Pathway User Guide

Contents

1 Introduction

Pathway is a tool for designing and executing performance engineering workflows for HPC applications. In the scope of READEX project, Pathway has been used to design a formal model of the READEX workflow. This design process has required the implementation of new features for helping to organize and guide the whole optimization process when the READEX tool suite is applied to HPC applications. The goal of this document is to serve as a guidance on how to use Pathway performance engineering workflow tool.

2 Installation

The following illustrates the instructions to install Pathway on a Debian-based Linux system:

2.1 User approach

- Install libwebkitgtk:

```
sudo apt-get install libwebkitgtk-1.0-0
```

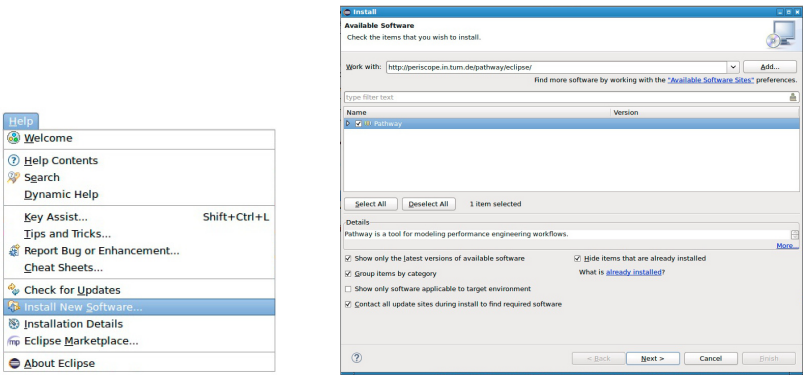
- Install git:

```
sudo apt-get install git
```

- Download the `Luna` release of Eclipse.
- In Eclipse, select the option `"Install new Software"` (Figure 1a) and add the following update site:

```
http://periscope.in.tum.de/pathway/eclipse/
```

- Install Pathway from the update site (Figure 1b).



(a) `"Install new Software"` (b) Installation from the update site.

Figure 1: Pathway installation.

3 Configuration

This section brings together the configuration aspects that Pathway provides to the user for applying the READEX tool suite to the HPC applications. The following subsections are distinguished:

- `Quick start:` it describes how to create a new Pathway project and to import the READEX workflow.
- `Application configuration:` it shows how to register and configure an application to be part of the READEX workflow execution.
- `HPC System configuration:` it shows the options that allow to register/edit HPC systems in Pathway.
- `Tool configuration:` it describes the options that allow to register/edit tools in Pathway.
- `READEX workflow configuration:` it is focused on the configuration aspects related to the READEX workflow items.

3.1 Quick start

Start the Eclipse installation in which you have installed Pathway and open the `"Pathway perspective"` (Figure 2) with the `"+"` button in the upper right corner. Remember that you can always reset the perspective's layout to its default by right-clicking on the corresponding button and selecting `"reset"`.

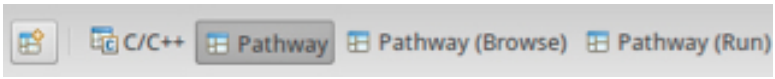


Figure 2: open the `"Pathway perspective"`.

Before you can start working on READEX workflow, you need to create a Pathway project from within the Navigator view (Figure 3). Within this project, you can create a new workflow by right-clicking on the project name and selecting the `"New"` menu and the `"Workflow"` option (Figure 4a). Finally, select the `"READEX"` option offered by the wizard (Figure 4b) and type a `"File name"` (Figure 5). Figure 6 depicts the imported READEX workflow.

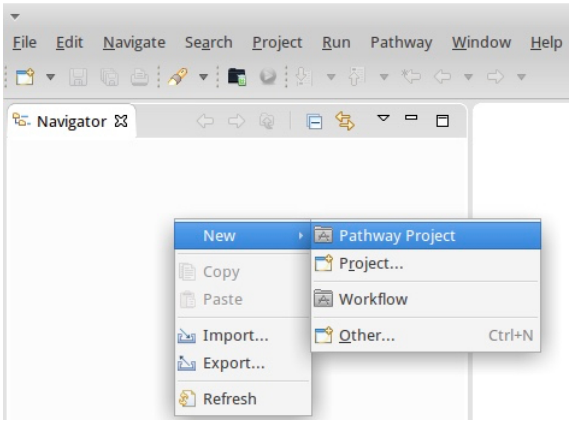
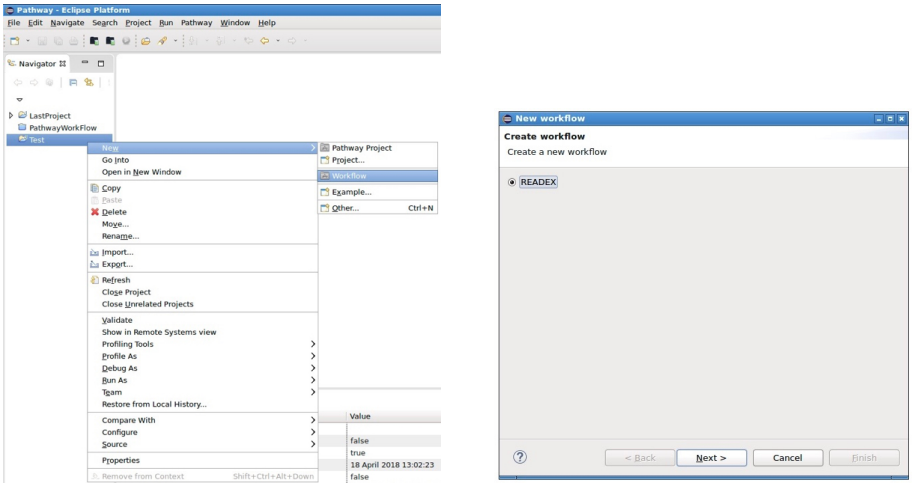


