IST-004527 ARTIST2
Network of Excellence
on Embedded Systems Design

Activity Progress Report for Year 4

JPIA-Platform
A Common Infrastructure for Adaptive Real-time Systems

Clusters:

Adaptive Real-Time

Activity Leader:

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

Policy Objective (abstract)

The objective of this activity is to show how current operating systems and network protocols have to be extended to support emerging real-time applications that exhibit a high degree of complexity and operate in dynamic environments.

The impact on operating system standards (like RT-POSIX and OSEK) as well as network protocols will also be taken into account.

Table of Contents

1. Overview of the Activity ... 3
1.1 ARTIST Participants and Roles ... 3
1.2 Affiliated Participants and Roles ... 3
1.3 Starting Date, and Expected Ending Date ... 3
1.4 Baseline ... 3
1.5 Problem Tackled in Year 4 ... 5
1.6 Comments From Year 3 Review ... 5
1.6.1 Reviewers' Comments ... 5
1.6.2 How These Have Been Addressed ... 6

2. Summary of Activity Progress ... 8
2.1 Previous Work in Year 1 ... 8
2.2 Previous Work in Year 2 ... 8
2.3 Previous Work in Year 3 ... 8
2.4 Final Results ... 9
2.4.1 Technical Achievements ... 9
2.4.2 Individual Publications Resulting from these Achievements ... 10
2.4.3 Interaction and Building Excellence between Partners ... 12
2.4.4 Joint Publications Resulting from these Achievements ... 12
2.4.5 Keynotes, Workshops, Tutorials ... 12

3. Milestones, and Future Evolution Beyond the NoE ... 15
3.1 Milestones ... 15
3.2 Indicators for Integration ... 15
3.3 Main Funding ... 15
3.4 Future Evolution Beyond the Artist2 NoE ... 16

4. Internal Reviewers for this Deliverable ... 16

1. Overview of the Activity

1.1 ARTIST Participants and Roles

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 – Technical 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.

1.2 Affiliated Participants and Roles

Team Leader: Ivo De Lotto – University of Pavia (Italy)
Role: kernel maintenance.

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

Team Leader: Lucia Lo Bello – University of Catania (Italy)
Role: Distributed real-time applications.

Team Leader: Pau Marti – Universitat Politècnica de Catalunya (Italy)
Role: control applications.

1.3 Starting Date, and Expected Ending Date

Starting date: September 1st, 2004

Ending date. August 31st, 2008.

1.4 Baseline

A research platform for real-time systems is composed of competencies, resources, and tools targeting at the development of control applications with performance and timing requirements.

A shared research platform is essential for experimenting new real-time software technology, including novel scheduling algorithms, resource management techniques, energy-aware policies and overload handling approaches to increase robustness and predictability. Platforms are also used as the basis for transfer research results to industry, as they allow teaching practical knowledge of the concepts and techniques.

Two platforms have been used within ARTIST2 to share competences and test methodologies for real-time operating systems:

1. SHARK (Soft and HArd Real-time Kernel) is an open source modular operating system developed at the ReTiS Lab of the Scuola Superiore Sant'Anna of Pisa running on Intel-based personal computers. It integrates novel real-time algorithms resulting from long-term research efforts. The kernel modularity allows the user to replace scheduling and mutual exclusion mechanisms without changing the application code. This was the main motivation for adopting this kernel. Shark is compliant with the POSIX standard interface.
2. ERIKA Enterprise is a minimal real-time kernel developed by Evidence s.r.l. for small microcontrollers with severe resource constraints. Erika is compliant with the OSEK standard interface, which makes it suitable for developing portable applications, especially in the automotive domain. It runs on a variety of platforms, including Microchip dsPic, ARM 7, Atmel AVR, H8, and Altera NIOS II. ERIKA implements innovative scheduling algorithms such as Fixed Priority with preemption thresholds, Stack Resource Policy (SRP), and Earliest Deadline First (EDF). The kernel is available in double licensing, both GPL and commercial, and has a minimum footprint of 800 bytes of code.

