Performance Analysis with Vampir

Ronny Tschüter, and Tobias Hilbrich
Technische Universität Dresden
Part I: Welcome to the Vampir Tool Suite
   – Mission
   – Event Trace Visualization
   – Vampir & VampirServer
   – The Vampir Displays

Part II: Vampir Hands On
   – Visualizing and analyzing NPB-MZ-MPI / BT

Part III: Summary and Conclusion
Mission

• Visualization of dynamics of complex parallel processes
• Requires two components
  – Monitor/Collector (Score-P)
  – Charts/Browser (Vampir)

Typical questions that Vampir helps to answer:
  – What happens in my application execution during a given time in a given process or thread?
  – How do the communication patterns of my application execute on a real system?
  – Are there any imbalances in computation, I/O or memory usage and how do they affect the parallel execution of my application?
Event Trace Visualization with Vampir

- Alternative and supplement to automatic analysis
- Show dynamic run-time behavior graphically at any level of detail
- Provide statistics and performance metrics

Timeline charts
- Show application activities and communication along a time axis

Summary charts
- Provide quantitative results for the currently selected time interval
Vampir – Visualization Modes (1)

• Directly on front end or local machine

% export MODULEPATH=/zhome/academic/HLRS/xhp/xhprt/privatemodules:$MODULEPATH
% module load vampir
% vampir
Vampir – Visualization Modes (2)

- On local machine with remote VampirServer

```bash
% module load vampirserver
% vampirserver start
```

```bash
% module load vampir
% vampir
```

![Diagram of VampirServer, Score-P, Many-Core Program, Large Trace File (OTF2), LAN/WAN, MPI parallel application]
1. Instrument your application with Score-P

2. Run your application with an appropriate test set

3. Analyze your trace file with Vampir
   - Small trace files can be analyzed on your local workstation
     1. Start your local Vampir
     2. Load trace file from your local disk
   - Large trace files should be stored on the HPC file system
     1. Start VampirServer on your HPC system
     2. Start your local Vampir
     3. Connect local Vampir with the VampirServer on the HPC system
     4. Load trace file from the HPC file system
The main displays of Vampir

**Timeline Charts:**
- Master Timeline
- Process Timeline
- Counter Data Timeline
- Performance Radar

**Summary Charts:**
- Function Summary
- Message Summary
- Process Summary
- Communication Matrix View
Vampir hands-on

Visualizing and analyzing NPB-MZ-MPI / BT
Help! Where is my trace file?

- If you followed the Score-P hands-on up to the trace experiment, yours is in:

  % ls $SCRATCH/NPB3.3-MZ-MPI/bin.scorep/scorep_trace
  profile.cubex  scorep.cfg  traces/  traces.def  traces.otf2

  That’s what you will open with Vampir

- If you removed the trace or did not follow to that point, copy a prepared trace

  % cd $SCRATCH/NPB3.3-MZ-MPI/bin.scorep
  % cp /zhome/academic/HLRS/xhp/xhprt/scorep_trace ./
Option A: Starting Vampir on Hornet

• Load modules

  % export MODULEPATH=/zhome/academic/HLRS/xhp/xhprt/privatemodules:$MODULEPATH
  % module load vampir

• Start Vampir on the frontend (small traces only!)

  % vampir

  • Use the following to connect to Hornet:
    $ ssh -XC ...
  • This enables X11 forwarding AND compression, the latter is very helpful for a good usage experience
Vampir: Visualization of the NPB-MZ-MPI / BT trace

- **Master Timeline**
- **Navigation Toolbar**
- **Function Summary**
- **Function Legend**
Master Timeline

Detailed information about functions, communication and synchronization events for collection of processes.
Process Timeline

Detailed information about different levels of function calls in a stacked bar chart for an individual process.
Typical program phases

- **Initialisation Phase**
- **Computation Phase**
Vampir: Visualization of the NPB-MZ-MPI / BT trace

Counter Data Timeline

Detailed counter information over time for an individual process.
Performance Radar

Detailed counter information over time for a collection of processes.
Zoom in: Initialisation Phase

Context View: Detailed information about function “initialize_”.
Execution of function “initialize_” results in higher page fault rates.
Computation Phase

Computation phase results in higher floating point operations.
Zoom in: Computation Phase

MPI communication results in lower floating point operations.
Zoom in: Finalisation Phase

“Early reduce” bottleneck.
Function Summary:
Overview of the accumulated information across all functions and for a collection of processes.

