

Introducing PThWay

Structured and methodical performance engineering

Isaías A. Comprés Ureña

Ventsislav Petkov

Michael Firbach

Michael Gerndt

Technical University of Munich

Overview

- **Tuning Challenges**
- Optimization Cycle
- PAThWay Features
- Implementation
- Architecture
- Scalability Analysis

Tuning Challenges

- Multiple performance characteristics
 - Application scalability
 - Scaling with processes and threads
 - Platform dependent optimizations
 - Single thread performance
 - MPI communication time
 - Communication and computation overlap
- Tuning for each aspect is time consuming
- Typically done by hand by a performance expert
 - But the process is well posed for automation

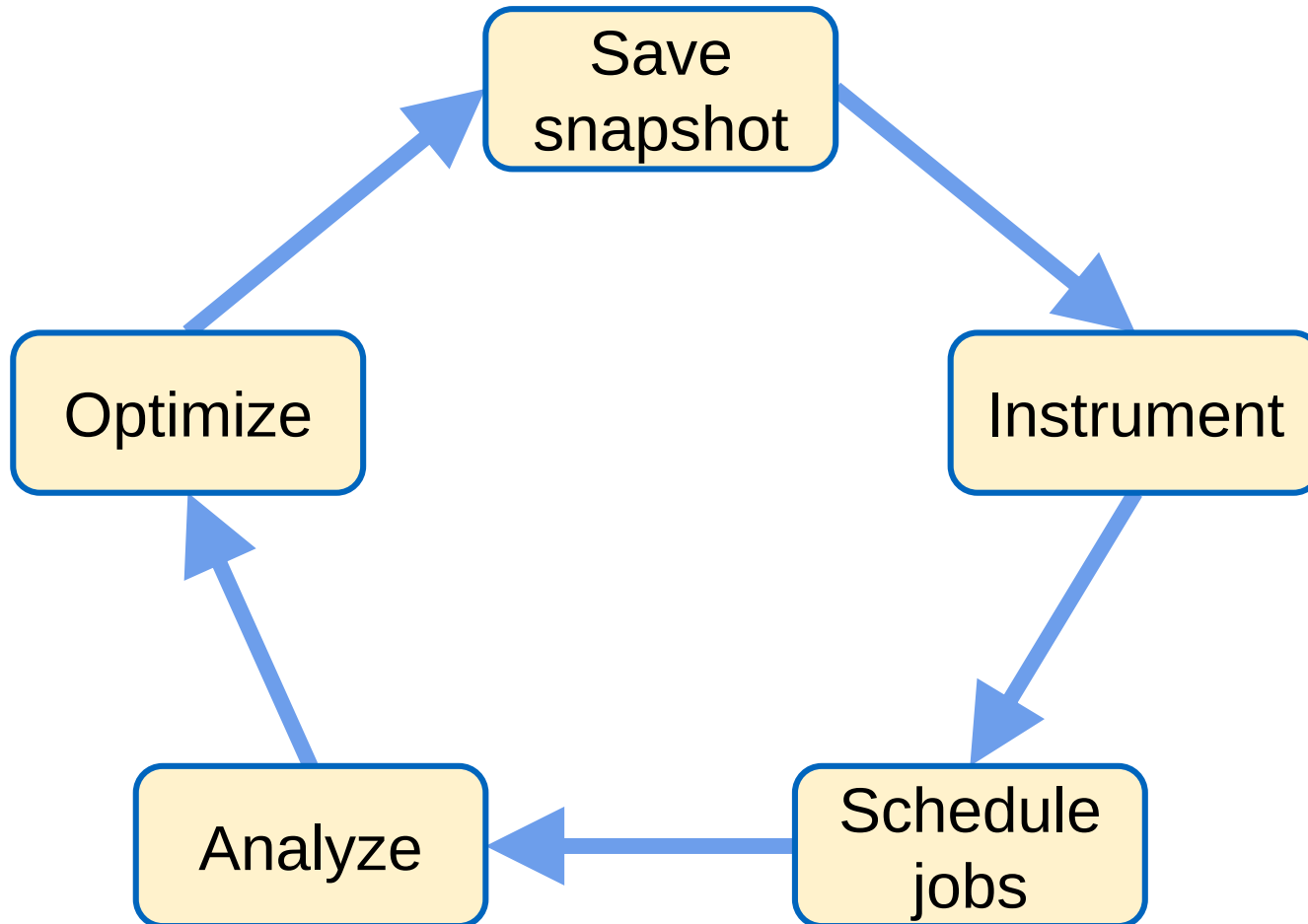
Tuning Challenges

- Large number of tools from multiple vendors
 - Compiler, instrumenter, profiler, visualization
 - Integration per vendor, but typically not across vendors
- Different HPC systems
 - Batch scripts (SLURM, LoadLeveler, ect.)
 - Package managers or software modules
 - Different Linux or Unix distributions and versions
 - Different runtime environments
 - Environment variables
- Makes cross-platform analysis difficult

Overview

- Tuning Challenges
- **Optimization Cycle**
- PAThWay Features
- Implementation
- Architecture
- Scalability Analysis

Optimization Cycle



Overview

- Tuning Challenges
- Optimization Cycle
- **PAThWay Features**
- Implementation
- Architecture
- Scalability Analysis

PATHWay Features

- Formal, graphical view on optimization process
 - Pre-defined workflows available
 - Can be customized with graphical editor
 - Can be explained to new team members
- Transparently serve different HPC systems
 - Define connection parameters for each system
 - PATHWay automatically generates batch scripts for different schedulers
 - Integrates with Eclipse's Parallel Tools Platform (PTP)

PATHWay Features

- Creates a snapshot of your application source
 - Does not interfere with your development repository
- Automatic tool invocations
