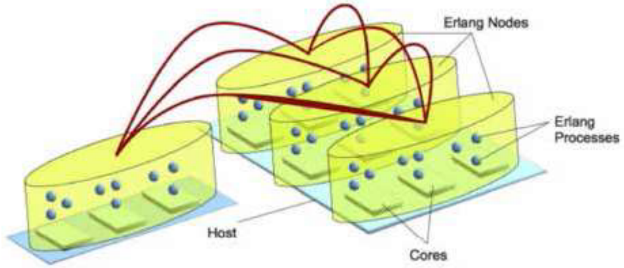
Every eighteen months during the last thirty years has seen the power of the computer that can be built on a silicon chip double – this has now come to a halt. Instead, chip manufacturers build multiple computers – or cores – on each chip: nearly all PCs are now 'dual' or 'quad' core, and the number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

By Natalia Chechina and Phil Trinder



© kIRr/Shutterstock

The RELEASE Project

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[1] 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 crush 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 RELASE project scalability was constrained by aspects of the language, Virtual Machine (VM) and toolset.

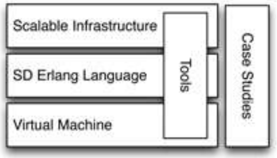
The RELEASE consortium attacks these problems at three levels:

- 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 adapt the OTP 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.



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 scalably 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-Diasca simulation framework[2] to SD Erlang on the Blue Gene parallel computing platform. Sim-Diasca (SIMulation of DIscrete systems of All SCAles) 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[3] 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 to move existing products to emerging hardware platforms to maintain their competitive position. EDF is working on simulation of smart energy grids using the Sim-Diasca simulation engine to model times more accurately than the previous version, leading to more efficient electricity supply and potentially to lower energy costs.

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
Electricité de France SA (EDF)	France
Uppsala Universitet	Sweden

KEY ACHIEVEMENTS TO DATE

- To improve the scalability of the Erlang VM we have made five important changes and architectural improvements that have been included in the Erlang/OTP release R16B03[4]. A prototype version of a sharing-preserving variant of the Erlang VM has also been developed and is currently being tested for inclusion into March 2013 Erlang/OTP release R17.

- To improve language-level scalability we have implemented a reliable s_group library. The implementation is validated by giving an operational semantics and validating the semantics against the library using QuickCheck[5]. SD Erlang is open source and available from[6].

- We have produced a deployment and management infrastructure Wombat for exploiting multiple heterogeneous clusters/cloud resources.

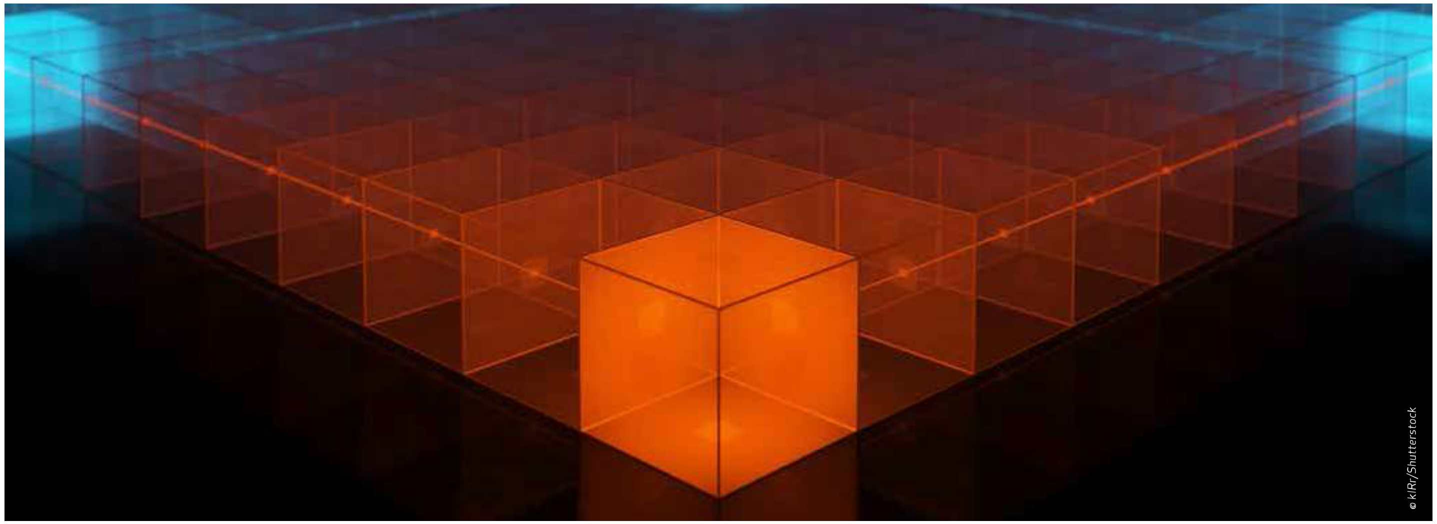
- We have developed, improved, and made open source releases at http://www.release-project.eu/ of five concurrency tools: two online/offline profiling tools, i.e. Persept2[7] and DTrace/SystemTap, a prototype visualisation tool[8], the extended refactoring tool Wrangler[9], and the concurrency error detection tool Concuerror[10].