Process Summary:
Overview of the accumulated information across all functions and for every process independently.
Vampir: Visualization of the NPB-MZ-MPI / BT trace

Process Summary

Find groups of similar processes and threads by using summarized function information.
Option B: Larger Traces – Overview

Step 1: Start VampirServer on Hornet backend

Step 2: Portforwarding to Hornet

Step 3: GUI on your local machine

For reference
Option B: Step 1, Starting Vampirserver on Hornet

- Load modules

```bash
% export MODULEPATH=
   /zhome/academic/HLRS/xhp/xhprt/privatemodules:$MODULEPATH
% module load vampirserver
```

- Start a vampirserver job on the backend

```bash
% vampirserver start
Launching VampirServer...
Submitting PBS batch job (this might take a while)...
Batch job is submitted - Job ID: 168008.hornet-batch.hww.de
VampirServer 8.4.1 (r9456)
Licensed to VI-HPS Tools Workshop 02/2015
Running 16 analysis processes... (abort with vampirserver stop 23731)
VampirServer <23731> listens on: mom12:30081
```

Remember these coordinates we will need them in a second
Option B: Step 2, Port Forwarding to Hornet

- Write down the host on which the server runs

```bash
% vampirserver start
Launching VampirServer...
Submitting PBS batch job (this might take a while)...
Batch job is submitted - Job ID: 168008.hornet-batch.hww.de
VampirServer 8.4.1 (r9456)
Licensed to VI-HPS Tools Workshop 02/2015
Running 16 analysis processes... (abort with vampirserver stop 23731)
VampirServer <23731> listens on: mom12:30081
```

- Establish Port Forwarding from your local machine to Hornet

```bash
% ssh \
-L 30000:mon12:30081 \
<user>@hornet.hww.de
```
Option B: Step 2, Install Vampir client locally

- Start a new shell on your laptop
- Copy the appropriate Vampir package from Hornet to your laptop

```bash
% ls /zhome/academic/HLRS/xhp/xhprt/tutorial
vampir-linux-ia32.zip   vampir-win-x64.zip   vampir-linux-x86_64.zip
vampir-win-x86.zip      vampir-mac.zip
% scp <user>@hornet.hww.de:/zhome/academic/HLRS/xhp/xhprt/tutorial/ <version> ./
```

- Extract the archive and install
  (example with linux-x86-64)

```bash
% unzip vampir-linux-x86_64.zip
% cd vampir-linux-x86_64
% ls
vampir-8.4.1-linux-x86_64-setup.bin
vampir-remote.license
```

- Install and start the Vampir GUI
Option B: Step 2, Install Vampir client locally

- Select the `vampir-remote.licence` from the archive file when asked
Option B: Step 2, Install Vampir client locally

Use the “Open Other” option
Option B: Step 2, Install Vampir client locally

Select “Remote File”
Option B: Step 2, Install Vampir client locally

- Server is "localhost"
- Port is "30000"
- Connection type "Socket"
Option B: Step 2, Install Vampir client locally
Summary and Conclusion
Summary

• Vampir & VampirServer
  – Interactive trace visualization and analysis
  – Intuitive browsing and zooming
  – Scalable to large trace data sizes (20 TiByte)
  – Scalable to high parallelism (200,000 processes)

• Vampir for Linux, Windows, and Mac OS X

• **Note:** Vampir does neither solve your problems automatically, nor point you directly at them. It does, however, give you FULL insight into the execution of your application.
Conclusion

• Performance analysis very important in HPC

• Use performance analysis tools for profiling and tracing

• Do not spend effort in DIY solutions, e.g. like printf-debugging

• Use tracing tools with some precautions
  – Overhead
  – Data volume

• Let us know about problems and about feature wishes

• vampirsupport@zih.tu-dresden.de
Vampir is available at http://www.vampir.eu,
get support via vampirsupport@zih.tu-dresden.de