

# ***MPI test automation framework***

*Technical presentation*

*Attila Döme Lehóczy*

**Cranfield University, School of Engineering**  
**2012/2013**

Supervisor: Mark Stillwell

---

---

# *Background / Motivation: MPI*

- Inter-process communication API
- Language-independent
- Just a specification
- MPI implementations
  - OpenMPI
    - Used on Grid'5000
  - MPICH

# *Background / Motivation: Simulation*

- Algorithmic abstraction of a real-world system
- Performance analysis can be conducted
- It has multiple advantages...
- SimGrid
  - Generic simulation framework
    - Grids, Clouds, HPC, P2P systems
  - Scalable, extensible engine
  - Very active project
  - Favored by researchers

# *Background / Motivation: SMPI*

- Single-node MPI simulation
- Part of the SimGrid project
- Accurate, scalable and fast
  - Validated via a large set of experiments
- Actively developed
  - Continuous validation is necessary



# *Background / Motivation: Testing*

- A multiple-step process
    - Allocation of nodes
    - Deployment of images
    - Configuration
    - Post-processing
    - ...
  - No step-by-step, universal guide
  - Repetitive, error-prone
  - Need for a framework
- 
-

# Goals

- As much automation as possible
- Remove the tedious, repetitive steps
- Construction of workflows
- Modularity
  - Independent “blocks”
- Reusability
  - Interchangeable “blocks”



# *Implementation*

- XPflow
  - Experimentation engine
  - New project
    - Top-down approach, taken from Business Process Management
      - Understand the problem
      - Model it as a workflow
      - Execution and monitoring
      - Improve activities and processes

# *Implementation: XPflow*

- 2 main concepts
    - Processes
      - High-level description of an experiment
      - Orchestrate other processes and activities
      - Written in a DSL
    - Activities
      - Low-level building blocks
      - “Real” work
      - Written in Ruby
- 
-



# *Progress: Preparation*

- Testing “by hand”
  - Both on Astral and on Grid'5000
  - Configuration
  - Post-processing
  - Visualization
  - “Examples”: how testing is done

# *Progress: Implementation*

- Various features are implemented
    - Node reservation
    - Image deployment
    - Broadcast of runnables
  - Multiple methods under development
    - Actual execution of the experiment
    - Post-processing
    - Problems on Grid'5000 ...
  - Current implementation is fairly Grid'5000-specific
- 
-

# *Plans for the remaining time*

- Finish implementation
    - Execution
    - Post-processing
      - Gathering of traces
      - Conversion
  - Thorough testing
  - Maybe introduce a few other features
    - Metadata collection
    - As time allows...
  - Finish documentation
- 
-

# Summary

- SMPI
    - Huge amount of testing needed
    - Conducting tests is tedious
      - A repetitive, multiple-step process
      - Error-prone
  - Test automation framework
    - Real-life MPI tests
    - SMPI tests
    - Grid'5000-specific
    - A lot of possible directions for development
      - Time constraints ...
- 
-