

OSPERT 2016

The 12th annual workshop on Operating Systems Platforms for Embedded Real-Time applications.

July 5, 2016. Toulouse, France. Held in conjunction with ECRTS'16.

CALL FOR CONTRIBUTIONS

OSPERT is a forum for researchers and engineers working on (and with) **Real-Time Operating Systems (RTOSs)** to present recent advances in RTOS technology, to promote new and existing initiatives and projects, and to identify and discuss the challenges that lie ahead. The workshop, now in its eleventh year, provides the RTOS community with an opportunity to meet, to exchange ideas, to network, and to discuss future directions.

OSPERT'16 strives for an inclusive and diverse program and solicits a range of varied contributions. To this end, the following types of submissions are sought:

1. proposals for presentations (including talks on open problems, **demos & tutorials**, calls to action, etc.);
2. proposals for reports on empirical experiments (including **replication studies**, preliminary experiments preceding a full conference submission, and negative experience reports discussing failed approaches); and
3. **technical papers** (including short work-in-progress papers and full workshop papers).

Important Dates

April 28th, 2016

submission deadline

May 27th, 2016

acceptance notification

June 10th, 2016

submission of camera-ready papers

July 5th, 2016

workshop day

July 6-8, 2016

ECRTS Conference

Submission Formats

Presentation Proposals

abstract (~500 words)

Short WiP Papers

up to three A4 pages

Full Workshop Papers

up to six A4 pages

Workshop Chairs

Robert Kaiser

RheinMain University of Applied Sciences

Marcus Völz

University of Luxembourg

SCOPE

OSPERT'16 is open to all topics related to providing a reliable operating environment for real-time and embedded applications.

Developers of embedded RTOSs are faced with many challenges arising from two opposite needs: on the one hand there is a need for extreme resource usage optimization (processor cycles, energy, network bandwidth, etc.), and on the other hand there are also increasing demands in terms of scalability, flexibility, isolation, adaptivity, reconfigurability, predictability, serviceability, and certifiability, to name a few. Further, while special-purpose RTOSs continue to be used for many embedded applications, real-time services are also increasingly introduced and used in general-purpose operating systems, and market pressures continue to blur the lines between the two formerly distinct classes of operating systems. Notable examples are the various flavors of real-time Linux that support time-sensitive applications, the emergence of commercial and open-source real-time hypervisors, as well as the growth in features and scope of embedded OS and middleware specifications such as AUTOSAR.

OSPERT'16 is dedicated to the advances in RTOS technology required to address these trends. As such, areas of interest include, *but are not limited to*, the following topics:

- Case studies and experience reports
- Certification and verification of RTOSs and middleware
- Coordinated management of multiple resources
- Dynamic reconfiguration and upgrading
- Empirical comparisons and evaluations of RTOSs
- Flexible processor, memory, and I/O scheduling
- Interaction with reconfigurable hardware
- Operating system standards (e.g., AUTOSAR, ARINC, POSIX, etc.)
- Power and energy management
- Quality of Service guarantees
- Real-time Linux variants
- Real-time virtualization and hypervisors
- RTOSs for manycore platforms
- Scalability, from very small scale embedded systems to full-fledged RTOSs
- Security and fault tolerance for embedded real-time systems
- Support for multiprocessor architectures
- Support for component-based development

Visit <http://www.cs.hs-rm.de/~kaiser/events/ospert16/> for further details.

OSPERT is co-located with ECRTS, the premier European venue for presenting research into the broad area of real-time and embedded systems. Along with RTSS and RTAS, ECRTS ranks as one of the three top international conferences on real-time systems.