The Shark operating system was mainly used in the first three years of the project to test the effectiveness of novel scheduling and resource management algorithms, whereas the Erika kernel was adopted in the last years for experimenting the development of real-time applications on top of typical embedded platforms with limited resources, like memory, processing power, and energy.

As a shared platform for the Erika kernel we selected the FLEX board (developed by Evidence s.r.l.), which is a Microchip dsPIC evaluation board for embedded applications. The compact design of FLEX makes it suitable not only for development purposes, but also for a direct deployment in the working environment. Such a solution was selected for a number of reasons, including:

- Robust electronic design, which makes it suitable for non expert users, like students and people not familiar with electronic circuits.
- Modular architecture. The board exports most of the available pins of the dsPic and it is designed to be connected in a piggy-back fashion with other dauther boards that can be easily developed by the user for specific applications.
- Availability of a growing number of dauther boards and application notes. Available applications include servomotor control, accelerometer and gyroscope acquisition, bus to bus bridges, image processing from an embedded camera, wireless communication protocols, and various sensor acquisition and filtering.
- Full support of the Erika Enterprise real-time kernel, which allows the user to test a control application under different real-time scheduling algorithms.
- Low cost (in the order of 100 Euros per board).
- Availability of tools for schedulability analysis and automatic code generation.

RT-Druid is the development environment for ERIKA Enterprise. Based on Eclipse, RT-Druid allows writing, compiling, and analyzing an application in a comfortable environment. It is composed by a set of plugins for the Eclipse Framework, including:

- RT-Druid Core. The RT-Druid Core plug-in contains all the internal metamodel representation, providing a common infrastructure for the other plugins, together with ANT scripting support.

- RT-Druid Code Generator. The RT-Druid Code Generator plug-in implements the OIL file editor and configurator, together with target independent code generation routines for ERIKA Enterprise.
- RT-Druid Schedulability Analysis plug-in. The RT-Druid Schedulability Analysis plug-in provides the Schedulability Analysis framework, implementing algorithms like scheduling acceptance tests, sensitivity analysis, task offset calculation, and so on.
- RT-Druid aiT plug-in. This plug-in provides a coupling with AbsInt's (AbsInt Angewandte Informatik GmbH) aiT WCET Analyzer that computes tight bounds for the worst-case execution time of the tasks in your system. It is an integrating task response time analysis with code level timing analysis.

Web links

SHARK: http://shark.sssup.it/

Erika: http://www.evidence.eu.com/content/view/27/254/

FLEX: http://www.evidence.eu.com/content/view/114/204/

1.5 Problem Tackled in Year 4

In the Year 4, we concentrated on the development of an embedded platform for running real-time applications under severe resource constraints. When resources are limited, as in most embedded systems, the role of the operating systems is crucial for exploiting the available resources. In fact, the internal kernel mechanisms (like scheduling and resource management protocols) must be efficient and continuously monitor the consumption of the available resources, like processor, memory, and energy. The Erika Enterprise real-time kernel and the Flex board as a hardware platform were selected to make such experiments for the reasons explained in Section 1.4.

1.6 Comments From Year 3 Review

1.6.1 Reviewers' Comments

5.4.2 D07-ART-Y3 A common infrastructure for adaptive Real-time Systems (Platform)

ACCEPTED
The document is well written. It is concise and clearly exposes what has been achieved in year 3, dissemination and cooperation activities. The list of publication is impressive and there are a lot of collaborations between cluster partners around the Shark operating system. However the interaction with other clusters seems to remain a bit weak (there is just some mentioning of integrating component model on the OS and interactions with Marte activities). The future work and evolution is also a bit light. What will be the future of Shark? Is there any plan for industrialisation? Is there any plan to integrate tool with the operating system for monitoring its behaviour and ease development? This was discussed during the review meeting and these points will be addressed in the final review and project report.

1.6.2 How These Have Been Addressed

Comment:

The interaction with other clusters seems to remain a bit weak.