 - Several tools from VI-HPS are preconfigured
 - Scalasca
 - Periscope
 - Score-P
 - Others can be added through configuration dialogs
 - More preconfigurations are planned

Experiment Browser

The screenshot shows the 'Experiment Browser' window with a table of experiments. A callout points to the table, and another points to the 'Experiment Details' window below it.

Date	Time	HPC System	Application	MPI Proce	OMP Thre	Performance Tool	Job status
Oct 22, 2013	October 15, 2013 3:25:17 PM CEST	Local	omp_prime_sum	1	1	Uninstrumented	Completed
Oct 21, 2013	October 15, 2013 3:25:17 PM CEST	Local	omp_prime_sum	1	2	Uninstrumented	Completed
Oct 18, 2013	October 15, 2013 3:25:17 PM CEST	Local	omp_prime_sum	1	3	Uninstrumented	Completed
Oct 16, 2013	October 15, 2013 3:25:17 PM CEST	Local	omp_prime_sum	1	4	Uninstrumented	Completed
Oct 15, 2013	October 15, 2013 3:25:17 PM CEST	Local	omp_prime_sum	1	5	Uninstrumented	Completed
Oct 14, 2013	October 15, 2013 3:25:17 PM CEST	Local	omp_prime_sum	1	6	Uninstrumented	Completed
Oct 11, 2013	October 15, 2013 3:25:17 PM CEST	Local	omp_prime_sum	1	7	Uninstrumented	Completed
Oct 10, 2013	October 15, 2013 4:27:22 PM CEST	Local	omp_prime_sum	1	8	Uninstrumented	Completed
Oct 9, 2013	October 15, 2013 4:27:22 PM CEST	Local	omp_prime_sum	1	1	Uninstrumented	Completed
Oct 8, 2013	October 15, 2013 4:27:22 PM CEST	Local	omp_prime_sum	1	2	Uninstrumented	Completed
Oct 7, 2013	October 15, 2013 4:30:41 PM CEST	Local	omp_prime_sum	1	4	Uninstrumented	Completed
Oct 4, 2013	October 15, 2013 4:30:41 PM CEST	Local	omp_prime_sum	1	5	Uninstrumented	Completed
Oct 2, 2013	October 15, 2013 4:30:41 PM CEST	Local	omp_prime_sum	1	6	Uninstrumented	Completed
Sep 30, 2013	October 15, 2013 4:59:35 PM CEST	Local	omp_prime_sum	1	1	Uninstrumented	Completed
Sep 29, 2013			omp_prime_sum	1	2	Uninstrumented	Completed
Sep 28, 2013			omp_prime_sum	1	5	Uninstrumented	Completed
Sep 27, 2013			omp_prime_sum	1	6	Uninstrumented	Completed
Sep 26, 2013			omp_prime_sum	1	7	Uninstrumented	Completed