Figure 3: New Pathway project.



(a) New workflow. (b) READEX option.

Figure 4: Import READEX workflow.

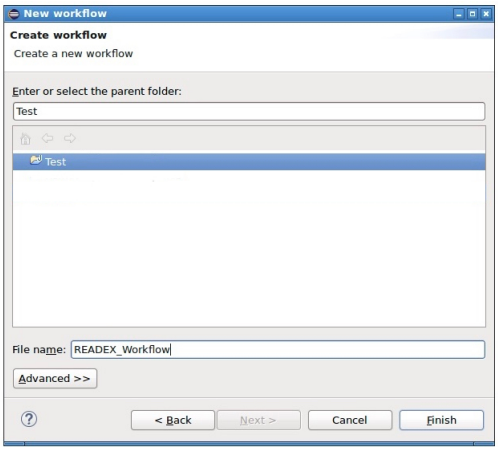


Figure 5: Enter `"File name"`.

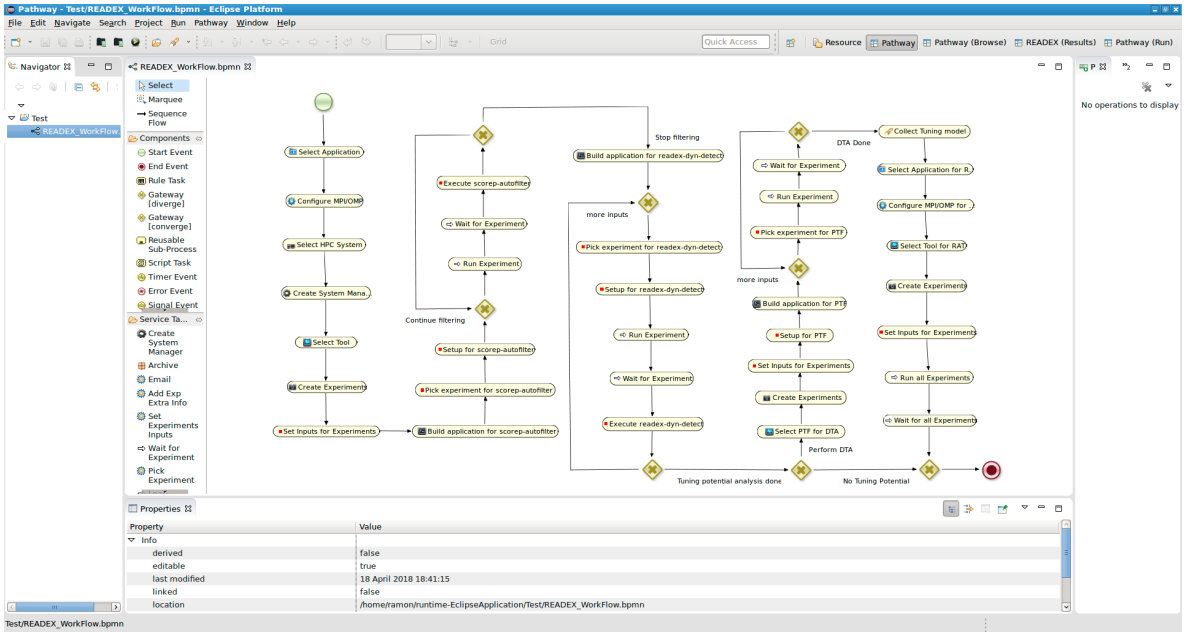


Figure 6: READEX workflow.

3.2 Application configuration

Before you can run a workflow with your application as the test subject, you need to let Pathway know how to use your application. To open the application configuration dialog, select "Pathway" from the menu and go to `"Edit Application"` (Figure 7).

