TERA^FLUX Exploiting Dataflow Parallelism in Teradevice Computing

Future Teradevice systems will expose a large amount of parallelism (1000+ cores) that cannot be exploited efficiently by current applications and programming models. The aim of this project is to propose a complete solution that is able to harness the large-scale parallelism in an efficient way. The main objectives of the project are the programming model, compiler analysis, and a scalable, reliable, architecture based mostly on commodity components. Data-flow principles are exploited at all levels as to overcome the current limitations.

[KEYWORDS: Multicores, Multithreaded Execution, Programming Models, Reliability]

At A Glance:

TERA^FLUX

www.teraflux.eu

Project Coordinator

Università degli Studi di Siena
Prof. Roberto Giorgi
giorgi@dii.unisi.it

Partners:

- Università degli Studi di Siena (Italy)
- Barcelona Supercomputing Center (Spain)
- CAPS (France)
- Hewlett Packard Labs (Spain)
- INRIA (France)
- MICROSOFT R&D (Israel)
- THALES (France)
- University of Augsburg (Germany)
- University of Cyprus (Cyprus)
- University of Manchester (UK)

Duration: *48 months*
Start Date: 2010.01.01
Total Cost: € 7.543.714
EC Contribution: € 5.700.000
Grant Agreement Number: 249013

Main Objectives

The technology trends show that by 2020 chips will accommodate teradevice systems or 1000+ of cores. The success of these future architectures depends on addressing important challenges such as programming applications to use such large-scale systems, developing compiler analysis and optimizations required for the generation of code, developing appropriate execution models, which can take can on both performance and reliability. In addition, the architecture cannot be reinvented from scratch each time. Therefore, it should be composed of commodity modules such as the execution cores and the interconnection network.

TERA^FLUX focus is on developing an infrastructure for programming future multicore systems

One of the key aspects of this project is the proposal of a new programming and execution model based on data-flow instead of traditional control-flow. Data-flow is known to overcome the limitations of the traditional control-flow model by exploring the maximum parallelism and reducing the synchronization overhead. Although its benefits are well known and have been presented a long time ago, this model has not yet been fully exploited for commercial systems.

This project represents a unique opportunity to integrate complementary essential aspects from the applications through the whole tool chain, encompassing reliability, an appropriate architecture and resource management (that accounts for power, temperature, faults) and to test the research ideas in a simulated Teradevice system.
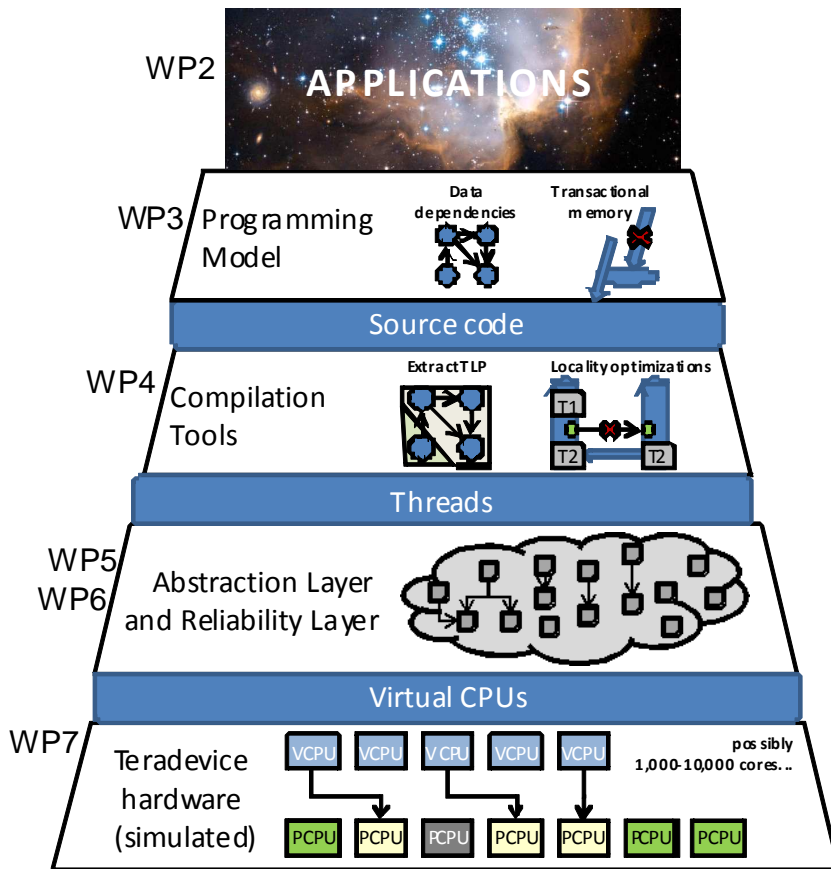
Expected Impact

We expect to develop a coarse grain dataflow model (or fine grain multithreaded model) that will encompass fine grain transactional isolation, scalable to many cores and distributed memory, with built-in application-unaware resilience, with novel hardware support structures as needed.

Moreover we will provide an open evaluation platform based on an x86 simulator based on COTSon by TERAFLUX partner HPLabs (http://cotson.sourceforge.net/) that enables leveraging the large software body out there (OS, middleware, libraries, applications).

Technical Approach & Key Issues

WP2 — **APPLICATIONS**

WP3 — Programming Model — Data dependencies — Transactional memory

Source code

WP4 — Compilation Tools — Extract TLP — Locality optimizations — T1 T2 T2

Threads

WP5 WP6 — Abstraction Layer and Reliability Layer

Virtual CPUs

WP7 — Teradevice hardware (simulated) — VCPU VCPU VCPU VCPU VCPU — possibly 1,000-10,000 cores… — PCPU PCPU PCPU PCPU PCPU PCPU PCPU

- Applications are becoming more and more complex, demand for higher degrees of accuracy, and process larger amounts of data. As such, in this project we will look at current and emerging demanding applications to be executed on the Teradevice systems. Their evaluation will allow us to study the limits of such large-scale future systems.

- The programming challenge is how to make the parallel resources easily available to the programmers for such large-scale systems. In TERA^{F}LUX, we propose a two-level parallel programming (efficient programming + performance programming) approach. In TERA^{F}LUX, we explore programming models that combine the benefits of data-flow with transactional memory principles.

- Compiler optimizations are required to coarsen the grain of concurrency, allocate memory statically, convert streams into shared memory buffers, overlap communications and computations, instrument the code with resource and power management probes/actions.

- In modern architectures reliability is a major aspect for system designers and users. New process technology exposes us to new challenges such as aging, process variability, soft errors. Defects and errors will dramatically increase in the near future. Therefore, building a reliable system out of unreliable components becomes a major problem for future systems and needs special attention

- Simple technology improvements will allow for the scaling of the current designs to 1000+ of cores on the same chip. One concern is to keep the system within the required power budget. In order to satisfy that goal and at the same time provide a large degree of parallelism, the TERA^{F}LUX architecture will be composed of heterogeneous multi-cores supporting the same instruction set. Simpler more power efficient cores will provide the parallelism while more complex and less power efficient cores will be used to execute codes requiring Instruction Level Parallelism (ILP).

- An existing infrastructure for full system simulation (COTSon by HP labs) current supported as MIT license has been chosen as the simulation infrastructure able to provide fast and accurate evaluation of current and future computing systems, covering the full software stack and complete hardware models.