Experiment Details

Number of threads: 5
Sum: 72619548630277
Time: 17.138833

Overview | Configuration | HPC System | Periscope | Environment | Output | Error log

Wiki-based Experiment Notes

[Wiki Home](#) [← Back](#) [→ Forward](#) [↻ Refresh](#) [✖ Stop](#) [🚀 Launch](#)

PATHWAY_NOTE.20130829

Experiments for 29. August 2013

- Cross-Platform Memory Analysis - [PAThWay:Exp:ff80818140ca86920140ca87702d0000](#)

Application: my_mpi_test - default
Tool: Uninstrumented
Number of sub-experiments: 2

- MPI: 2 / OpenMP: 1 (LRZ Linux Cluster New): ff80818140ca86920140ca87702d0000
- MPI: 4 / OpenMP: 1 (LRZ Linux Cluster New): ff80818140ca86920140ca87a5330001

- Cross-Platform Memory Analysis - [PAThWay:Exp:ff80818140ca62750140ca65a06b0000](#)

Application: my_mpi_test - default
Tool: Uninstrumented
Number of sub-experiments: 2

- MPI: 2 / OpenMP: 1 (LRZ Linux Cluster New): ff80818140ca62750140ca65a06b0000
- MPI: 4 / OpenMP: 1 (LRZ Linux Cluster New): ff80818140ca62750140ca65d7ca0001

- Scalability Analysis - [PAThWay:Exp:ff80818140cac69b0140cac6d7f70000](#)

Application: my_mpi_test - local
Tool: Uninstrumented
Number of sub-experiments: 8

- MPI: 1 / OpenMP: 1 (Local): ff80818140cac69b0140cac6d7f70000
- MPI: 2 / OpenMP: 1 (Local): ff80818140cac69b0140cac6dc330001

Wiki-based Experiment Notes

- Documentation is entered manually

PAThWay streamlines the process:

- Keeps track of experiments
- Prepares templates
- Central location in database

Overview

- Tuning Challenges
- Optimization Cycle
- PAtHWay Features
- **Implementation**
- Architecture
- Scalability Analysis

Implementation

- PATHWay is an Eclipse Plug-in
 - Easy installation from our Eclipse Update Site
 - Development & Optimization in one IDE
 - Many existing functionality reused
- Job management: Parallel Tools Platform
 - Manages SSH connections
 - Understands most batch schedulers
- Workflow definition and execution: jBPM
 - Uses standard BPMN workflow notation
 - Comes with graphical editor

Implementation

- Application snapshots: git
 - Hidden repository, does not interfere with your revision control
 - Fast and space-efficient
 - Fast checkpoint performance after initialization
 - Even on large code-bases, delays are not noticeable
- Experiment data, tools and system configuration
 - Stored in a database (H2 currently, with other connectors to follow)
 - Can be hosted centrally for a team

Implementation

- PArThWay is basically a vast collection of tools working together
 - Integrated in a common workflow
 - Like an IDE for code optimization

