

understanding the perf file format

Urs Fässler

CERN Openlab

01.09.2011

Problem

- processors does not get faster
- more into parallel computing
- software has to be improved

Problem

- processors does not get faster
- more into parallel computing
- software has to be improved
- performance monitoring to find bottlenecks
- performance monitoring system of Linux is perf
- understanding monitoring system → extract relevant information

Process (1)

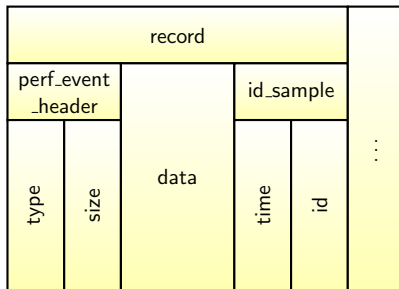
- sparse online resources
- perf report reads the data file

Process (1)

- sparse online resources
- perf report reads the data file
 - stepping through code with debugger
 - static analyze code
 - analyzing data structures

Process (1)

- sparse online resources
 - perf report reads the data file
 - stepping through code with debugger
 - static analyze code
 - analyzing data structures
- ⇒ Pitfall: overseen data structures



Process (2)

- analyzing code / structures → thinking → idea → verifying
 - write code to see if it works
 - compare it with perf report

Process (2)

- analyzing code / structures → thinking → idea → verifying
 - write code to see if it works
 - compare it with perf report
- Andrzej / → contact to other people

Achievements

- good understanding of perf data format

- good understanding of perf data format
- "proof of concept" application (readperf), verified against perf report
 - only 1k lines of code (perf 34k)
 - no kernel code / dependencies
 - small dependencies

- good understanding of perf data format
- "proof of concept" application (readperf), verified against perf report
 - only 1k lines of code (perf 34k)
 - no kernel code / dependencies
 - small dependencies
- report as description of file format and readperf

- good understanding of perf data format
 - "proof of concept" application (readperf), verified against perf report
 - only 1k lines of code (perf 34k)
 - no kernel code / dependencies
 - small dependencies
 - report as description of file format and readperf
- ⇒ basis for performance analysis

Conclusion

- perf seems like historically grown
 - ⇒ complex, difficult to understand

Conclusion

- perf seems like historically grown
 - ⇒ complex, difficult to understand
- Open Source is good, still need time to understand the code
 - ⇒ but without code it would be much harder