Performance Tools for Petascale Systems

The BMBF project SILC

Zellescher Weg 12
Willers-Bau A114
Tel. +49 351 - 463 - 38323

Andreas Knüpfert (andreas.knuefer@tu-dresden.de)
Overview

Performance tools for HPC
- Periscope
- Scalasca
- Vampir Tools

Project goals
- The tool's perspective
- The user's perspective

Status and outlook
Overview

Performance tools for HPC
- Periscope
- Scalasca
- Vampir Tools

Project goals
- The tool's perspective
- The user's perspective

Status and outlook
Why HPC Performance Tools matter

- HPC performance analysis and optimization is hard
- So is the scalability challenge
- Need to “look inside parallel application runs“

Use appropriate tools!

- No do-it-yourself solutions with MPI_WTime, printf, and write

There are specific support tools:
- For MPI, OpenMP, Pthreads, and hybrid applications
- Experimental support for Cell, OpenCL, CUDA
- Including timer management, performance counters, I/O monitoring, etc.
- For all relevant HPC platforms
Periscope

- Automatic online performance analysis
  - Start with performance flaw hypotheses
  - Investigate hypotheses with measurement during one/few iterations
  - Revise hypotheses after each step
- For parallel applications
- Local and global evaluation
- Run for successive iterations, restart application if needed
- Eclipse integration
Scalasca

- Automatic post-mortem performance analysis
  - Record per-process event traces first
  - Automatically analyse for known performance problems then
- Analysis on original processes at original scale
- Communication replay to exchange performance/timing data
- Finally summarize global results, map to source code

- Graphical interface CUBE
- Scales to $>290,000$ cores
- Developed for more than 10 years
VampirTrace and Vampir GUI

- Code instrumentation and event tracing with VampirTrace
  - Also official part of Open MPI

- Event trace visualization with Vampir

- Interactive analysis
  - Parallel evaluation
  - Distributed trace data

- Scalable to
  - Ten thousands of processes/threads
  - Hundreds of gigabytes of trace data

- Since more than 15 years
Overview

Performance tools for HPC
- Periscope
- Scalasca
- Vampir Tools

Project goals
- The tool's perspective
- The user's perspective

Status and outlook
Project Goals

Measurement Infrastructure
- unification, support for long run times, enhance scalability, parallelization of tool’s subtasks, common data formats, online interface, minimize overhead, support for OpenMP 3.0, wait time analysis

Analysis
- adapt analysis tools to new measurement infrastructure, enhance analysis functionality

Software Quality
- tests, documentation, code maintenance

Evaluation and use cases
- qualitative and quantitative evaluation

Dissemination
Tool's Perspective: Common Functionality

- Code Instrumentation
  - Compiler wrappers, instrumentation API, helper tools, ...

- Run-time recording
  - Timer management,
  - Performance counter support, ...

- Handling of data and meta data

- Data formats
  - Event tracing and profiling
  - Data interoperability

- Platform portability and production quality

- Extensive feature set and overhead reduction
Common Functionality

- Currently, similar functionality with different infrastructure

- Periscope
- Scalasca
- Vampir
- TAU

Application Code
Common Functionality

Goal: common unified infrastructure

Periscope  Scalasca  Vampir  TAU

Application Code
User's Perspective

- Different complimentary tools
- Learning curve for tools
- User friendly but not trivial

- Need to re-compile or re-link once or multiple times
- Investigate same experiment runs with several tools
User's Wishes

- Common user perspective
- Same command line tools
- Same configuration options

scalasca -instrument mpicc hello_world.c

mpicc-vt hello_world.c

icc -finstrument-function
hello_world.c -lxyz -lmpi
Overview

- Performance tools for HPC
  - Periscope
  - Scalasca
  - Vampir Tools

- Project goals
  - The tool's perspective
  - The user's perspective

- Status and outlook
**Project Status**

**Status at middle of project time**

- Thorough specification of requirements and design
- Combination/unification of existing functionalities
- Three main infrastructure components
  - Instrumentation and data collection layer (Adapters)
  - Run-time measurement and buffering layer (Core)
  - Open Trace Format 2 (OTF2)
- First working version

**Dissemination**

- Series of VI-HPS tuning workshops
- Announcement at SC BOF in New Orleans in November
SILC Partners

- Forschungszentrum Jülich, Jülich Supercomputing Centre (JSC)
  - Scalasca tools for event tracing and automatic analysis
- Gesellschaft für numerische Simulation mbH (GNS)
  - KMU, vendor for industrial simulation applications
  - Case studies with real world codes and scenarios
- Rheinisch-Westfälische Technische Hochschule Aachen (RWTH)
  - Member of OpenMP committee
- Technische Universität München,
  Lehrstuhl für Rechnertechnik und Rechnerorganisation
  - Periscope tools for automatic online performance analysis
SILC Partners

SILC Partners contd.

- Technische Universität Dresden, Zentrum für Informationsdienste und Hochleistungsrechnen (ZIH), SILC coordinator
  - Vampir tools for event tracing and interactive visualization

Associated partner

- Gesellschaft für Wissens- und Technologietransfer der TU Dresden mbH (GWT)
  - KMU, Vendor for industry simulation software

New associated partner

- University of Oregon
  - TAU tools for parallel profiling and event tracing
Conclusions

The SILC project:
- Performance tools integration
- Feature enhancements
- Scalability improvements
- Part of long term development plus user support
- Part of traditional strength of HPC research in Germany

See also:
- Virtual Institute High Productivity Supercomputing at http://www.vi-hps.org/
- SILC project at http://www.vi-hps.org/projects/silc/

Thanks to BMBF and DLR!