Review

Marc-Andre Hermanns
JARA-HPC, RWTH Aachen University

Summary

- You've been introduced to a variety of tools
  - with hints to apply and use the tools effectively
- Tools provide complementary capabilities
  - computational kernel & processor analyses
  - communication/synchronization analyses
  - load-balance, scheduling, scaling, …
- Tools are designed with various trade-offs
  - general-purpose versus specialized
  - platform-specific versus agnostic
  - simple/basic versus complex/powerful

Tool selection

- Which tool you use and when depends on the situation
  - which are available on (or for) your computer system
  - which support your programming paradigms and languages
  - which you are familiar (comfortable) with
  - which type of issue you suspect
  - which question you want to have answered

- Being aware of tools (and their capabilities) can help finding the most appropriate tool

Workflow (getting started)

- First ensure that the parallel application runs correctly
  - no-one cares how fast you can compute the wrong answer
  - parallel debuggers help isolate known problems
  - correctness checking tools can help identify other issues
  - (that might not cause problems right now, but will eventually)
    - e.g., race conditions, invalid/non-compliant usage
- Start with an overview of execution performance
  - fraction of time spent in computation vs comm/synch vs I/O
  - which sections of the application/library code are most costly
- Investigate changes with different scales and configurations
  - processes vs threads, mappings, bindings

Workflow (communication/synchronization)

- These problems generally apply to every computer system (to different extents)
- Typically grow with the number of processes/threads
  - *Weak scaling*: fixed computation per thread, and perhaps fixed localities, but increasingly distributed
  - *Strong scaling*: constant total computation, increasingly divided amongst threads, while communication grows
  - Increased data movement through collective communications
    - particularly all-to-all-like communications
  - Synchronizations of larger groups are increasingly costly
  - Load-balancing becomes increasingly challenging, and imbalances increasingly expensive
    - generally manifests as waiting time at following collective ops

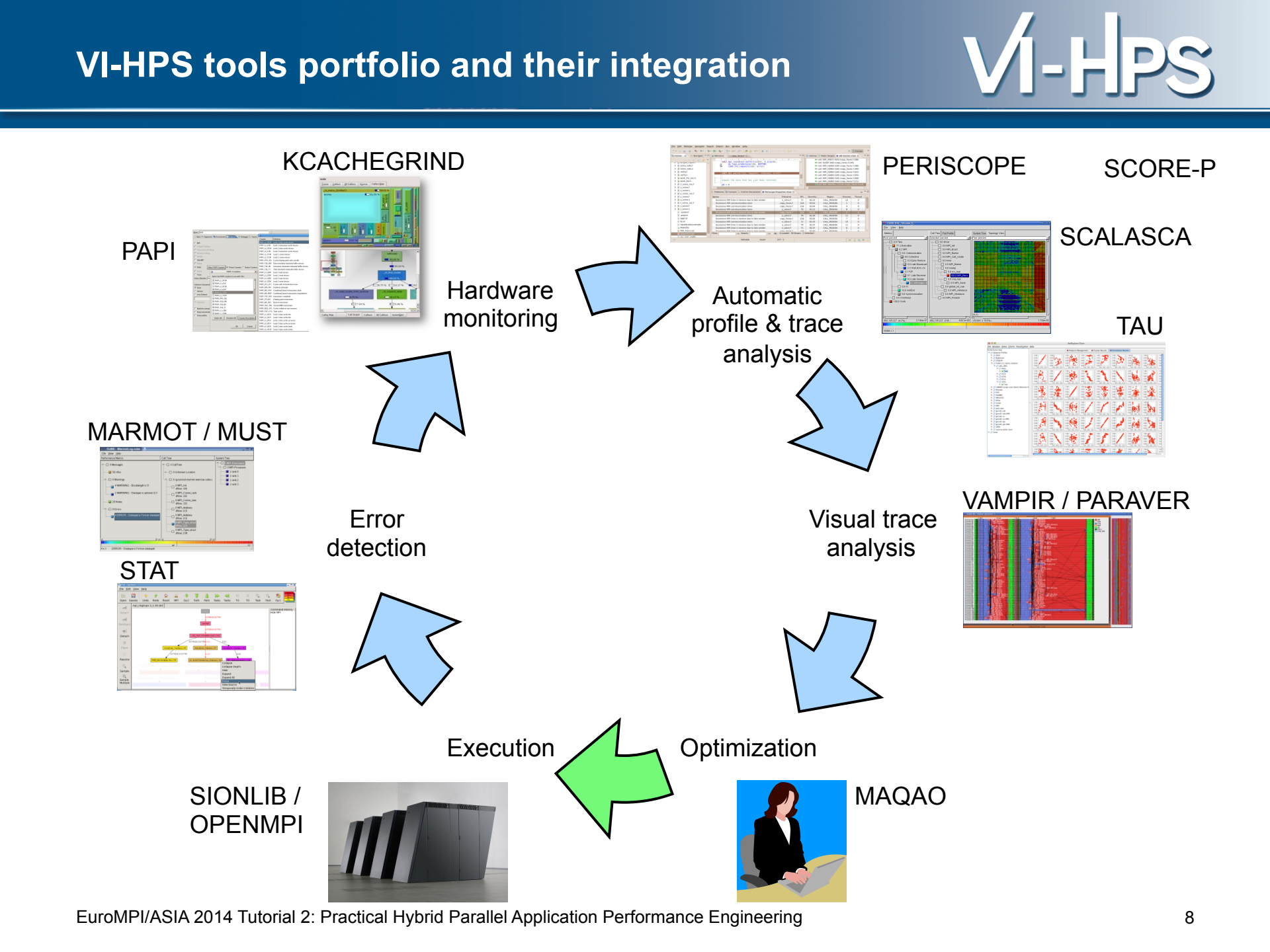
Workflow (waiting time)

- Waiting times are difficult to determine in basic profiles
  - Part of the time each process/thread spends in communication & synchronization operations may be wasted waiting time
  - May require correlation between processes
  - Online tools
    - *Periscope* uses augmented messages to transfer timestamps and additional on-line analysis processes
    - Score-P compares execution times to their estimated optimal execution time
  - Post-mortem tools avoid interference and provide a complete history
    - *Scalasca* automates trace analysis and ensures waiting times are completely quantified
    - *Vampir* allows interactive exploration and detailed examination of reasons for inefficiencies

Effective computation within processors/cores is also vital

- Use optimized libraries where available, as they may out-perform self-implemented and generic libraries
- Help the compiler to do most of the optimization work
  - provided the code is clearly written and not too complex
  - appropriate directives and other hints can also help
- Processor hardware counters can also provide insight
  - although hardware-specific interpretation required
- Tools available from processor and system vendors help navigate and interpret processor-specific performance issues

VI-HPS tools portfolio and their integration



Featured VI-HPS tools

- **Score-P**
  - community-developed instrumenter & measurement libraries for parallel profiling and event tracing
- **CUBE & ParaProf/PerfExplorer**
  - interactive parallel profile analyses
- **Scalasca**
  - automated event-trace analysis
- **Vampir**
  - interactive event-trace visualizations and analyses
- **TAU/PDT**
  - comprehensive performance system

VI-HPS Website

- Introductory information about the VI-HPS tool portfolio
  - VI-HPS Tools Guide
  - Links to individual tools sites for details and download
  - Training material
    - Tutorial slides
    - Latest ISO image of VI-HPS Linux DVD with productivity tools
    - User guides and reference manuals for tools
  - News of upcoming events
    - tutorials and workshops
    - mailing-list sign-up for announcements