Performance Analysis with Vampir

Bert Wesarg, Ronny Tschüter, Andreas Knüpfer
ZIH, Technische Universität Dresden
Outline

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

% vampir
Vampir – Visualization Modes (2)

- On local machine with remote VampirServer

```bash
% vampirserver start -n 12
```

```bash
% vampir
```

**VampirServer**

**Score-P**

**Many-Core Program**

**Trace File (OTF2)**

**LAN/WAN**

**Large Trace File (stays on remote machine)**

**MPI parallel application**

Many-Core Program

Large Trace File (stays on remote machine)
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
Starting Vampir Server on JUQUEEN

• Load modules

```bash
% module load UNITE
UNITE loaded

% module load vampirserver/8.1-be
vampirserver/8.1-be loaded
```

• Start a vampirserver job on the compute nodes

```bash
% vampirserver start -n 127 -t 600 -- --reservation=$LL_RES_ID
Launching VampirServer...
Submitting batch job (this might take a while)...
llsubmit: Processed command file through Submit Filter: "/bgdata/…
```
Installing Vampir Client on your Laptop

- Start a new shell on your laptop
- Copy the vampir-remote.zip package from JUQUEEN to your laptop

```
% scp juqueen.fz-juelich.de:~train001/vampir-remote.zip .
```

- Extract the archive and choose an appropriate package for your platform

```
% unzip vampir-remote.zip
% cd vampir-remote
% ls
vampir-8.1.0-remote-linux-ia32-setup.bin
vampir-8.1.0-remote-linux-x86_64-setup.bin
Vampir-8.1.0-remote-mac.dmg
Vampir-8.1.0-Remote-x64-setup.exe
Vampir-8.1.0-Remote-x86-setup.exe
vampir-remote.license
```

- Start the Vampir GUI
Selecting the Licence File

• Select the `vampir-remote.licence` from the archive file when asked
Vampir Welcome Dialog
Vampir Open Remote Trace Dialog

Server:
- Default
- Description: Default
- Server: localhost
- Port: 30000
- Authentication: None
- Connection type: Socket
- More Options

File Help

[Image of the Vampir Open Remote Trace Dialog window with fields for description, server, port, authentication, and connection type, along with 'Connect' and 'Cancel' buttons.]
Establish Port Forwarding to the Vampir Server

- Write down the host on which the server runs

  ```bash
  % vampirserver start -n 127 -t 600 -- --reservation=$LL_RES_ID
  Launching VampirServer...
  Submitting batch job (this might take a while)...
  llsubmit: Processed command file through Submit Filter: "/bgdata/...
  VampirServer 8.1.0 (r8451)
  Licensed to JSC
  Running 31 analysis processes... (abort with vampirserver stop 743)
  VampirServer <743> listens on: R63-ID-J02.zam.kfa-juelich.de:30003
  ```

- Establish Port Forwarding from your local machine to JUQUEEN

  ```bash
  % ssh \
  -L 30000:R63-ID-J02.zam.kfa-juelich.de:30003 \ 
  juqueen.fz-juelich.de
  ```
Select trace to open from remote Server
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
Detailed counter information over time for an individual process.
Vampir: Visualization of the NPB-MZ-MPI / BT trace

Performance Radar

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

Context View: Detailed information about function “initialize_”.

Vampir: Visualization of the NPB-MZ-MPI / BT trace
Feature: Find Function

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.
Process Summary

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.
Summary and Conclusion
Summary

- **Vampir & VampirServer**
  - Interactive trace visualization and analysis
  - Intuitive browsing and zooming
  - Scalable to large trace data sizes (20 TByte)
  - Scalable to high parallelism (200000 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
Staff at ZIH - TU Dresden:

Ronny Brendel, Holger Brunst, Jens Doleschal, Ronald Geisler, Daniel Hackenberg, Michael Heyde, Matthias Jurenz, Michael Kluge, Andreas Knüpfer, Matthias Lieber, Holger Mickler, Hartmut Mix, Matthias Weber, Bert Wesarg, Frank Winkler, Matthias Müller, Wolfgang E. Nagel