Future Generation Computer Systems 53 (2015) 88–89

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Editorial

Special Section on Terascale Computing

High Performance Computing is becoming increasingly relevant for industry and academia. With the current development on the processor market, modern systems quickly grow in size, i.e. in number of cores, but only little in terms of performance, i.e. in actual execution speed. Reason for that is the increasing impact of the memory and communication wall on the one hand and the lack of clock speed increment on a per core level.

Future systems therefore must find means to compensate these deficiencies either through integration of specialized resources and accelerators beyond GPGPUs or through improved parallelization and performance in the software stack.

This special section focuses on different approaches towards dealing with such large-scale future systems that are expected to be highly heterogeneous and most likely specifically designed and configured for a given use case scope, i.e. application domain. While some of the presented approaches focus on the co-design of dedicated accelerators others deliver solutions to tackle the complexity for the software developer by offering new programming model approaches.

Results achieved and presented are partially funded by the Future Emerging Technology (FET) funding programme of the European Commission and are results of the projects Exploiting Dataflow Parallelism in Teradevice Computing (TERAFLUX), European REference TILed architecture Experiment (EURETILE), and Service oriented Operating Systems (S(o)OS).

The first article [1] deals with the challenges of fault tolerance that become more pertinent in large scale systems ahead of us. The paper presents an approach for detecting faults on a system level. The approach is based on a high performance Network Interface Card (NIC), implementing an *N*-dimensional mesh topology and a Service Network. The hierarchical watchdog mechanism realized with this NIC is able to quickly detect faults on each node, as the Host and the high performance NIC guard each other while every node monitors the immediate neighbors in the mesh. The paper describes the implementation of this hard- & software co-design and the approach preventing routed diagnostic messages to affect the system performances.

The second article [2] presents a solution to address the increasingly large synchronization and communication overhead in large scale computing systems. The paper explores alternative execution models to exploit the high parallelism offered by future massive many-core chips. In particular the paper proposes the integration of standard cores with dedicated co-processing units enabling a fine-grain data-flow execution model. The proposed solution covers a programming model as well as the realization of dedicated hardware solutions. The proposed concept is validated with experimental results.

The third paper [3] discusses a point-to-point, low latency, 3D torus Network Controller integrated in an FPGA-based PCIe board relying on a Remote Direct Memory Access (RDMA) communication protocol. RDMA requires the ability to directly access the main memory with no or minimal OS or CPU intervention. A first implementation was realized on a soft-core μC on the FPGA. In a second iteration, an accelerated version on basis of an application-specific processor (ASIP) was designed. The benefit of the approach is demonstrated with benchmark results for Buffer Search and Virtual-to-Physical tasks on the ASIP.

The fourth paper [4] discusses how future operating systems running across many chips with many cores can help dealing with the complexity of future terascale computing systems. In such a highly distributed environment, resource discovery is an important and critical building block. Resource discovery aims to match the application's demands to the existing (distributed) resources, by identifying the most suitable resource configuration at run-time. The main contribution of this paper is the design and evolution of a highly scalable and flexible resource discovery model for such heterogeneous environments. The model is based on self-organizing processing resources in the system according to a hierarchical resource description where each group of resources has a local directory that collects and keeps the information of the underlying resource members (cores) in different layers. Operationally, at each layer, it consists of a peer-to-peer architecture of modules that, by interacting with each other, provide a global view of the resource availability in a large, dynamic and heterogeneous distributed environment.

In the fifth paper [5] a hardware accelerated approach for the programming model OmpSs is discussed. The OmpSs model provides a simple and powerful way to exploit heterogeneity and task parallelism based on runtime data dependency analysis. Such data dependency is exposed to the runtime scheduler using a dedicated annotation model. The paper presents Picos, an implementation of the Task Superscalar (TSS) architecture that provides hardware support to the OmpSs programming model. It discusses the Hardware Design and the improvements in terms of achievable latencies. The paper compares Picos and the software based Nanos++ runtime performance scalability with a set of real benchmarks for the different approaches.

The guest editors would like to show their appreciation to all the authors and reviewers for their constructive and valuable contributions to this special section. We would also like to extend

http://dx.doi.org/10.1016/j.future.2015.07.015
0167-739X/

our thanks to Peter Sloot, Editor-in-Chief, for his invaluable help and productive advice in preparing this special section.

References

[1] Roberto Ammendola, Andrea Biagioni, Ottorino Frezza, Francesca Lo Cicero, Alessandro Lonardo, Pier Stanislao Paolucci, Davide Rossetti, Francesco Simula, Laura Tosoratto, Piero Vicini, A hierarchical watchdog mechanism for systemic fault awareness on distributed systems, Future Generation Computer Systems 53 (2015) 90–99.
[2] Roberto Giorgi, Alberto Scionti, A scalable thread scheduling co-processor based on data-flow principles, Future Generation Computer Systems 53 (2015) 100–108.
[3] Roberto Ammendola, Andrea Biagioni, Ottorino Frezza, Werner Geurts, Gert Goossens, Francesca Lo Cicero, Alessandro Lonardo, Pier Stanislao Paolucci, Davide Rossetti, Francesco Simula, Laura Tosoratto, Piero Vicini, ASIP acceleration for virtual-to-physical address translation on RDMA-enabled FPGA-based network interfaces, Future Generation Computer Systems 53 (2015) 109–118.
[4] Javad Zarrin, Rui L. Aguiar, João P. Barraca, Dynamic Scalable and flexible resource discovery for large-dimension many-core systems, Future Generation Computer Systems 53 (2015) 119–129.
[5] Fahimeh Yazdanpanah, Carlos Álvarez, Daniel Jiménez-González, Rosa M. Badia, Mateo Valero, Picos: A hardware runtime architecture support for OmpSs, Future Generation Computer Systems 53 (2015) 130–139.

Stefan Wesner
Lutz Schubert
Universität Ulm, Germany
E-mail addresses: stefan.wesner@uni-ulm.de (S. Wesner), lutz.schubert@uni-ulm.de (L. Schubert).

Rosa M. Badia
BSC, Spain
E-mail address: rosa.m.badia@bsc.es.

Antonio Rubio
UPC, United States
E-mail address: antonio.rubio@upc.edu.

Pier Paolucci
INFN, Italy
E-mail address: pier.paolucci@roma1.infn.it.

Roberto Giorgi
UNISI, Italy
E-mail address: giorgi@dii.unisi.it.