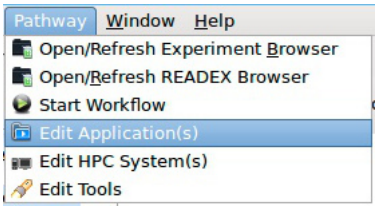


Figure 7: `"Edit Application"` menu.

Then, `"HPC Applications"` window (Figure 8) appears providing the functionality needed. Although its description is supported by an example (configuration of `"miniMD"` application), it is highly recommended the reading of READEX tool suite user guide for a better understanding of the READEX workflow.

- `New:` application registration in Pathway. Figure 9 depicts the registration of the `miniMD` application, `1.0` version.
- `Application:` selection of an application previously registered in Pathway.
- `Configuration:` selection of the application version previously registered.
- `Remove:` it allows removing an application configuration.

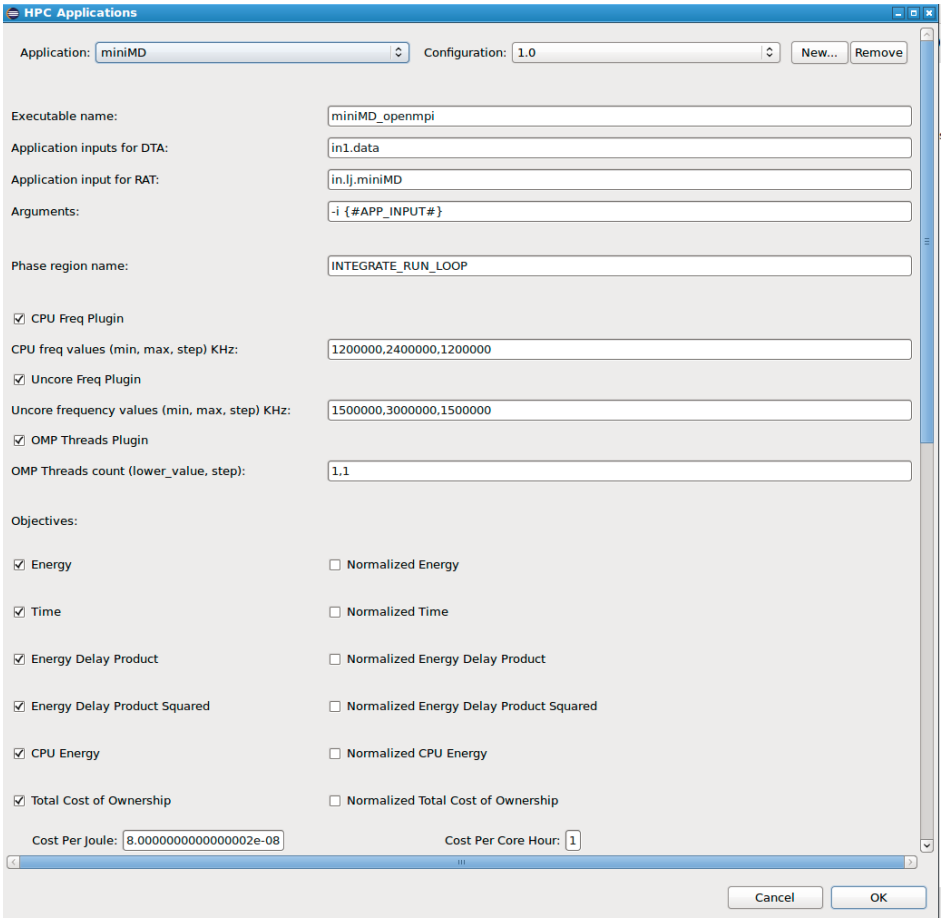


Figure 8: "HPC Applications" window (I).

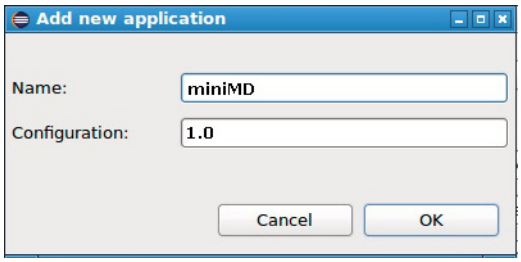


Figure 9: "Add new application" in Pathway.

- Executable name: enter the name of the executable file to run (e.g.: miniMD_openmpi).
- Application inputs for DTA: enter the application inputs for Design Time Analysis (e.g.: in1.data). For multiple inputs, it is necessary to separate them with commas (e.g.: in1.data, in.data2, in.data3).
- Application input for RAT: enter the application input for Runtime Application Tuning (e.g.: in.lj.miniMD).
- Arguments: enter the arguments to pass to the executable (e.g.: -i {#APP_INPUT#}).
- Phase region name: specify the phase region name of the application (e.g.: INTEGRATE_RUN_LOOP).