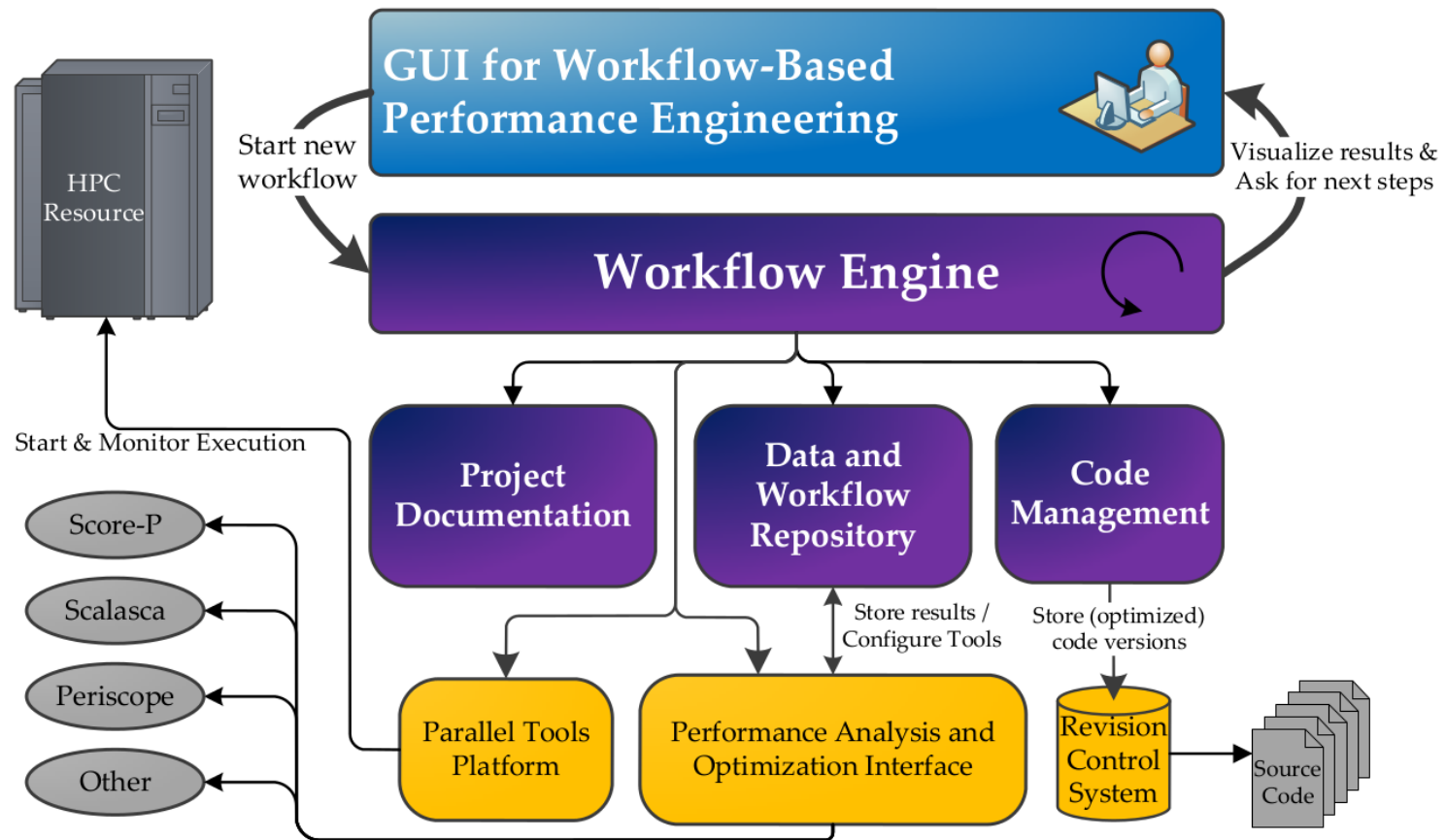
Recent Improvements

- Recent efforts were centered around usability:
 - Wizards for new projects/workflows
 - Simplified workflows (new elements)
 - Workflow execution can be aborted, jobs canceled
 - Improved error handling
 - Simplified installation
 - Score-P workflow

Overview

- Tuning Challenges
- Optimization Cycle
- PAThWay Features
- Implementation
- **Architecture**
- Scalability Analysis

