Every eighteen months during the last thirty years has seen the power of the computer that can be built on a silicon chip double — this has now come to a halt. Instead, chip manufacturers build multiple computers — or cores — on each chip; nearly all PCs are now ‘dual’ or ‘quad’ core, and the number of cores it is possible to put on each chip is growing exponentially.

**KEY INNOVATION**

Building software for these multicore systems requires radically new software development technologies that can exploit the platform. Instead of programming a single core, the cores have to be programmed to work together in a coordinated way, and in a way that scales with the numbers of cores. Many expect 100,000-core platforms to become commonplace, and the best predictions are that core failures on such an architecture will be common, perhaps one an hour. Hence we require a programming model that is not only highly scalable but also reliable.

The RELEASE project develops the first a scalable concurrency-oriented programming infrastructure and its associated tool set, and hence aims to reduce development times of multicore solutions while delivering increased reliability.

**TECHNICAL APPROACH**

Our platform builds on the Erlang language and Open Telecom Platform (OTP) libraries. Erlang is a functional programming language. Its concurrency-oriented programming paradigm is novel in being very high level, predominantly stateless, and having both parallelism and reliability built-in rather than added-on. Some of the principles of the Erlang philosophy are as follows. Share nothing implies that isolated processes do not share memory and variables are not reusable, i.e. once a value is assigned it cannot be changed. Let it crash is a non-defensive approach that lets failing processes to crash, and then other processes detect and fix the problem. Erlang/OTP has inherently scalable computation and reliability models, but in practice at the beginning of the RELEASE project scalability was constrained by aspects of the language, Virtual Machine (VM) and toolset.

The RELEASE consortium attacks these problems at three levels:

- We evolve the Erlang VM — which implements Erlang on each core — so that it can work effectively in large-scale multicore systems.
We also evolve the language to Scalable Distributed (SD) Erlang and add the framework to provide constructs to control how computations are spread across multicore platforms, and coordination patterns to allow SD Erlang to effectively describe computations on large platforms, while preserving performance portability.

On top of the language and the VM we develop a scalable Erlang infrastructure to integrate multiple heterogeneous clusters.

We have developed, improved, and made open source releases at http://www.release-project.eu/ of five concurrency tools: two online/offline profiling tools, i.e. Perceptor and DTrace/SystemsTap, a prototype visualisation tool, the extended refactoring tool Wrangler, and the concurrency error detection tool Concurerror.

As a case study we have adapted Sim-Diasco, a substantial distributed Erlang simulation engine, to be reliable and more scalable.

Future plans include further concurrency improvements to the Erlang VM, deployment and evaluation of the tools at scale for SD Erlang to become a part of the standard Erlang OTP library, and performance evaluation of the scalable infrastructure on an IBM Blue Gene/Q.

PROJECT PARTNERS COUNTRY
University of Glasgow UK
Heriot-Watt University UK
University of Kent UK
Erlang Solutions Ltd UK
Ericsson AB Sweden
Institute of Communication and Computer Systems Greece
Electrotechnical Institute, France SA (EDEF) France
Uppsala Universitet Sweden

ACKNOWLEDGEMENTS
We would like to thank all our RELEASE colleagues. This work is supported by the European Union grant RI3-CT-2005-026133 ‘SCEnce: Symbolic Computing Infrastructure in Europe’, IST-2011-287510 ‘RELEASE: A High-Level Paradigm for Reliable Large-Scale Server Software’, and by the UK’s Engineering and Physical Sciences Research Council grant EP/G035181/1 ‘HPC-GAP: High Performance Computational Algebra and Discrete Mathematics’.

GENERAL PROJECT INFORMATION
Project coordinator: University of Glasgow Contact person: Phil Trinder School of Computing Science, University of Glasgow 612-802 Glasgow, United Kingdom Tel: +44 141 330 3627 Fax: +44 141 330 4913 Ph: +44 141 330 4913 Phil.Trinder@glasgow.ac.uk Project website: www.release-project.eu Community contribution to the project: 2,400,000 Euro Project start date: 01.10.2011 Duration: 36 months

BIBLIOGRAPHY


To exploit such large platforms, programmers need to be able to understand how their programs are behaving in practice. We build online SD Erlang monitoring and visualization tools to enable programmers to profile and visualize their SD Erlang applications; to refactor Erlang programs to run scalable and efficiently under SD Erlang; and to debug SD Erlang systems.

DEMONSTRATION AND USE
We demonstrate the effectiveness of the RELEASE approach in two case studies. EDF will port the Sim-Diasco simulation framework to SD Erlang on the Blue Gene parallel computing platform. Sim-Diasco (SIMulation of Discrete systems of All CALEs) is a distributed engine for large scale discrete simulations implemented in Erlang. The engine is able to handle more than one million relatively complex model instances using a hundred of cores.

In an example of commercial use, Erlang Solutions has developed a deployment and management infrastructure Wombat to exploit multiple heterogeneous cluster and cloud resources.

SCIENTIFIC, ECONOMIC AND SOCIETAL IMPACT
The presence of major European industrial players such as Ericsson and EDF in the consortium enables rapid commercialisation of the project outputs, enhancing European competitiveness in the software development market and ultimately leading to new high technology jobs in Europe. The Erlang Solutions SME will gain additional revenues from marketing deployment and management infrastructure Wombat developed in the project. Ericsson exploits the new technology in new products and services, and uses existing products to emerging hardware platforms to maintain their competitive position. EDF is working on simulation of smart energy grids using the Sim-Diasco simulation engine to model times more accurately than the previous version, leading to more efficient electricity supply and potentially to lower energy costs.