- `CPU Freq Plugin:` enable or disable the processor core frequency parameter.
- `CPU freq values (min, max, step) mHz:` specify the ranges (minimum, maximun and the step size) for the processor core frequency in kHz (e.g.: `1200,2400,200`).
- `Uncore Freq.Plugin:` enable or disable the uncore frequency parameter.
- `Uncore frequency values (min, max, step) mHz:` specify the ranges (minimum, maximun and the step size) for the uncore frequency in kHz (e.g.: `1200,3000,200`). Internally, these ranges are expressed in 100 million Hz.
- `OMP Threads Plugin:` enable or disable the number of OpenMP threads.
- `OMP Threads count (lower_value, step):` specify the lower bound and the step size to increment to the next value (e.g.: `1,2`).
- `Objectives:` select at least one objective from `"Energy"`, `"Execution Time"`, `"Energy Delay Product"`, `"Energy Delay Product Squared"`, `"CPU Energy"` and `"Total Cost of Ownership" (TCO)`. The normalized version of each objective can also be specified. To compute TCO, the `"Cost Per Joule"` and `"Cost Per Core Hour"` also needs to be specified (e.g.: Figure 8 shows the selection of 6 objectives. `"Cost Per Joule"` and `"Cost Per Core Hour"` has been set to 8.0000000000000002e-08 and 1).

Figure 10 depicts the functionality of `"HPC Applications"` window when moving the scroll bar down. We continue with its description:

- `Energy metrics:` specify the energy plugin name (`"Plugin name"`) and associated metric names (`"Metric names"`). Figure 10 shows an example of energy metrics configuration.
- `PTF search algorithm`: specify a search algorithm from `"Exhaustive"` (see Figure 10), `"Individual"`, `"Random"`, or `"Genetic"`. Configuration examples of all search strategies are described in the `"How to use READEX Tool Suite"` guide. The following settings are mandatory:
  - `"Individual"`: number of tuning parameter values to `"Keep"` in the search space.
  - `"Random"`: number of samples `"Samples"` that the plugin should limit to.
  - `"Genetic"`: `"Population Size"` , the maximum number of generations `"Max.Generations"` and the `"Timer"` to set an upper limit on the tuning execution time.
- `ATP library:` enable or disable `"ATP library"`. When ATP library is enabled, it is mandatory to specify a search algorithm from among `"ATP Exhaustive"` and `"ATP Individual"` strategies.