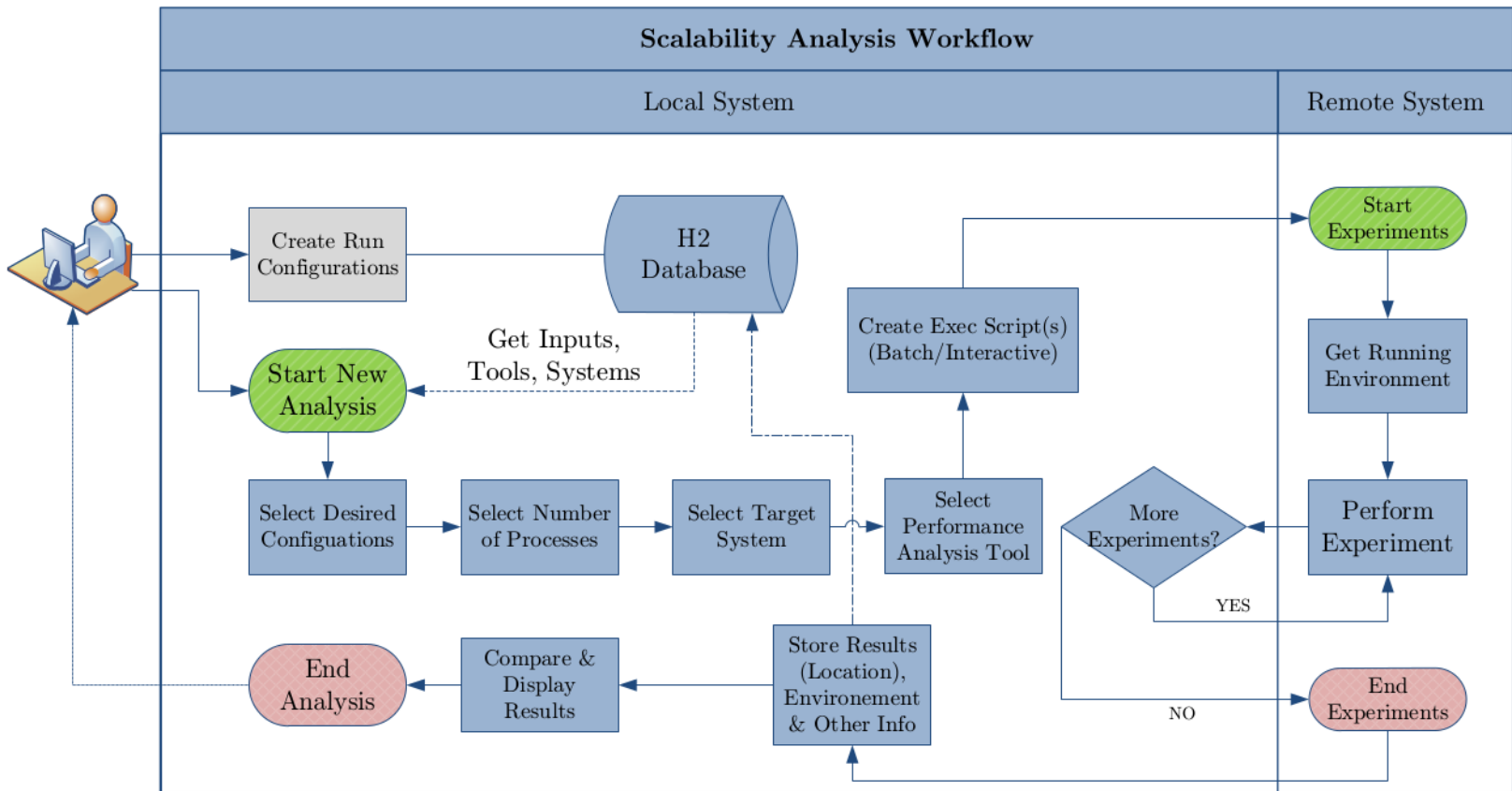
Architecture



Overview

- Tuning Challenges
- Optimization Cycle
- PAThWay Features
- Implementation
- Architecture
- **Scalability Analysis**

Scalability Analysis



Questions?