Answer:

In the last year there was an intense interaction with three other clusters:

- University Carlos III of Madrid and Polytechnical University of Madrid continued the collaboration with the cluster on Real-Time Components on QoS management of component-based embedded systems.
- Pisa, Pavia and University of Catalonia continued to interact with the cluster on Control for Embedded System to integrate real-time and control theory for implementing adaptive systems. New jitter/delay reduction methods have been proposed and evaluated to improve the performance of real-time control systems.
- Pisa also started a collaboration with the cluster on Execution Platforms (and in particular with Bologna, Saarland, Dortmund and ETH Zurich) for evaluating the effects of scheduling algorithms on cache memories, with the purpose of reducing variability in the task worst-case execution times. The idea is that preemptive scheduling destroys cache data and prefetch queues, so a number of experiments were discussed to verify how scheduling affects execution times.

Comment:

The future work and evolution is also a bit light. What will be the future of Shark?

Answer:

Shark is actually used in more than 30 universities all over the world for education in real-time systems courses and for evaluating the effect of novel scheduling algorithms on the performance of real-time control applications. All the algorithms and applications developed within ARTIST2 are available on the web in a repository.

URL: http://shark.sssup.it/repository/shark

A repository for all real-time applications developed using the Shark kernel was created to facilitate the users in the development of new real-time software. The repository includes a folder of supported applications and a folder of all unsupported software. The supported applications are tested and maintained by the developers to be consistent with the current kernel version, while the unsupported folder includes all programs, demos, and advanced applications developed under Shark and made available by the maintenance team.

URL: http://shark.sssup.it/repository/applications

URL: http://shark.sssup.it/repository/unsupported

A repository of all kernel modules developed for the Shark operating system was created to facilitate the users in the development of new kernel mechanisms. The modules include scheduling modules (implementing periodic schedulers, aperiodic servers, or overload management policies) and resource modules (implementing concurrency control protocols for accessing shared resources). Each module contains the C source code and required headers compliant with the Shark module specifications.

URL: http://shark.sssup.it/repository/modules

On going activities on Shark include:

- Porting on hardware emulators for facilitating the development of real-time applications on different hardware platforms.
- Support for multicore architectures.
- Development of Scilab/Scicos modules for automatic code generation to simplify the development of control applications by non expert programmers.

Comment:

Is there any plan for industrialisation?

Answer:

After testing a set of new real-time algorithms on embedded control systems using the Shark platform, a much smaller platform has been introduced in the last year, the FLEX+Erika platform, to develop applications in a more realistic environment with severe resource constraints. The FLEX+Erika platform was developed in collaboration with three industrial partners working in the field of embedded systems:

- Michrochip Technology pushed the production of the FLEX board as a low-cost real-time platform for the easy development of embedded applications using 16-bit microcontrollers (as the dsPIC30 and the dsPIC33).
- Embedded Solutions designed the hardware and is in charge of producing the board and assembling the electronic components.
- Evidence srl is responsible for the low level software, including the support for the Erika kernel and the drivers for the peripherals.

These three companies combined their respective skills on real-time systems and electronic boards development, to create a complete, easy-to-use, and compact solution for the development of complex real-time applications based on the Microchip dsPIC® DSC microcontroller.

Comment:

Is there any plan to integrate tool with the operating system for monitoring its behaviour and ease development?

Answer:

RT-Druid is the development environment for ERIKA Enterprise. Based on Eclipse, RT-Druid allows writing, compiling, and analyzing an application in a comfortable environment. It is composed by a set of plugins for the Eclipse Framework. See Section 1.4 for additional details.