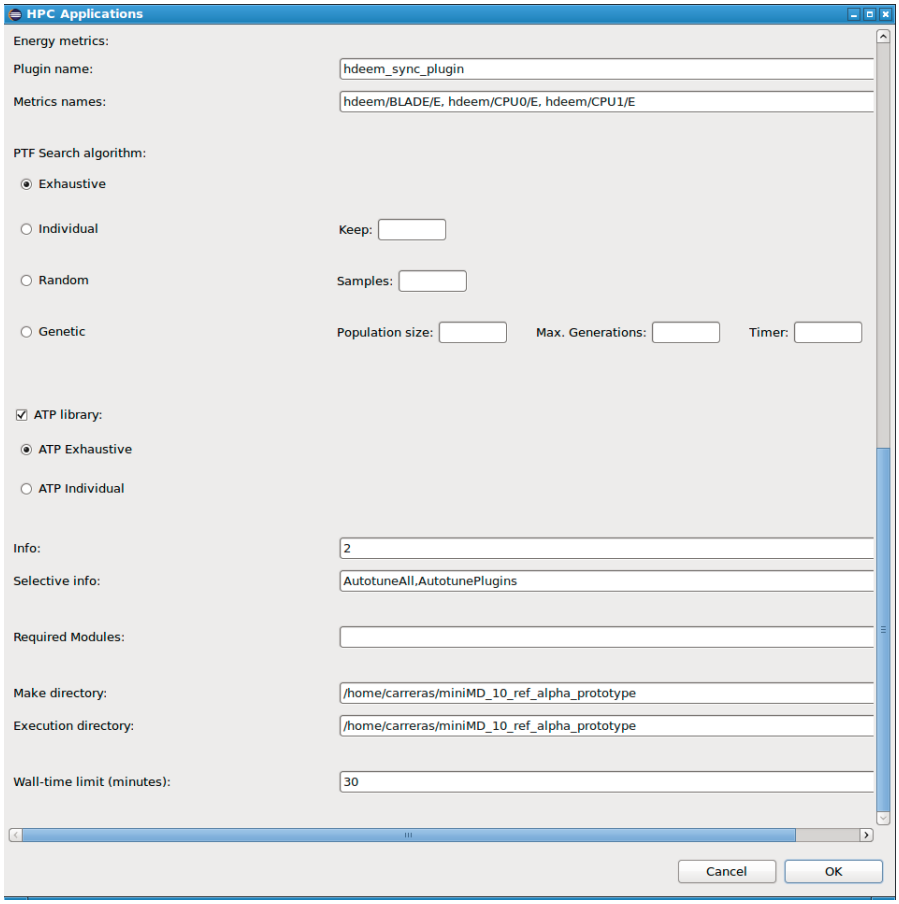


Figure 10: `"HPC Applications" window (II).`

- `Info:` set this option (not mandatory) between 2 and 7 according with the desired debug output (see Figure 10).
- `Selective info:` enter a comma sepated list of information levels (not mandatory) to be reported (see Figure 10).
- `Required Modules:` specify a comma sepated list of the required modules for running the application.
- `Make directory:` if Pathway is supposed to call make, this indicates the location of the makefile (see Figure 10).
- `Execution directory:` specify the directory where the executable can be found and is executed (see Figure 10).
- `Wall-time limit (minutes):` set the limit on the total run time of the job allocation, in minutes (see Figure 10).

  Note that is possible to use placeholders in many of the above fields. Using a placeholder, you can insert the value of a workflow variable during the execution of the workflow. The general syntax is: {`#VARIABLE:varName#`}. So if you want to run executables from test1 to test4 for example, you can specify `"test{#VARIABLE:i#}"` as the executable name, where i is a workflow variable you increment within the logic of your workflow.

3.3 HPC System Configuration

The HPC system configuration settings are available by clicking on the Pathway `"Edit HPC System" menu` (Figure 11).

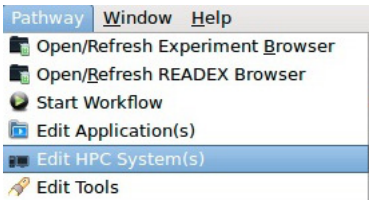
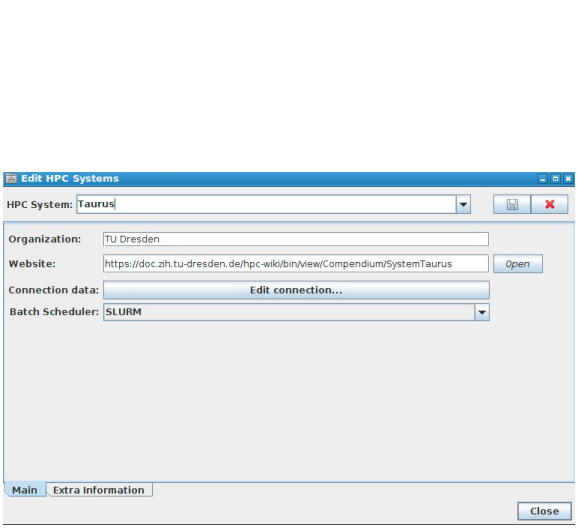


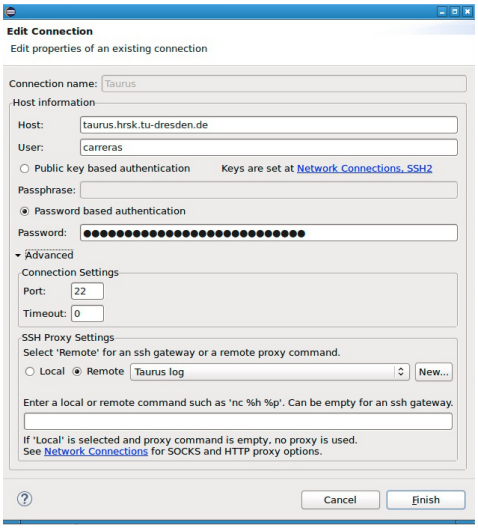
Figure 11: `"Edit HPC System"` menu.

After selecting this menu, the `"Edit HPC Systems"` window (Figure 12) appears providing the functionality required through two tabs:

- `Main` tab: it contains mandatory fields to be specified by the user. Figure 12a depicts a configuration example of `Taurus`:
  - `HPC System`: enter HPC system name. Options to save and delete HPC system on the right.
  - `Organization`: enter the organization name which maintains the HPC system.
  - `Website`: specify the website related to the HPC system. Option to redirect to the link on the right.
  - `Batch Scheduler`: specify the batch scheduler of the HPC system.
  - `Connection data`: set the connection properties to establish communication with the HPC system. Figure 12b shows a configuration example for securely getting access to `Taurus` via SSH.
- `Extra Information` tab: it contains fields (not mandatory) to add extra information about the HPC system.



(a) `"Main"` tab.



(b) `"Edit Connection"` window.

Figure 12: `"Edit HPC System"` window.

3.4 Tool Configuration

The Tool configuration settings are available through the Pathway `"Edit Tools" menu` (Figure 13).

