sept. 2010.
8. G. F. Anastasi, E. Bini, A. Romano, G. Lipari, “A Real-Time Reconfiguration Infrastructure for Distributed Embedded Control Systems”, Proceedings of ETFA, Bilbao, 13-15 Sept. 2010.

York

9. T. Richardson, A. Wellings, “An Admission Control Protocol for Real-Time OSGi”, The 13th IEEE International Symposium on Object/component/service-oriented Real-time distributed computing, 2010.
10. Yang Chang, Andy Wellings, “Garbage Collection for Flexible Hard Real-Time Systems”, IEEE Transactions on Computers, Vol 59, pp1063-1075, 2010.
11. Osmar Marchi Dos Santos, Andy Wellings, “Measuring and policing blocking times in real-time systems”, ACM Trans. Embed. Comput. Syst., Vol 10, pp1-29, 2010.
12. Ian Gray, Neil Audsley, “Supporting Islands of Coherency for highly-parallel embedded architectures using Compile-Time Virtualisation”, 13th International Workshop on Software and Compilers for Embedded Systems (SCOPES), 2010.
13. Richardson, T., Wellings, A. J., Dianes, J. A., Diaz, M., “Towards memory management for service-oriented real-time systems” JTRES '10: Proceedings of the 8th International Workshop on Java Technologies for Real-Time and Embedded Systems, pp128-137, 2010.
14. Jack Whitham, Neil Audsley, “Investigating average versus worst-case timing behavior of data caches and data scratchpads”, Proc. ECRTS, pp165-174, 2010.
15. Jack Whitham, Neil Audsley, “Studying the Applicability of the Scratchpad Memory Management Unit” Proc. RTAS, pp205-214, 2010.
16. Jack Whitham, Neil Audsley, “Time-Predictable Out-of-Order Execution for Hard Real-Time Systems”, IEEE Transactions on Computers, Vol 59, pp1210-1223, 2010.

Cantabria

17. Mario Aldea Rivas and Michael González Harbour. "Execution time monitoring and interrupt handlers". 14th International Real-Time Ada Workshop, Porto Venere (Italy), ACM SIGAda Ada Letters, Volume XXX, Number 1, pp. 91-103, pp 68-73, April 2010.

UC3M (affiliated Cantabria)

18. Marisol García-Valls, Iria Estévez-Ayres, Pablo Basanta-Val. Dynamic priority assignment scheme for contract-based QoS resource Management . 7th IEEE International Conference on Embedded Software and Systems. (IEEE ICESS 2010). IEEE Computer Society Press. Bradford, UK. May 29 - June 1, 2010.
19. P. Basanta-Val, M. García-Valls and I. Estévez-Ayres. Extending the Concurrency Model of the Real-Time Specification for Java. Accepted (2010, July) for Concurrency and Computation: Practice and Experience.
20. I. Estévez-Ayres, M.García-Valls and P. Basanta-Val. On the reconfiguration of service-based real-time applications. In Proc. Of WiP session of IEEE RTAS-2010, Stockholm, Sweden, 12-15 April, 2010.

21. J. Díez-Sánchez (advisor: I. Estévez-Ayres). Implementation and evaluation of composition algorithms for distributed real-time service-based applications. Master Thesis, Universidad Carlos III de Madrid. March 2010

22. I. Fernandez-Pacheco (advisor I. Estévez-Ayres). Design and development of a video-surveillance system based on Android technology. Master Thesis. Universidad Carlos III de Madrid. June 2010.

Porto

23. [LESL10] Jinkyu Lee, Arvind Easwaran, Insik Shin, Insup Lee, "Multiprocessor Real-Time Scheduling Considering Concurrency and Urgency", In SIGBED Review, Special Issue on the Work-in-Progress (WIP) Session at the 2009 IEEE Real-Time Systems Symposium (RTSS), Volume 7, Number 1, January 2010.

24. [AEL10] Björn Andersson, Arvind Easwaran, Jinkyu Lee,"Finding an Upper Bound on the Increase in Execution Time Due to Contention on the Memory Bus in COTS-Based Multicore Systems", in SIGBED Review, Special Issue on the Work-in-Progress (WIP) Session at the 2009 IEEE Real-Time Systems Symposium (RTSS), Volume 7, Number 1, January 2010.

