SDPS Journal http://www.fpl.uni-kl.de/SDPS/SDPSjournal.html

Transactions of the SDPS:
Integrated Design & Process Science
Transdisciplinary International Journal

Special Issue on Future Computer Systems:

Reconfigurable Computing reinvents the Computing Discipline

Preface. *Because of high energy consumption our computer-based infrastructure may become unaffordable - without reinventing the entire computing discipline, also due to cope with the manycore programming crisis. We have to recognize essential facts, trends, and a roadmap to by-pass this crisis and to reach new horizons.*

Introduction

The von Neumann syndrome. Still main focus of CS education, the *von Neumann (vN)* model lost its strong dominance more than a decade ago: *its principles are fundamentally wrong, since data processing targets data streams - not instruction streams*. The vN methodology has been criticized for being tremendously overhead-prone already 3 decades ago. John Backus has asked in 1978: "Can programming be liberated from the von Neumann style?", and in 1981: "Is computer science based on the wrong fundamental concept of 'program'?". 1000 processors running in parallel means that 1000 instruction streams with all their typical overhead phenomena yield a drastic programmer productivity decline. Burton J. Smith in his keynote at ISC Dresden (2007) summarizes: „In practice we are limited to a few instructions per clock cycle." Traditional software engineering problems are now topped by *the manycore programming crisis*. In industry the vN model has been replaced by a cooperation of vN CPU and non-VN accelerators. To-day, most MIPS equivalents are running on FPGAs (Field-Programmable Gate Arrays: the fastest growing segment of the semiconductor market), where the microprocessor has become the tail wagging the dog and the basic accelerator model is *data-stream-based* - not instruction-stream-based. To make it much easier for developers to implement high performance systems, we urgently must reinvent not only computing but also the computing profession. instead of physical limits, *fundamental misconceptions of algorithmic complexity theory* limit the progress and will necessitate new breakthroughs. We've to completely re-think basic assumptions behind computing. Everything we know is wrong. This requires a paradigm revision for execution and programming models.

The most disruptive revolution since the mainframe: it's *Reconfigurable Computing (RC)*, still mostly FPGA-based. Its pervasiveness is obvious. It comes with a second machine paradigm: which we may call the *anti-machine*, counterpart of vN. Meanwhile RC has become mainstream, not only in embedded systems. Software to configware migration promises tremendous speedups and energy savings - at much lower cost. This indicates, that our common models and implementation principles are fundamentally wrong. Almost 200 international conference series deal with RC and its applications. Meanwhile *field-programmability is a must*. Since 2006, RC is also a hot spot in supercomputing.

Transdisciplinary issues. Since 1980 mathematicians liked systolic array schemes to solve complex algebraic problems and came up with an excellent definition of *data stream*. Their reply to organizing *data streams* to run systolic arrays was: „this is not our job. [the job of hardware people.]", so that inventing the non-vN antimachine paradigm has been missed. Their synthesis methods to derive systolic architectures from math formula has been, of course, algebraic: supporting only applications with strictly *regular data dependencies* - a far-ranging limitation. The „transdisciplinary" break-through away from this tunnel view happened elsewhere - more than a decade later.

The hardware / software chasm. Computer Science is even split up into subdisciplines (hardware, software and theory) causing trans-subdisciplinary mapping problems, also one of the reasons of the manycore programming crisis. This is astonishing since such mapping models have been published in the 50ies (with microprogramming), in the 60ies („Register Transfer Modules" and „Macromodular Computing Systems") and in the 70ies with introducing hardware description languages using software methodologies.

The Education Wall. Mapping an application from software to configware means *mapping from time to space* - hated by people with a software-only background. This backlog is due to the *tunnel view of the software-only mind set*. This essential problem is ignored by our curriculum recommendation groups like the joint ACM/IEEE-CS task force, still hopelessly reluctant to discuss RC issues. Such a software-only mind set misses to hit the IT job market.

A new paradigm cannot be avoided. Most experts in HPC are reluctant to go for a paradigm extension. We need a bridge strategy by developing advanced tools for training the software community. This makes sense, because of the manycore programming crisis is running in parallel with the break-through of RC. Because we cannot afford to discard the von Neumann model, *we have to support a twin paradigm approach:* a well and clearly organized dual model supporting both, von Neumann and the antimachine paradigm.

Understandable modelling scheme needed. A global system view is required for grasping the principles and essential issues of heterogeneous twin paradigm systems, in all levels of education. *We need an intuitive*

terminology and an understandable common modeling scheme - a style of schematics with a clear distinction between von Neumann subsystems and antimachine blocks.

Topics:

Topics of this special issue include, but are not limited to, the following:
Supercomputing Architecture History - e. g. lessons learned from Top500 history
Reconfigurable Supercomputing
The manycore programming crisis
Heterogeneous manycore system architectures
Programming heterogeneous manycore systems
Reconfigurable Computing and saving energy
Reconfigurable Computing education
Reconfigurable Computing as part of CS and IT curricula
Configware tools and twin paradigm tools acceptable by software people
Dynamically reconfigurable systems
Configware operating systems
Twin paradigm operating systems
Funding programs in Europe, America and elsewhere

Motivation. To get some more ideas about the motivation behind this special issue, you may also have a look into the paper " The von Neumann Syndrome":
http://www.fpl.uni-kl.de/staff/hartenstein/Hartenstein-Delft-Sep2007.pdf

Submission Procedure:

All the papers should be full journal length versions between 10 and 20 pages long and follow the guidelines set out by the template: see http://hartenstein.de/SDPS/Template-SDPS-journal-07.doc
All the papers of this issue are invited papers.

Important Dates:

The deadline of the first version of the manuscript is December 15, 2007. It will be reviewed and you might be asked for modifications. The final full manuscript due: February 1, 2008 by e-mail to reiner@hartenstein.de Publication of the special issue: toward early March in 2008.

Guest editor:

Dr.-Ing. Reiner Hartenstein, Professor, TU Kaiserslautern
URL: http://hartestein.de/index.html
E-mail: reiner@hartenstein.de
Tel: +49-7251 300 575

Editor-in-Chief:

Azad M. Madni, Intelligent Systems Technology Inc.
3250 Ocean Park Blvd., Suite 100
Santa Monica, California 90405, USA
E-mail: amadni@intelsystech.com

Dear Author,

Please, stress trans-disciplinary aspects.

Note the wide variety of back-grounds of the readership:

maybe, not specialized in HPC nor Reconfigurable Computing.

Be an evangelist!

all URLs: see http://www.fpl.uni-kl.de/SDPS/