2. Summary of Activity Progress

2.1 Previous Work in Year 1

Initial definition of the operating system and network features. *The SHARK operating system developed at the Scuola Superiore Sant'Anna of Pisa has been identified (for the reasons explained in* Deliverable 2-2 JPIA-a-ART-Y1*) as the most suited kernel for building a common infrastructure to perform advanced experiments on real-time systems.*

2.2 Previous Work in Year 2

Deployment of a working platform for experimenting RTOS and network development. *The SHARK operating system was upgraded according to the partners' needs and deployed on each partner site. A specific workshop has been organized in Pontedera (Pisa) to teach partners how to use the kernel for writing a real-time application and how to write new scheduling and resource modules.*

2.3 Previous Work in Year 3

Extensive testing was performed on Shark to identify algorithms and tools to support adaptive RT systems. Specific applications were developed under Shark by the cluster members. Examples are listed below.

- **Ball balancing**. This control application has been developed by the Scuola Superiore Sant'Anna, in collaboration with the University of Pavia (for sensors and actuators interfacing), Evidence (for kernel support), and Lund (for controller implementation). A two-degree of freedom device has been built using two servomotors and a camera has been used to track the position and velocity of a ball moving on the plate.
- **Inverted pendulum**. This application has been developed by University of Catalonia (affiliated to Pisa). Shark was used to perform sensory acquisition, actuation, control and to test overload management strategies.
- **Mobile Robots**. This application has been developed by University of Aveiro. An inverted pendulum has been mounted on a robot car, which has been controlled to keep the pole in a vertical position.
- **Overload management techniques**. A number of multitask real-time applications have been developed at the Scuola Superiore Sant'Anna to test the behaviour of real-time systems under overload conditions. *Resource Reservations* and *Elastic Scheduling* techniques have been evaluated to cope with transient and permanent overload conditions. *Resource Reservations* techniques basically isolate the temporal behaviour of a task (or subset of tasks) protecting the rest of the systems from potential overruns. On the other hand, *Elastic Scheduling* provides an effective solution to cope with permanent overload conditions. According to this method, task utilizations are treated as flexible springs that can be compressed (by enlarging periods) to reduce the load up to a desired value. Such novel techniques (not yet available in commercial operating systems) have been implemented into Shark as basic scheduling modules to be tested and evaluated in actual control applications.
- **Feedback Scheduling in SHaRK: a First Approach**. Research Report ESAII-RR-06-13, Automatic Control Department, Technical university of Catalonia, July 2006 (http://www.upcnet.es/~pmc16/shark06.pdf). This work reports basic modifications done

in the shark kernel to facilitate the application of existing feedback scheduling results. No particular application was implemented. Unofficial modification of the S.Ha.R.K. kernel available for download (http://www.upcnet.es/~pmc16/all.tar). Unofficial Knoppix ISO with S.Ha.R.K. available for download (http://www.upcnet.es/~pmc16/knoppix.iso). Authors: Josep Guardia, Pau Martí, Manel Velasco and Rosa Castañé.

- **Porting of HOLA-QoS**. The group at UPM is continuing the porting of HOLA-QoS on top of SHARK. Up till now, the lower levels (Resource Manager) have been ported and a new version of the quality manager (higher layers) is being developed. When it will be ready, the software will run on top of SHARK, so that the negotiation and optimization features of HOLA-QoS can be used in SHARK applications.
- **Cibermouse client**. University of Aveiro is developing a Shark client for the CiberMouse@RTSS2006 students design competition. The code can be found here: http://www.ieeta.pt/~lau/web_ciberRTSS/tools.htm
- **Video processing**. TUKL is using Shark as a platform for video processing applications, in particular adaptive resource management for user quality. Algorithms for stream adaptation have been implemented and evaluated on Shark. Representatives from TUKL participated in training for Shark.
- **Education at TUKL**. TUKL is using the Shark kernel in undergraduate education. Two labs have been: one in which students develop a scheduling algorithm, which they analyse theoretically and practically as implementation on an operating system, i.e., Shark. In the other lab, students implement a simple video processing algorithm which they implement on Shark, learning implementation and overhead issues.
- **Traffic Smoothing Techniques**. University of Catania (affiliated to Pisa) is implementing Traffic Smoothing Techniques on the Shark OS, to make experiments on distributed real-time systems.