25. [NP10] Luis Miguel Nogueira, Luis Miguel Pinho, "A Capacity Sharing and Stealing Strategy for Open Real-time Systems", in Journal of Systems Architecture, Volume 56, Issues 4-6, April-June 2010, Pages 163-179

26. [AE10] Björn Andersson, Arvind Easwaran, "Provably Good Multiprocessor Scheduling with Resource Sharing", in the Real-Time Systems Journal, Volume 46, Issue 2, 2010, Pages 153-156.

27. [ARB10] Björn Andersson, Gurulingesh Raravi, Konstantinos Bletsas, "Assigning Real-Time Tasks on Heterogeneous Multiprocessors with Two Unrelated Types of Processors," in Proceedings of the 31st IEEE Real-Time Systems Symposium November 30 - December 3, 2010, San Diego, CA, USA

28. [APTP10] Björn Andersson, Nuno Pereira, Eduardo Tovar, Filipe Pacheco, "Using a Prioritized MAC Protocol to Execute the Database Operation Join in Networked Embedded Computer Systems", Published in proceedings 8th IEEE Workshop on Intelligent Solutions in Embedded Systems (WISES 2010), Heraklion, Crete, Greece, 8-9 July 2010.

29. [A10a] Björn Andersson, "Conjecture about global fixed-priority preemptive multiprocessor scheduling of implicit-deadline sporadic tasks: The utilization bound of SM-US(sqrt(2)-1) is sqrt(2)-1", 1st International Real-Time Scheduling Open Problems Seminar, July 6, 2010, Brussels, Belgium, in conjunction with the 22th Euromicro Intl Conference on Real-Time Systems, July 7-9, 2010, Brussels, Belgium

30. [MNP10a] Cláudio Maia, Luis Miguel Nogueira, Luis Miguel Pinho,"Evaluating Android OS for Embedded Real-Time Systems", Proceedings of the 6th International Workshop on Operating Systems Platforms for Embedded Real-Time Applications, Brussels, Belgium, July 2010. pp. 63-70.

31. [RA10] Gurulingesh Raravi, Björn Andersson, "Calculating an upper bound on the finishing time of a group of threads executing on a GPU: A preliminary case study", WiP session of 16th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA 2010), Macau SAR, P.R.C., Aug 23-25, 2010.

32. [MNP10b] Cláudio Maia, Luis Miguel Nogueira, Luis Miguel Pinho, "Experiences on the Implementation of a Cooperative Embedded System Framework," in Proceedings of the

8th International Workshop on Java Technologies for Real-Time and Embedded Systems, Prague, Czech Republic, August 2010. pp. 70-72.

33. [LES10] Jinkyu Lee, Arvind Easwaran, Insik Shin, "LLF Schedulability Analysis on Multiprocessor Platforms", IEEE Real-Time Systems Symposium (RTSS 2010), in Proceedings of the 31st IEEE Real-Time Systems Symposium November 30 - December 3, 2010, San Diego, CA, USA.
34. [LSE10] Jinkyu Lee, Insik Shin, Arvind Easwaran, "Online Robust Optimization Framework for QoS Guarantees in Distributed Soft Real-Time Systems", ACM and IEEE International Conference on Embedded Software (EMSOFT 2010), part of ESWEEK 2010.
35. [A10b] Björn Andersson, "A preliminary idea for an 8-competitive, log2 DMAX + log2 log2 (1/U) asymptotic-space, interface generation algorithm for two-level hierarchical scheduling of constrained-deadline sporadic tasks on a uniprocessor", 3rd Workshop on Compositional Theory and Technology for Real-Time Embedded Systems; November 30, 2010, San Diego, CA, USA (co-located with RTSS 2010)

Univ. of Porto

36. [SASM10] Pedro Silva, Luís Almeida, Mário Sousa, Ricardo Marau. Temporal behavior of Ethernet communications: Impact of the operating system and protocol stack. MICS 2010 - Int. Conf. on Models of Information and Communication Systems. Rabat, Morocco. 2-4 Nov 2010.

Catalonia (affiliated ULUND)

37. [CMVLVFG10] A. Camacho, P. Martí, M. Velasco, C. Lozoya, R. Villà, J.M. Fuertes, E. Griful. "Self-Triggered Networked Control Systems: an Experimental Case Study". In IEEE 2010 International Conference on Industrial Technology (ICIT2010) Valparaiso, Chile, March 2010.
38. [GVM10] D. García, M. Velasco, and P. Martí. "SimpleRTK: Minimal Real-Time Kernel for Time-Driven and Event-Driven Control". Research report ESAII-RR-10-01, Automatic Control Department, Technical University of Catalonia, Barcelona, Spain, January, 2010.