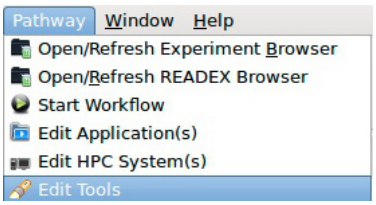


Figure 13: `"Edit Tools"` menu.

Then, `"Performance Tools"` window (Figure 14) appears providing the functionality to configure a tool in Pathway:

- `New:` tool registration in Pathway. Figure 14 depicts the registration of the `READEX Runtime Application Tuning (RAT)` tool, `1.0` version.
- `Tool name:` selection of a tool previously registered in Pathway.
- `Version:` selection of the tool version previously registered.
- `Remove:` it allows removing a tool configuration.
- `Website:` specify the website related to the tool (not mandatory).

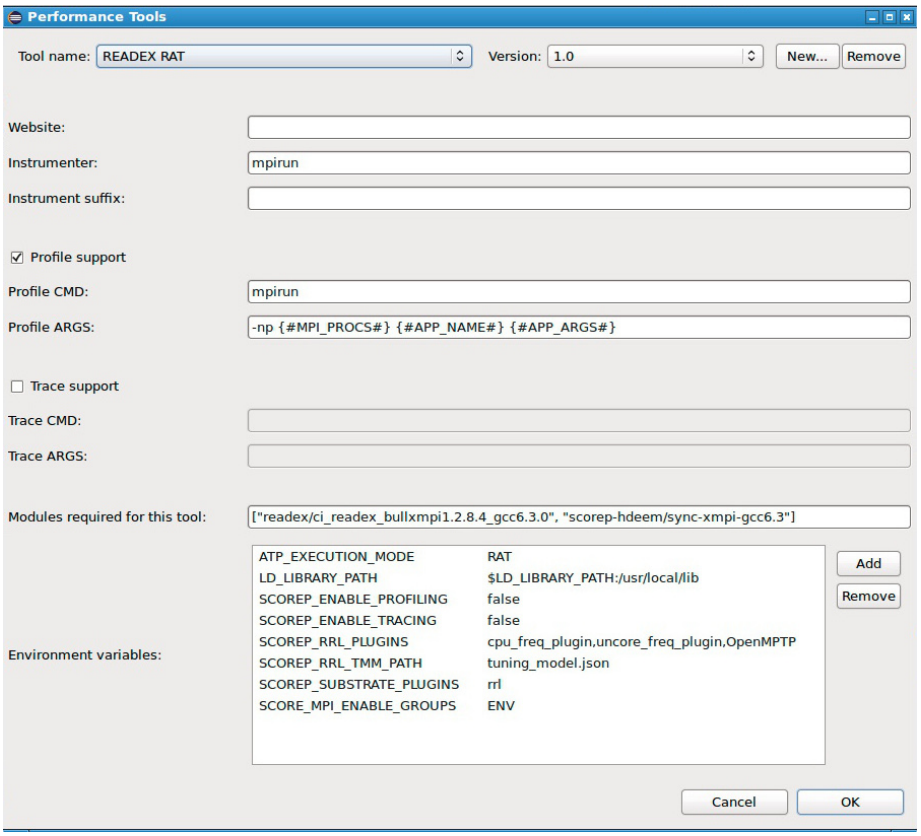


Figure 14: `"Performance Tools"` window.

- `Instrumenter:` specify the instrumenter. Previously, mpirun has been chosen for executing serial and parallel jobs in Open MPI (Figure 14).
- `Instrumenter suffix:` enter the instrumenter suffix.
- `Profile support:` enable or disable profile support.
- `Profile CMD:` specify the command to apply the tool.
- `Profile ARGS:` specify the input arguments of the tool.
- `Trace support:` enable or disable trace support (not mandatory).
- `Trace CMD:` specify the command associated with trace support (not mandatory).
- `Trace ARGS:` specify the arguments related to trace support (not mandatory).
- `Modules required for this tool:` specify the READEX modules required.
- `Environment variables:` set the environment variables using `"Add"` and `"Remove"` buttons.

3.5 READEX Workflow Configuration

The last configuration step is performed by selecting the `"Pathway"` perspective and it consists on the customization of the following workitems that compose the READEX workflow (Figure 6):

- `Build application for scorep-autofilter:` enter the values of properties `"Script"` (script to build HPC application for scorep-autofilter) and `"WorkingDir"` (working directory). Figure 15 depicts a configuration example of building `miniMD` application for scorep-autofilter being `sh compile_for_saf.sh` and `/home/carreras/miniMD_10_ref_alpha_prototype/` the values specified by the user.