At the same time, a new platform was developed in the third year for experimenting the development of real-time applications on top of typical embedded platforms with limited resources. The new platform, consisting of the FLEX board and the Erika Enterprise kernel, is described in Section 1.4.

2.4 Final Results

2.4.1 Technical Achievements

In the Year 4, we concentrated on the development of an embedded platform for running real-time applications under severe resource constraints. The following control applications have been developed using Erika Enterprise as a real-time kernel and Flex as a hardware platform.

Ball and plate balancing. A two-degrees-of-freedom balancing device has been built at the Scuola Superiore Sant'Anna to control the trajectory of a ball on a plate actuated by two servomotors. The position of the ball is detected by a resistive touch screen mounted on the plate. http://www.evidence.eu.com/content/view/276/266/

Visual tracking. A visual tracking application was developed at the Scuola Superiore Sant'Anna, where a moving ball was followed by a mobile CMOS camera using the FLEX board. The CMOS Camera is capable of returning JPEG images to the connected FLEX board hosting a Microchip dsPIC. The Flex Board also controls two servomotors which are used to

articulate the camera, thereby maintaining the focus on the rolling ball.
http://www.evidence.eu.com/content/view/277/266/

Inverted Pendulum at SUSPI. An inverted pendulum was controlled using the FLEX board with Scilab/Scicos at SUPSI (Scuola Universitaria Professionale della Svizzera Italiana), Lugano, Switzerland. The FLEX Base Board and the FLEX Multibus Board with a CAN module were used for swinging-up and maintaining the inverted equilibrium. The Source code was entirely generated using Scilab/Scicos, an automatic code generator for control systems.
http://www.evidence.eu.com/content/view/274/266/

Inverted Pendulum at UPC. An inverted pendulum was implemented using the FLEX board and Erika by the Distributed Control Systems group at the Automatic Control Department, Technical University of Catalonia, Barcelona, Spain. The FLEX Board and the FLEX Multibus Board with a RS232 module was used for both swinging-up and maintaining the inverted equilibrium.
http://paginespersonals.upcnet.es/~pmc16/08EvidenceNoteRTpend.zip

DC Motor control. A DC servomotor was controlled using the FLEX board with Scilab/Scicos at SUPSI (Scuola Universitaria Professionale della Svizzera Italiana), Lugano, Switzerland. The FLEX Base Board and the FLEX Multibus Board with a CAN module were used for Servo control of a DC Motor. The Source code was entirely generated using Scilab/Scicos, an automatic code generator for control systems.
http://www.evidence.eu.com/content/view/273/266/

Hexapode robot control. An 18-DOF hexapod robot was completly designed and developed at the University of Florence by Andrea Foschi in 2005. It was later tamed by Marco Natalini and Alessandro Mambelli using Evidence Srl's FLEX Light board and ERIKA kernel. The main purpose for adopting FLEX was due to its low-cost development kit that permits easy addition of features, i.e., sensors and behaviour. Since then, a number of students have worked on this hexapod. http://www.evidence.eu.com/content/view/261/266/

Educational experiments. An embedded control system was developed with the Erika+Flex platform at the Automatic Control Department of the Technical University of Catalonia (Spain), with the purpose of setting a laboratory experiment for educational purposes. A real-time control of dynamical system was designed to drive students to a better understanding and integration of the diverse theoretical concepts that often come from different disciplines such as real-time and control systems.

2.4.2 Individual Publications Resulting from these Achievements

Scuola Superiore Sant'Anna of Pisa

1. Giorgio Buttazzo, "Artificial Consciousness: Hazardous Questions (and Answers)", Journal of Artificial Intelligence in Medicine, Elsevier, to appear.
2. Gianluca Franchino, Giorgio Buttazzo, and Tullio Facchinetti, "Properties of BuST and Timed Token Protocols in Managing Hard Real-Time Traffic", Proceedings of the 13th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2008), Hamburg, Germany, September 15-18, 2008.