-- The above are new references, not present in the Y2 deliverable --

2.3 Interaction and Building Excellence between Partners

- UC3M and Pisa have started a collaboration on QoS-based resource management for multimedia applications. Currently, a PhD student from Pisa is on a one-year stay at UC3M site to collaborate on research on the integration of QoS resource managers in real-time operating systems.
- UC3M and UPM are collaborating in defining OS extensions in the form of QoS resource managers for multimedia systems in a distributed environment.
- Pisa and Catalonia continued to collaborate on event-driven scheduling to improve the efficiency and the performance of real-time controllers.
- SSSUP and Catalonia have been working on the design of an embedded control systems laboratory experiment in order to provide a first step toward a multidisciplinary education for embedded systems engineers. By joining the know-how of SSSUP on real-time kernels

and UPC on control systems, the experiment permits to flexibly work real-time systems and control systems in an integrated manner. Based on this laboratory experiment, the participation of UPC in the "ARTIST Graduate School on RT Kernels for Microcontrollers" and "PhD course on Real-Time Control Systems", reported below, have shown that this approach benefits students with different backgrounds.

- The SimpleRTK kernel developed at UPC [GVM10] has been assessed by SSSUP and the distribution of this kernel together with the ERIKA kernel developed by Evidence (Pisa, Italy) is under study.

-- The above is new material, not present in the Y1 deliverable --

2.4 Joint Publications Resulting from these Achievements

1. Kleopatra Konstanteli, Tommaso Cucinotta, Theodora Varvarigou, "Optimum Allocation of Distributed Service Workflows with Probabilistic Real-Time Guarantees," Springer Service Oriented Computing and Applications (SOCA), 2010.
2. T. Cucinotta, F. Checconi, G. Kousiouris, et al. “Virtualised e-Learning with Real-Time Guarantees on the IRMOS Platform,” to appear in Proceedings of the IEEE International Conference on Service-Oriented Computing and Applications (SOCA 2010), Perth, Australia, December 2010.
3. D. Kyriazis, A. Menychtas, G. Kousiouris, K. Oberle, T. Voith, M. Boniface, E. Oliveros, T. Cucinotta, S. Berger, “A Real-time Service Oriented Infrastructure,” to appear in Proceedings of the Annual International Conference on Real-Time and Embedded Systems (RTES 2010), Singapore, November 2010.
4. T. Cucinotta, D. Giani, D. Faggioli, F. Checconi, “Effective Real-Time Computing on Linux,” in Proceedings of the 12th Real-Time Linux Workshop (RTLWS 2010), Nairobi, Kenya, October 2010.
5. Qi Zhu, Yang Yang, Marco Di Natale, Eelco Scholte and Alberto Sangiovanni-Vincentelli, "Optimizing the Software Architecture for Extensibility in Hard Real-time Distributed Systems", IEEE Transactions on Industrial Informatics, vol. 6 issue 3, 2010.
6. Di Natale, M., Sangiovanni-Vincentelli, A., “Moving from Federated to Integrated Architectures: the Role of Standards, Methods and Tools”, Proceedings of the IEEE, vol.98, no.4, pp.603-620, April 2010.
7. Marco Di Natale, Haibo Zeng, “System Identification and Extraction of Timing Properties from Controller Area Network (CAN) Message Traces”, ETFA10 - IEEE International Conference on Emerging Technology and Factory Automation, Bilbao, Spain, 13-16 September 2010.
8. Haibo Zeng, Marco Di Natale, “Improving Real-Time Feasibility Analysis for Use in Linear Optimization Methods”, Euromicro Conference on Real-Time Systems (ECRTS), Bruxelles, July 6-9 2010
9. Haibo Zeng, Arkadeb Ghosal, Marco Di Natale, “Timing Analysis and Optimization of FlexRay Dynamic Segment”, ICESS Conference, 2010, Bradford UK, June29-July1, 2010
10. Arkadeb Ghosal, Haibo Zeng, Marco Di Natale, Yakov Ben Haim, “Computing Robustness of FlexRay Schedules to Uncertainties in Design Parameters” DATE Conference, Dresden, Germany March 2010.