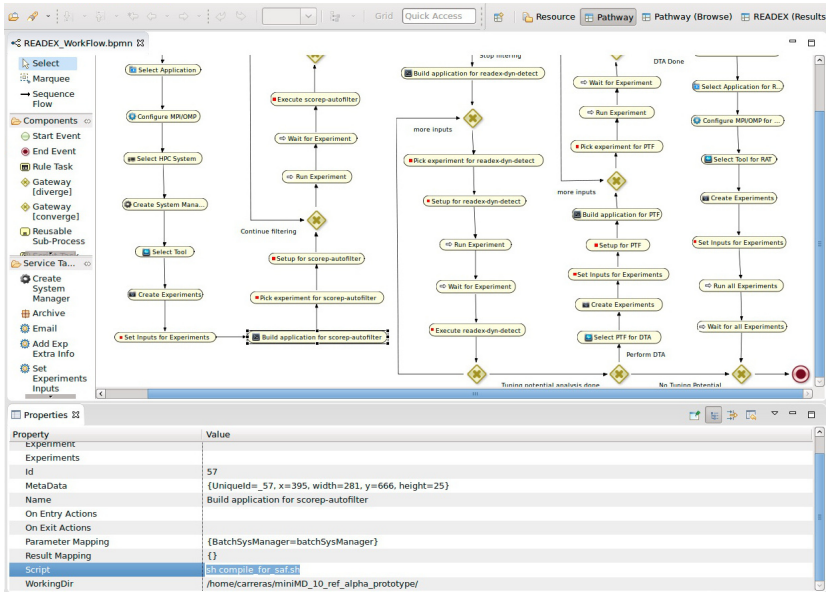


Figure 15: `"Workitem configuration"` example (Build application for scorep-autofilter).

- `Execute scorep-autofilter:` specify the value of property `"Arg_t"` (region granularity threshold in seconds). Use a value larger than 0.1 (100ms).

- `Build application for readex-dyn-detect:` enter the values of properties `"Script"` (script to build HPC application for readex-dyn-detect) and `"WorkingDir"` (working directory). For instance, the values specified to build `miniMD` were `sh compile_for_rdd.sh` and `/home/carreras/miniMD_10_ref_alpha_prototype/`.
- `Execute readex-dyn-detect:` specify the value of the following properties (see `"How to use READEX Tool Suite"` manual):
  - `"Arg_t":` region granularity threshold in seconds. Use a value larger than 0.1 (100ms).
  - `"Arg_p":` name of the phase region as given in the instrumentation (e.g. `INTEGRATE_RUN_LOOP`).
  - `"Arg_c":` compute intensity variation threshold (e.g. 10).
  - `"Arg_v":` execution time variation threshold in percent (e.g. 10).
  - `"Arg_w":` region execution time weight with regards to phase execution time in percent (e.g. 10).
- `Setup for PTF:` enter the value of property `"WorkingDir"`.
- `Build application for PTF:` enter the values of properties `"Script"` (script to build HPC application for Periscope Tuning Framework (PTF) tool) and `"WorkingDir"` (working directory). For instance, the values specified to build `miniMD` were `sh compile_for_ptf.sh` and `/home/carreras/miniMD_10_ref_alpha_prototype/`.
- `Collect Tuning model:` enter the value of property `"WorkingDir"`.

4 Operation

4.1 READEX Workflow Execution

The execution of the READEX worflow is carried out from the `"Pathway (Run)"` perspective. Initially, you can select your system and application in the execution parameters view shown to the right (see Figure 16). You may want to specify some of the other parameters also. Not all parameters need to be specified, but Pathway may ask you then during workflow execution for the required information. In general, it is more comfortable to set the values up-front.

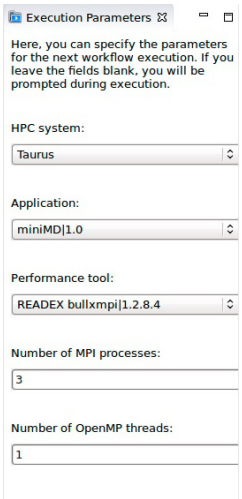
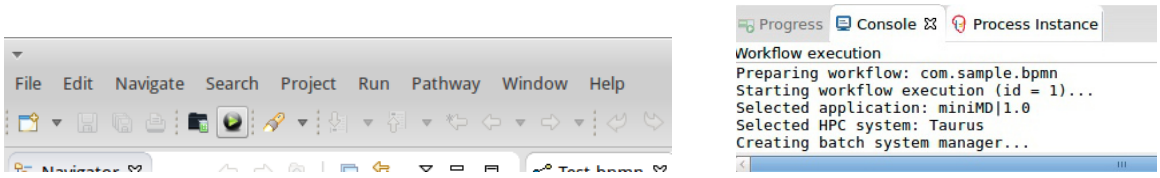


Figure 16: `"Execution parameters"` menu.

