



**Barcelona
Supercomputing
Center**
Centro Nacional de Supercomputación



BasicAnalysis, BurstClustering & CUDA cheatsheet

Judit Giménez, Lau Mercadal, [Germán Llort](#)

✉ tools@bsc.es

06/09/24

POP3 Training – BSC Tools

Setting up the environment in Karolina

- Source this file that sets paths for the tools after logging:

```
karolina> source /mnt/proj2/dd-24-88/bsctools/sourceme.sh
```

- Only the first time, you also need to install this dependency to use BasicAnalysis:

```
karolina> pip install -user seaborn
```

Commands to run BasicAnalysis

```
karolina> BasicAnalysis <trace1.prv> ... <traceN.prv>
```

- In addition to the output text, it produces graphical images with colored tables and other plots that can be displayed with any image viewer:

```
karolina> eog efficiency_table-matplot.png
```

Commands to run BurstClustering

- A sample configuration file “cluster.xml” can be found at `$CLUSTERING_HOME/share/example`

```
karolina> BurstClustering  
          -d cluster.xml \  
          -i <input_trace.prv> \  
          -o <output_trace.prv>
```

- The output scatterplot can be displayed with Gnuplot:

```
karolina> gnuplot <output_trace.IPC.PAPI_TOT_INS.gnuplot>
```

- And the clustered trace can be opened with Paraver:

```
karolina> wxparaver <output_trace.prv>
```

Changes to instrument CUDA / OpenACC

- Select tracing library in **trace.sh** / **LD_PRELOAD**

```
export LD_PRELOAD=$EXTRAE_HOME/lib/libcudatrace[f].so
```

CUDA: libcudatrace[f].so

MPI+CUDA: libcudampitrace[f].so

OpenACC: liboaccudatrace[f].so

MPI+OpenACC: liboaccudampitrace[f].so

** Trailing "f" is for Fortran-codes*

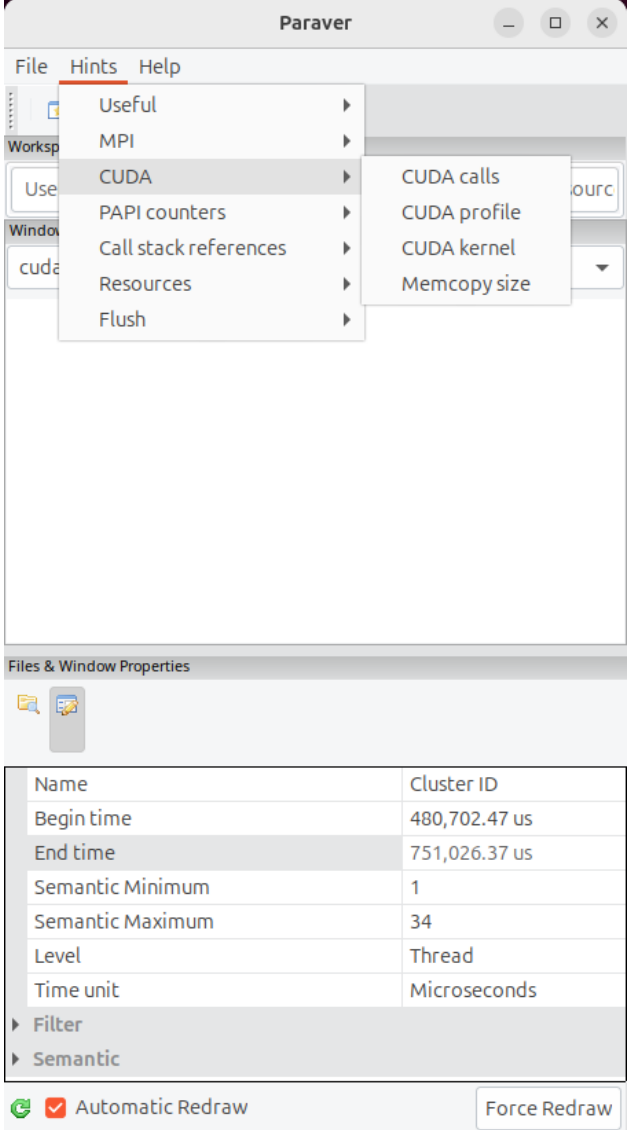
- Turn on these options (or add these lines) in **extrae.xml**

```
<cuda enabled="yes" />
```

```
<openacc enabled="yes" />
```

New Paraver Hints

- When traces contain CUDA events, Paraver suggests new views:
 - CUDA calls → Timeline of CUDA API calls
 - CUDA profile → Table with CUDA calls statistics
 - CUDA kernel → Timeline with GPU kernel executions
 - Memcopy size → Timeline with bytes copied in/out the GPU



The screenshot shows the Paraver application window with the 'Hints' menu open. The menu items are: Useful, MPI, CUDA, PAPI counters, Call stack references, Resources, and Flush. The 'CUDA' item is selected, and a sub-menu is visible with the following options: CUDA calls, CUDA profile, CUDA kernel, and Memcopy size.

Below the menu, the 'Files & Window Properties' section is visible, containing a table with the following data:

Name	Cluster ID
Begin time	480,702.47 us
End time	751,026.37 us
Semantic Minimum	1
Semantic Maximum	34
Level	Thread
Time unit	Microseconds

At the bottom of the window, there is a 'Filter' section and a 'Semantic' section. The 'Automatic Redraw' checkbox is checked, and there is a 'Force Redraw' button.