11. P. Gai, M. Di Natale, Huascar Espinoza, Francois Terrier, Sébastien Gérard, Reinhold Heckmann, Christian Ferdinand, Giacomo Gentile, Nicola Ariotti, “Time Predictability from System-level Design to Task Implementations in Automotive Applications”, SAE Conference, April 11-14, Detroit, 2010
12. M. Di Natale, Haibo Zeng, “Time Determinism and Semantics Preservation in the Implementation of Distributed Functions over FlexRay”, SAE Conference, April 11-14, Detroit, 2010
13. Sébastien Gérard, Huascar Espinoza, François Terrier, Felice Tufo, Giacomo Gentile, Marco Di Natale and Bran Selic, “Modeling Languages for Real-time and Embedded Systems: Requirements and Standards-Based Solutions”, Model-Based Engineering of Embedded Real-Time Systems (MBEERTS), of LNCS, Springer, 2010.
14. Tullio Vardanega, Michael González-Harbour, Luís Miguel Pinho. "Language and Distribution Issues: Session Summary" - 14th International Real-Time Ada Workshop, Porto Venere (Italy), ACM SIGAda Ada Letters, Volume XXX, Number 1, pp. 91-103, pp 152- 162, April 2010.
15. P. Burgio, M. Ruggiero, F. Esposito, M. Marinoni, and G. Buttazzo, L. Benini, "Adaptive TDMA bus Allocation and Elastic Scheduling: a unified approach for enhancing robustness in multi-core RT systems", Proceedings of the 28th IEEE International Conference on Computer Design (ICCD 2010), Amsterdam, the Netherlands, October 3-6, 2010.
16. Pau Marti, Manel Velasco, Josep M. Fuertes, Antonio Camacho, and Giorgio Buttazzo, "Design of an Embedded Control Systems Laboratory Experiment", IEEE Transactions on Industrial Electronics, Vol. 57, No. 10, October 2010.

-- The above are new references, not present in the Y2 deliverable --

2.5 Keynotes, Workshops, Tutorials

Tutorial: Graduate Course on Embedded Control Systems: Theory and Practice
Scuola Superiore Sant’Anna, Pisa, Italy – June 14-18, 2010

Objectives: The course was aimed at providing the fundamental concepts of real-time computing systems, including scheduling, resource management and timing analysis; introducing the OSEK/VDX standards, taking as a reference implementation the Erika Enterprise kernel; showing how to apply such concepts in practice, with examples based on the Flex platform and the Microchip dsPIC DSC microcontrollers; teaching participants how to develop simple control applications using Erika Enterprise with code generation from functional models.

Organizers: Giorgio Buttazzo (Scuola Superiore Sant’Anna), Ettore Ricciardi (ISTI-CNR, Pisa).

URL: http://www.artist-embedded.org/artist/ARTIST-Embedded-Control-2009.html

ARTEMIS Summit (Held joinly with ITEA2)
Gennt, Belgium – October 26-27, 2010
iLAND annual project presentation.

WARM Workshop (Held joinly with RTAS 2010)
Stockholm, Sweden – April 12th, 2010
Presentation of iLAND ARTEMIS project highlights and achievements.

Miscellaneous

- The Scuola Superiore Sant'Anna of Pisa has been involved in a standardization effort for including the SCHED_DEADLINE scheduler into the mainline Linux kernel. To achieve this goal, some core kernel developers, such as Thomas Gleixner (responsible for the i386 support) and Paul McKenney (responsible for the Read-Copy-Update synchronisation machinery into Linux), have been invited to fruitful technical meetings about the real-time support in the Linux kernel.
- The Scuola Superiore Sant'Anna of Pisa joined the OSADL "Open Source Automation Development Lab", focusing on the use of open-source in the real-time and automation industry.
- The Scuola Superiore Sant'Anna of Pisa organised training events about the IRMOS Real-Time scheduler, specifically:
  - a 1-hour training seminar co-located with General Assembly meetings of the IRMOS Project at the University of Southampton;
  - a 2-hour training seminar during the WATERS 2010 workshop, co-located with the ECRTS 2010, in Bruxelles;
  - a 3-hours training seminar during the 12th Real-Time Linux Workshop (RTLWS 2010), in Nairobi (Kenya), at the University of Strathmore;
- Professor Stefan M. Petters at the Polytechnic of Porto, Portugal was chair of the 6th International Workshop on Operating System Platforms for Embedded Real-Time Applications (OSPERT 2010).
- Prof. Lucia Lo Bello was Track co-Chair for the "Information Technology in Automation" Track at the 15th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Bilbao, Spain. (URL: http://www.etfa2010.org/).