3. Enrico Bini, Marco Di Natale, and Giorgio Buttazzo, "Sensitivity Analysis for Fixed-Priority Real-Time Systems", Real-Time Systems, Vol. 39, No. 1-3, pp. 5-30, August 2008.
4. Yifan Wu, Enrico Bini, and Giorgio Buttazzo, "A Framework for Designing Embedded Real-Time Controllers", Proceedings of the 14th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA 2008), Kaohsiung, Taiwan, August 25-27, 2008.
5. Gianluca Franchino, Tullio Facchinetti, and Giorgio Buttazzo, "Time Properties of the Bust Protocol Under the NPA Budget Allocation Scheme", Proceedings of the Conference on Design, Automation and Test in Europe (DATE 2008), Munich, Germany, 10-14 March 2008.
6. Gianluca Franchino, Giorgio Buttazzo, and Tullio Facchinetti, "BuST: Budget Sharing Token Protocol for Hard Real-Time Communication", Proceedings of the 12th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2007), Patras, Greece, September 2007.

Universtity of Pavia

7. Tullio Facchinetti, Gianluca Franchino, and Giorgio Buttazzo, "Distributed Coordination Protocol for the Connectivity Maintenance in a Network of Mobile Units", IEEE Proceedings of the International Conference on Advances in Mesh Networks (MESH 2008), Cap Esterel, France, August 25-31, 2008.

Universtity of Catalonia

8. Pau Martí, Manel Velasco and Giorgio Buttazzo, "AN EMBEDDED REAL-TIME CONTROL SYSTEMS LABORATORY ACTIVITY", Research report ESAII-RR-08-03, Automatic Control Dept., Technical University of Catalonia, Barcelona, Spain, Aug. 2008.

Universtity of York

9. Osmar M. dos Santos, and Andy Wellings (2008), Run Time Detection of Blocking Time Violations in Real-Time Systems, 14th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications.
10. Osmar Marchi dos Santos and Andy Wellings (2008), Blocking time monitoring in the Real-Time Specification for Java, JTRES '08: Proceedings of the 6th international workshop on Java technologies for real-time and embedded systems.
11. R.I. Davis, A. Zabos, A. Burns. Efficient Exact Schedulability Tests for Fixed Priority Real-Time Systems. IEEE Transactions on Computers, 2008.
12. A. Zuhily, A. Burns. Exact Response Time Scheduling Analysis of Accumulatively Monotonic Multiframe Real Time Tasks. Proceedings of the 5th International Colloquium on Theoretical Aspects of Computing (ICTAC), 2008.
13. A. Zuhily, A. Burns. Exact Scheduling Analysis of Accumulatively Monotonic Multiframe Tasks Subjected to Release Jitter and Arbitrary Deadlines. Proceedings of the 13th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), 2008.
14. P. Martins, A. Burns. On the Meaning of Modes in Uniprocessor Real-Time Systems. Proceedings of the 23rd Symposium on Applied Computing, 2008.
15. A. Burns, S. Baruah. Sustainability in Real-time Scheduling. Journal of Computing Science and Engineering, 2008.
16. Fengxiang Zhang, Alan Burns. Analysis of Hierarchical EDF Pre-emptive Scheduling. Proceedings of IEEE Real-Time Systems Symposium, Dec. 2007.
17. R.I. Davis, A, Burns. Robust Priority Assignment for Fixed Priority Real-Time Systems. Proceedings of IEEE Real-Time Systems Symposium, Dec. 2007.

2.4.3 Interaction and Building Excellence between Partners

University Carlos III of Madrid and Polytechnical University of Madrid continued their collaboration with the cluster on Real-Time Components on QoS management of component-based embedded systems.

