Systems & Programming Models at the High Performance Computing Center Stuttgart

Rainer Keller, HLRS
Context: Organizational

- HLRS is one of the three national supercomputing centers in Germany.
- The national supercomputing centers are working together in the Gauss Centre for Supercomputing GCS.
- GCS is the means to contribute to the Partnership for Advanced Computing in Europe (PRACE).
- All centers work within PRACE towards a European HPC Infrastructure and perform research with all PRACE partners towards Exascale computing.
- Additionally HLRS is responsible within PRACE and GCS for the support of the engineering community and the definition of the industrial offer.
Context: Main User’s Research Projects

- Aeroacoustics
- Aerodynamics
- Astrophysics
- Bioinformatics
- Combustion
- Fluid-Structure Interaction
- Helicopter Aerodynamics
- Meteorology
- Medical Imaging
- Nanotechnology
- Solid State Physics
- Turbo Machinery
- Turbulence Phenomena
Context: Systems in Stuttgart

- Large variety of HPC systems offered:

Clusters, e.g.:
- Laki 62 TF
- BW-Grid 14 TF

New architectures, e.g.:
- Cluster of Cell (2008)
- MD Grape (2007)
Context: The petaGCS Project (Phase1)

- The petaGCS project is a BMBF funded project covering the national share for investment and operation of national supercomputing in Germany
  - Covers currently Phase1 of all GCS centers
  - Next phases will be covered in a similar way
  - 50% co-funding is provided by the regional governments

For HLRS: the Ministry of Science, Research and the Arts Baden-Württemberg
Phase 1 Step 1: Hermit 1
3rd PRACE TIER-0 System
Programming models research

- Projects at HLRS regarding programming models:

  **User centric:**
  - PRACE
  - DEISA
  - HPC-Europa2

  **Tool centric:**
  - TEXT
  - CRESTA
  - Open MPI

  **Application centric:**
  - IMEMO
  - SFB716 D.2
  - APOS

  **Accelerator- centric:**
  - H4H
  - Insilico project
  - HMI-Tec

  **Programming model:**
  - LarKC
  - PRACE-1IP
  - ECOUSS
• Application Performance Optimisation and Scalability
• EU-funded STREP-Project in FP7-2011-EU-Russia
• Just started, runs for 24 months

• EU Partners:
  – EPCC, UK (Coordinator)
  – CAPS entreprise, France
  – Uniwersytet Warszawski, Poland
  – TOTAL, France
  – HLRS, Germany

• Russian Partners:
  – Kurchatov Institute, Moscow
  – Ugra Research Insitute, Khanty-Mansiysk
  – Institute for Mathematical Modelling, Moscow
  – Moscow Physics Technical Institute, Moscow

Contact: Colin Glass
• Porting of Codes from different scientific domains:
  – CFD, Magneto-hydrodynamics, molecular dynamics (work with open source GS2, Open Foam, LS1 codes)
  – Onto different target architectures (Multi-core, GPU)

• EU Contribution:
  WP1: Identification of codes
  WP2: Porting to Multicore
  WP3: Porting to GPGPU
  WP4: Prototype Tools
  WP5: Dissemination
GPU: Industrial Collaboration: HMI-Tec

• Parallelize KI Neuro-Sorter using CUDA

Speedup: Training phase
Pattern:
- 3766 words
- 3766 input neurons
- Vary # inner neurons

Data: Zaheer Ahmed
Programming Model: ECOUSS

- “Effiziente und offene Compiler Umgebung für Semantisch annotierte parallele Simulationen”
- BMBF-funded project within the HPC-Initiative
- Partners:
  - HLRS (Coordinator: Stefan Wesner)
  - Universität des Saarlandes, Intel Visual Computing Insittu
  - Deutsches Forschungszentrum für Künstliche Intelligenz
  - Universität Karlsruhe
  - Cray Computer Deutschland

Top-down: Use code annotations, to describe programmers intent:
#pragma ivdep, #pragma no_side_effect, attribute(hot)

Bottom-up: Improve hardware description for more efficient mapping:
number of cores per socket, memory-channels
Tool (and Application) centric: TEXT 1/3

- Towards EXascale ApplicatTions (TEXT)
- EU-funded CP & CSA in FP7-Infrastructures-2010-2
- Partners:
  - BSC, Spain
  - HLRS, Germany
  - FZJ, Germany
  - EPCC, UK
  - FORTH, Greece
  - University of Manchester, UK
  - Universite de Pau et des Pays de L’Adour, France
  - Universitat Jaume I de Castellon, Spain
  - IBM Research Zürich, Switzerland

Centered around the StarSS programming model by BSC:

```c
#pragma css task input(v1, v2, len) output(v3)
void vadd (float *v1, float *v2, float *v3, int len)
```
Tool (and Application) centric: TEXT 2/3

• Parallelization using SmpSS on the:
  – BEST / LBC Lattice Boltzmann codes
    Jose Gracia
  – LS1-Mardyn MD code
    Christoph Niethammer

• Develop a small testsuite to find implementation bugs and tools capabilities (valgrind-like errors)

• Develop debugger & techniques of threaded debugging
• Ease thread programming with graphical debugger:

Contact: Steffen Brinkmann
Tools and Parallel Programming support

- Cray XE6: a nice machine for PGAS-like languages
- The main parallel programming model: MPI and Hybrid MPI+Some Thread-Parallelism

- HLRS & Cray will provide software stack:
  - Set of compilers: Cray, PGI, GNU & Intel
  - Eclipse
  - Allinea DDT
  - CrayPat & Apprentice2
  - Vampir / VampirServer
  - Scalasca
  - Roguewave Threadspotter
• Parallel debugger (up 200k cores @ ORNL/ Jaguar)
Performance Analysis: Vampir

Process 0, Values of Counter "PAPI_FP_OPS" over Time

0 M

Process 0, Values of Counter "PAPI_FPU_IDL" over Time

60 G

30 G

78.925 s
Performance Analysis: ThreadSpotter

- Analysis of RogueWave Threadspotter
- Memory-access pattern analysis for cache-optimization
Thank You very much!

Any Questions?