-- The above is new material, not present in the Y2 deliverable --

3. Milestones, and Future Evolution

3.1 Problems to be Tackled in Year 4 *(Jan 2011 – Dec 2011)*

In the next 12 months the cluster will investigate the follwing issues:

- Continue the development of software modules to support real-time control applications on the educational kit.
- Evaluate the partitioning algorithm on a specific multicore platform, using the Erika real-time kernel and investigate the allocation of dynamic applications with highly variable resource requirements.
- Investigate a method for positioning preemption points in the task code to reduce the cache related pre-emption delays in task execution times.
- Extend the Erika kernel to support the development of real-time applications with energy constraints.
- Improve resource usage strategies inside real-time middleware platforms.
- Develop algorithms for real-time scheduling on heterogeneous multiprocessors where at task can migrate between processors of the same type but cannot migrate between processors of different types. This is relevant because processors of the same type tend to have the same instruction set.
- Finalize the assessment of the Android OS and appropriate changes to its virtual machine environment, in the context of platforms for QoS applications in mobile systems.
- Investigate the real-time properties and subsequently real-time guarantees which can be established on a manycore operating system, like Barrelfish.
- Characterize the demands that event-driven control systems put into the computing platform.

- The above is new material, not present in the Y2 deliverable --

3.2 Current and Future Milestones

Year 1:

1. **Model resource reservation on multi-core architectures (Achieved)**. The concept of resource reservation has been extended to multi-core architectures, with the objective of achieving temporal protection among different applications.
2. **Educational kit for embedded control applications (Achieved)**. An educational kit for embedded systems, based on Microchip dsPIC technology, has been developed at the Scuola Superiore Sant'Anna of Pisa, in collaboration with Pavia, Evidence, Embedded Solutions, and Microchip Technology. It consists of a number of modules that can easily be composed depending on specific application purposes. The modules include a set of libraries to simplify the access to the hardware devices (sensors, servomotors, wireless modules) and a number of sample real-time control applications that can be easily replicated by the users.

3. **XtratuM hypervisor (Achieved)**. The XtratuM hypervisor has been developed at the Technical University of Valencia and has substituted the RTLinux distribution to provide full temporal and spatial isolation to real-time partitions and Linux partition. In particular, XtratuM has been ported to Leon2 processors (the processor used by ESA in the space) under a project funded by CNES (France) following the ARINC 653 standard. XtratuM can now be used as a test platform for the techniques proposed in this activity.

Year 2:

1. **Partitioning real-time applications on multi-core platforms (Achieved)**. The problem of partitioning parallel real-time applications on top of multicore platforms has been addressed for exploiting the available parallelism offered by modern architectures and optimizing the required resources. A method has also been provided for guaranteeing the timing constraints of the application.
2. **Deadline Scheduling on Linux (Achieved)**. The Linux kernel has been extended to support deadline-based scheduling algorithms and resource reservations.
3. **Software support for embedded control applications (Achieved)**. A set of modules to simplify the development of real-time control applications on the educational control kit has been developed at the Scuola Superiore Sant'Anna of Pisa, in collaboration with Pavia, Evidence, Embedded Solutions, and Microchip Technology. The modules include a set of libraries to simplify the access to the hardware devices (sensors, servomotors, wireless modules), a number of sample real-time control applications that can be easily replicated by the users, a real-time communication MAC protocol for wireless sensor networks.

Year 3:

1. **Resource reservations on multi-core platforms (Achieved)**. The work on multicore resource reservation has been extended by providing a methodology that can partition a parallel application on a set uniprocessor reservations with minimal bandwidth requirements.
2. **Deadline Scheduling on Linux (Achieved)**. A great effort has been done in this Cluster to introduce resource reservation and deadline-based scheduling (EDF) in the Linux operating system. The implementation is now complete and allows a more efficient support of real-time applications, enabling the implementation of advanced resource reservations techniques with adaptive bandwidth management.
3. **Power-Aware Resource Management (Achieved)**. The work started by the Scuola Superiore Sant'Anna of Pisa on providing kernel support for managing power and temperature constraints has been completed. A C library for the Erika kernel has been developed to manage real-time applications with power and temperature requirements. A uniform approach has been adopted to handle different types of resources, such as processors, memories, buses, wireless transceivers, and I/O devices.