- As a case study we have adapted Sim-Diasca, 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 standard Erlang OTP library, and performance evaluation of the scalable infrastructure on an IBM Blue Gene/Q.

ACKNOWLEDGEMENTS

We would like to thank all our RELEASE colleagues. This work is supported by the European Union grant RII3-CT-2005-026133 'SCIEnce: 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/G055181/1 'HPC-GAP: High Performance Computational Algebra and Discrete Mathematics'.



GENERAL PROJECT INFORMATION

Project coordinator:
Glasgow University
Contact person: Phil Trinder
School of Computing Science,
University of Glasgow
G12 8QQ Glasgow, United Kingdom
Tel: +44 141 3303627
Fax: +44 141 3304913
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

[1] F. Cesarini and S. Thompson, Erlang Programming: A Concurrent Approach to Software Development, O'Reilly Media Inc., 2009, 1st edition.

[2] "Sim-Diasca," EDF, [Online]. Available: http://researchers.edf.com/software/sim-diasca-80704.html. [Accessed 2014].

[3] "D6.3 Distributed Erlang Component Ontology," Erlang Solutions Ltd, http://www.release-project.eu/, 2013.

[4] "OTP R16B03-1," Ericsson Computer Science Laboratory, [Online]. Available: http://www.erlang.org/download.html. [Accessed 2014].

[5] "QuickCheck," QuviQ, [Online]. Available: http://www.quviq.com/index.html. [Accessed 2014].

[6] "release-project/otp," RELEASE project, [Online]. Available: https://github.com/release-project/otp/tree/dev. [Accessed 2014].

[7] H. Li and S. Thompson, "Multicore Profiling for Erlang Programs Using Percept2," in Twelfth ACM SIGPLAN Erlang Workshop, pp. 33-42. ACM Press, Boston, USA, 2013.

[8] S. Thompson, R. Baker, P. Rodgers and H. Li, "Multi-level Visualization of Concurrent and Distributed Computation in Erlang," in 19th International Conference on Distributed Multimedia Systems (DMS 2013), Brighton, UK, 2013.

[9] Wrangler Refactoring Tool" H. Li, S. Thompson [Online]. Available: http://www.cs.kent.ac.uk/projects/wrangler/Wrangler/Home.html [Accessed 2014]

[10] M. Christakis, A. Gotovos and K. Sagonas, "Systematic Testing for Detecting Concurrency Errors in Erlang Programs," in Sixth IEEE International Conference on Software Testing, Verification and Validation, pp. 154—163. IEEE Computer Society, Luxemburg, 2013.

Natalia Chechina is a postdoctoral research associate at Glasgow University in the RELEASE project.

Phil Trinder is a Professor at Glasgow University, and is a principle investigator of the RELEASE project. He is mainly interested in how to better exploit multi- and many-core machines, Clouds, High Performance Computers (HPC), and other platforms.