The Scuola Superiore Sant'Anna of Pisa, the University of Pavia and the University of Catalonia continued to interact with the cluster on Control for Embedded Systems to integrate real-time and control theory for implementing adaptive systems. New jitter/delay reduction methods have been proposed and evaluated to improve the performance of real-time control systems. External collaborations on this topic were also established with the University of Illinois at Urbana Champaign (Prof. Lui Sha, Prof. Tarek Abdelzaher, and Prof. Marco Caccamo), the University of Virginia (Prof. John Stankovic), the University of North Carolina at Chapel Hill (Prof. Sanjoy Baruah), and the University of Halmstad, Sweden (Prof. Bertil Svensson).

The Scuola Superiore Sant'Anna of Pisa also started a collaboration with the cluster on Execution Platforms (and in particular with Bologna, Saarland, Dortmund and ETH Zurich) for evaluating the effects of scheduling algorithms on cache memories, with the purpose of reducing variability in the task worst-case execution times. The idea is that preemptive scheduling destroys cache data and prefetch queues, so a number of experiments were discussed to verify how scheduling affects execution times.

2.4.4 Joint Publications Resulting from these Achievements

1. Gianluca Franchino, Giorgio Buttazzo, and Tullio Facchinetti, "Properties of BuST and Timed Token Protocols in Managing Hard Real-Time Traffic", Proceedings of the 13th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2008), Hamburg, Germany, September 15-18, 2008.
2. Tullio Facchinetti, Gianluca Franchino, and Giorgio Buttazzo, "Distributed Coordination Protocol for the Connectivity Maintenance in a Network of Mobile Units", IEEE Proceedings of the International Conference on Advances in Mesh Networks (MESH 2008), Cap Esterel, France, August 25-31, 2008.
3. Gianluca Franchino, Tullio Facchinetti, and Giorgio Buttazzo, "Time Properties of the Bust Protocol Under the NPA Budget Allocation Scheme", Proceedings of the Conference on Design, Automation and Test in Europe (DATE 2008), Munich, Germany, 10-14 March 2008.
4. Gianluca Franchino, Giorgio Buttazzo, and Tullio Facchinetti, "BuST: Budget Sharing Token Protocol for Hard Real-Time Communication", Proceedings of the 12th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2007), Patras, Greece, September 2007.
5. Pau Martí, Manel Velasco and Giorgio Buttazzo, "AN EMBEDDED REAL-TIME CONTROL SYSTEMS LABORATORY ACTIVITY", Research report ESAII-RR-08-03, Automatic Control Dept., Technical University of Catalonia, Barcelona, Spain, Aug. 2008.

2.4.5 Keynotes, Workshops, Tutorials

Keynote talks:

1. "Scheduling and Resource Management"
   Given by Giorgio Buttazzo (Scuola Superiore Sant'Anna, Pisa, Italy)
   Autrans, France – Septembe 9, 2008.

Workshops:

1. WCET 2008: Worst Case Execution Time Analysis
   Prague, July 1, 2008
   URL: http://www.artist-embedded.org/artist/WCET-08.html

2. RTN 2008: Real-Time Networks
   Prague, July 1, 2008
   URL: http://www.hurray.isep.ipp.pt/rtn08/index.php

3. OSPERT 2008: Worst-Case Execution Time Analysis
   Prague, July 1, 2008
   URL: http://www.cs.unc.edu/~anderson/meetings/ospert08/OSPERT.html

Courses:

1. P. Gai (Evidence), G. Buttazzo and M. Marinoni (Scuola Superiore Sant'Anna, Pisa). Course on “Real-Time Kernels for Microcontrollers: Theory and Practice”, Pisa, Italy, June 23-25, 2008.
   URL: http://www.evidence.eu.com/content/view/271/275/

2. M. Velasco (University of Catalonia, Spain). Course on “Real-Time Control Systems”, Pisa, Italy, April 2-17, 2008.

Competitions:

1. L. Almeida, N. Lau, P. Pedreiras and A. Pereira. CyberMouse@RTSS2007, Phoenix, USA, Dec. 2007. Students design competition within the scope of RTSS 2007. Similar to a satellite workshop but targeting students and where students have to develop the control software for a small robot and run it against the other teams. http://www.ieeta.pt/~lau/web_ciberRTSS07/

Contribution to Standards:

POSIX

The University of Cantabria (UC) has continued participation in the POSIX standard. There is currently a new revision of the standard being produced with technical corrigenda, and the UC participates in the debate and ballot process. Initial steps have been taken in the Real-Time System Services Working Group to start a revision of the POSIX.13 standard that defines the real-time profiles. The UC is also participating in the revision of the POSIX-Ada bindings, which is a project that is just starting.