-- The above is new material, not present in the Y2 deliverable --

3.3 Main Funding

The ArtistDesign NoE funds integration and building excellence with the partners, and with the European research landscape as a whole. Beyond this "glue" for integration and excellence, during Year 3 this activity has benefited from direct funding from:

ACTORS - Adaptivity and Control of Resources in Embedded Systems

ACTORS is a three-year research project within the European Commission's 7th Framework Programme involving the following ArtistDesign partners: Lund University, Scuola Superiore Sant'Anna, and Technical University of Kairserslautern. It addresses design of resource-constrained software-intensive embedded systems with high requirements on adaptivity and efficiency. Three techniques are combined: virtualization, feedback control, and data-flow programming models. Virtualization techniques such as reservation-based scheduling provide spatial and temporal separation of concerns and enforce dependability and predictability. Reservations can be composed, are easier to develop and test, and provide security support. Using feedback-based resource management, the resource allocation is based on a comparison of the actual resource utilization of, e.g., a set of activities or tasks, with the desired resource utilization. The difference is used for deciding how the resources should be dynamically allocated. Feedback control makes it possible to deal with uncertainties and variations in a controlled way and provides adaptivity to on-line changes in objectives, external conditions and use cases. By combining feedback control with resource reservations it is possible to handle incorrect reservations, reclaim and redistribute unused resources, and adjust to dynamic changes in resource requirements.

http://www.actors-project.eu/index.php/

PREDATOR - Design for predictability and efficiency

PREDATOR is a three-year focused-research project within the European Commission's 7th Framework Programme on Research, Technological Development and Demonstration. The following ArtistDesign partners are involved: Saarland University, Scuola Superiore Sant'Anna, Technische Universität Dortmund, and University of Bologna. The project is concerned with embedded systems that are characterised by efficiency requirements on the one hand and critical constraints on the other. This combination of requirements typically occurs in application domains such as automotive, aeronautics, multi-media and industrial automation.

http://www.predator-project.eu/

iLAND – m**I**dd**L**ew**A**re for deterministic dynamically reconfigurable **N**etworke**D** embedded systems

iLAND is a European project of Call 1 of ARTEMIS JTU. The technical coordination is carried by University Carlos III de Madrid. The high level objective of iLAND is to develop enabling technology and infrastructure of a modular component-based middleware for in-building Infrastructured NES that have strong needs for deterministic dynamic functional composition and reconfiguration. The goal is to improve system flexibility, scalability, and composability. Also, maintainability will be improved since spontaneous reconfiguration of the system will be supported. This will enable dynamic functionality reconfiguration (i.e., new node activation, removal of crashed or damaged nodes and reallocation of functionality, etc.).

http://www.iland-artemis.org

THREAD Spanish project

The following ArtistDesign partners are involved: Technical University of Madrid, University of Cantabria, and Technical University of Valencia.

http://polaris.dit.upm.es/%7Estr/proyectos/thread/

WASP - Wirelessly Accessible Sensor Populations

The WASP project (Coordinated by Philips Research) aims at narrowing the mismatch between research at the application level and the node and network level in a sensor network, by covering the whole range from basic hardware, sensors, processor, communication, over the packaging of the nodes. The emphasis in the project lays in the self-organisation and the services, which link the application to the sensor network.

http://www.wasp-project.org/

RESCORE (Real-time Scheduling on Multicores - PTDC/EIA/78141/2006), project funded by Portuguese Science Council (FCT).

REHEAT (Real-time scheduling on heterogeneous multicore architectures - PTDC/EIA-CCO/105716/2008), project funded by Portuguese Science Council (FCT).

RECOMP (Reduced Certification Costs for Trusted Multi-core Platforms - Artemis 100202), supported by the European Community through the Joint Undertaking ARTEMIS.

REJOIN (REal-time scheduling on multicore processors: addressing two open problems JOINtly.), project funded by the Portuguese-American Development Foundation (FLAD)

C3DE - Control, Communications and Real-time Computing in Distributed Embedded Nodes, CICYT DPI2007-61527, 2007-2009 (Spanish national project).

-- Changes wrt Y2 deliverable --

New projects contributed to support the reported research.

4. Internal Reviewers for this Deliverable

- **Karl-Erik Arzen** (University of Lund)
- **Luis Almeida** (University of Porto, Portugal)