There are several possible formats in which you can specify the number of `"MPI processes"` and `"OMP threads"` to use. Since Pathway is all about automation, it is possible to specify a range of numbers and the experiment will then be performed for each configuration. Below is a list of possible formats:

- Single number: performs the experiment one time (e.g. 5).
- List of numbers: performs the experiment three times (e.g. 5, 12, 20).
- Range of numbers: for instance, 4:8 performs the experiment 5 times (for 4 . . . 8 threads).
- Range with step size: for instance, 2:8:2 performs the experiment four times (for 2, 4, 6 and 8 threads).

Finally, click on the `"Start workflow"` button (see Figure 17a) to start the workflow execution. Pathway will show the progress of the workflow execution. In the editor-like view, the workflow is shown, with the current activity highlighted. Another important view is the log window (see Figure 17b), where you can find specific information about what Pathway is currently doing.



(a) `"Start the Workflow"` button. (b) `"log"` window.

Figure 17: Start-up and monitorization of workflow execution.

4.2 Browsing results

In order to see the results of the workflow execution, switch to the `"READEX Browser"` (see Figure 18). The READEX Browser perspective is composed by 2 windows. The window on the left contains a list of HPC applications. The window on the right shows READEX information regarding the application selected by the user. It should be noted that every time an application is selected, the READEX information is updated. There are different types of READEX information:

- `Performed experiments:` information related to the `Date`, `MPI Processes`, `OMP Threads`, `Input Data File` and `HPC System`. Selecting one experiment allows you to see all its details in the other READEX information sources.
- `Name` of each significant region and its time `variation` related to `inter-phase dynamism`.
- `Name` of each significant region and its `Granularity`, `Weight`, `Time Variation Reg`, `Time Variation Phase` and `Compute Intensity` related to `intra-phase dynamism`.
- `Tuning Model:` contains the list of `runtime situations (rtss)` tuned by PTF tool, the `scenarios` they are classified into, and the best configuration for each scenario.

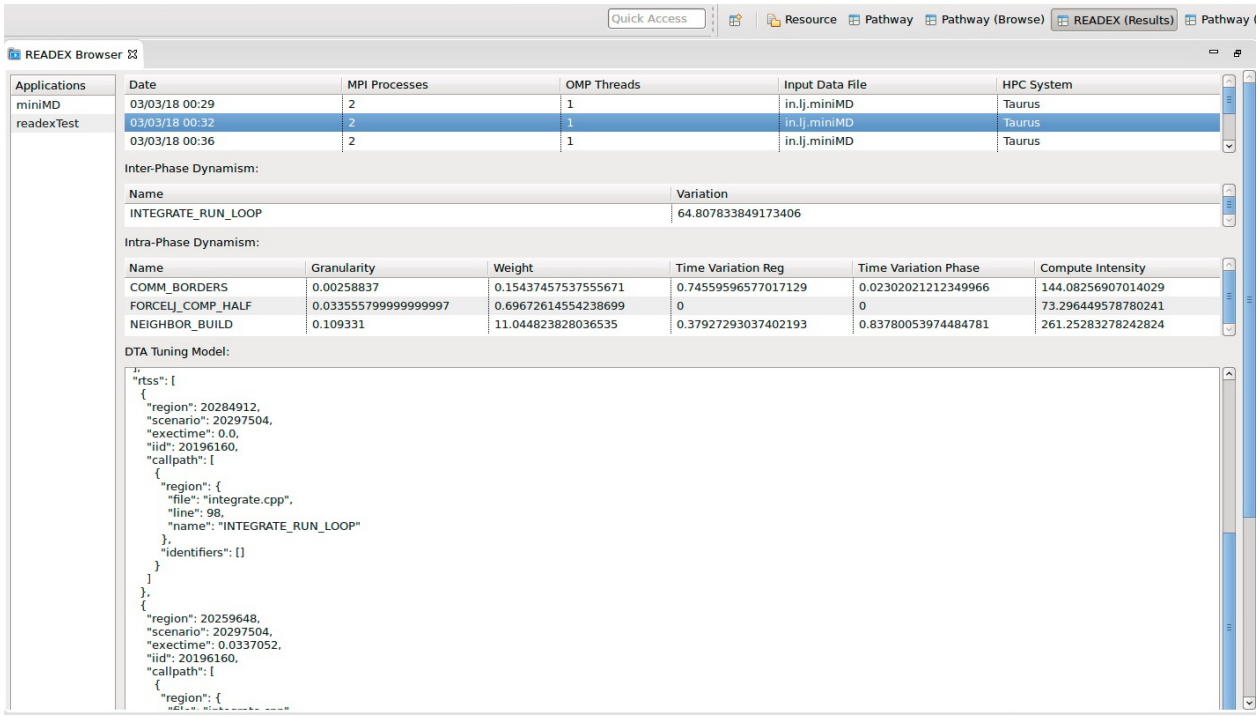


Figure 18: "READEX Browser".