OMG–MARTE

In this period we continue working in the technical activities of the OMG, attending the Technical Meeting in San Diego from 26 to 30 March 2007, and sending a preliminary submission in response to the UML Profile for Modeling and Analysis of Real-Time and Embedded systems (MARTE) request for proposals. The submission was presented to the RTESS (Real-Time Embedded and Specialized Systems) Platform Task Force and was very well received. The standard has now been approved.

UC will continue to work in the Finalization Task Force of the UML Profile for Modeling and Analysis of Real-Time and Embedded systems (MARTE), in order to solve the issues that may be raised by the industrial community about MARTE, and ensure its applicability in the modelling of platforms that can deal with flexible scheduling technologies.

3. Milestones, and Future Evolution Beyond the NoE

3.1 Milestones

1. Year1 **(achieved)**: Initial definition of the operating system and network features. *The SHARK operating system developed at the Scuola Superiore Sant'Anna of Pisa has been identified (for the reasons explained in* Deliverable 2-2 JPIA-a-ART-Y1*) as the most suited kernel for building a common infrastructure to perform advanced experiments on real-time systems.*
2. Year2 **(achieved)**: Deploy a working platform for experimenting RTOS and network development. *The SHARK operating system was upgraded according to the partners' needs and deployed on each partner site. A specific workshop has been organized in Pontedera (Pisa) to teach partners how to use the kernel for writing a real-time application and how to write new scheduling and resource modules.*
3. Year3 **(achieved)**: Perform extensive testing to identify algorithms and tools to support adaptive RT systems. Participate in the evolution of RTOS standards by introducing advanced scheduling methods for enhancing the predictability of real-time systems and handle their increased complexity. *Extensive tests were performed on Shark to identify algorithms and tools to support adaptive RT systems. Several applications were developed under Shark by all cluster members.*
4. **Year4: Identify the problems to be solved for developing a component-based real-time operating system. Provide operating system support for real-time embedded systems running on platforms with severe resource constraints.**

3.2 Indicators for Integration

The indicators for integration will be of different kinds, including:

- real-time control applications developed by the joint collaboration among partners, each focusing on a different aspect (e.g., control, scheduling, analysis, languages, etc.) related to his peculiar expertise;
- specific scheduling and resource modules developed by the partners and integrated in the kernel;
- new device drivers developed for the Shark kernel by the cluster members;
- joint publications of research papers;
- joint events (e.g., workshops, summer schools, graduate courses, etc.) organized by the cluster members to spread real-time technologies.

3.3 Main Funding

The basic research on real-time operating systems and advanced scheduling techniques come from the following National and European projects:

Italian National project MPS2007, founded by Fondazione Monte dei Paschi di Siena: "Distributed Real-Time Embedded Systems for Monitoring Physiological Parameters".

European Strep project: "FRESCOR - Framework for Real-Time Embedded Systems based on Contracts".

3.4 Future Evolution Beyond the Artist2 NoE

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.

Within ARTIST2 we mainly focused on defining the desired features and the critical problems that need to be solved at the technical level to decouple kernel mechanisms between them and from the application. This work can be continued to identify the problems to be solved for developing a fully component-based real-time operating system. Another important issue to be addressed is the extension of current uniprocessor RTOSs to multicore systems, with the objective of making optimal usage of the available CPUs, as well as minimizing power consumption.

4. Internal Reviewers for this Deliverable

- Gianluca Franchino (University of Pavia)
- Marco Caccamo (University of Illinois at Urbana Champaign)