



**Barcelona  
Supercomputing  
Center**  
*Centro Nacional de Supercomputación*



EXCELENCIA  
SEVERO  
OCHOA

# Extrae & Paraver Hands-On

Judit Giménez, Germán Llort, Lau Mercadal,  
Sandra Méndez

✉ [tools@bsc.es](mailto:tools@bsc.es)

19/04/2021

POP Performance Analysis

# Extrace features

- Platforms
  - Intel, Cray, BlueGene, MIC, ARM, Android, Fujitsu Sparc ...
- Parallel programming models
  - MPI, OpenMP, pthreads, OmpSs, CUDA, OpenCL, Java, Python ...
- Performance Counters
  - Using PAPI interface
- Link to source code
  - Callstack at MPI routines
  - OpenMP outlined routines
  - Selected user functions (Dyninst)
- Periodic sampling
- User events (Extrace API)

No need  
to  
recompile  
or relink!

# How does Extrae work?

- Symbol substitution through LD\_PRELOAD
  - Specific libraries for each combination of runtimes
    - MPI
    - OpenMP
    - OpenMP+MPI
    - ...
- Dynamic instrumentation
  - Based on Dyninst (developed by U.Wisconsin / U.Maryland)
    - Instrumentation in memory
    - Binary rewriting
- Alternatives
  - Static link (i.e., PMPI, Extrae API)

Recommended

# Using Extrae in 3 steps

1. **Adapt** your job submission scripts
2. **Configure** what to trace
  - XML configuration file
  - Example configurations at \$EXTRAE\_HOME/share/example
3. **Run it!**
  - For further reference check the **Extrae User Guide**:
    - <https://tools.bsc.es/doc/html/extrae>
    - Also distributed with Extrae at \$EXTRAE\_HOME/share/doc

# Log in to MareNostrum4

```
laptop$ ssh -Y <USER>@mn1.bsc.es
```

- The following directory contains all the examples:

```
mn$ cp -R /home/nct00/nct00003/tools-material  
$HOME  
mn$ ls -l $HOME/tools-material  
bin/  
clustering/  
extrae/  
slides/←  
traces/
```

Here you have a copy of this slides.

Copy them to your laptop  
or open remotely with:

> evince slides/Extrae-Paraver-Hands-On.pdf

# Step 1: Adapt the job script to load Exrae

```
mn$ vi $HOME/tools-material/exrae/job_27p.sh
```

```
#!/usr/bin/env bash

#SBATCH --job-name=lulesh2.0_i_27p
#SBATCH --output=lulesh2.0_i_27p.out
#SBATCH --error=lulesh2.0_i_27p.err
#SBATCH --ntasks=27
#SBATCH --cpus-per-task=1
#SBATCH --time=00:10:00
#SBATCH --exclusive
#SBATCH --reservation=Training21

export OMP_NUM_THREADS=1

srun --cpu-bind=cores ./bin/lulesh2.0_i -i 10 -s
65 -p
```

Request resources

Run the program

# Step 1: Adapt the job script to load Exrae

```
mn$ vi $HOME/tools-material/exrae/job_27p.sh
```

```
#!/usr/bin/env bash

#SBATCH --job-name=lulesh2.0_i_27p
#SBATCH --output=lulesh2.0_i_27p.out
#SBATCH --error=lulesh2.0_i_27p.err
#SBATCH --ntasks=27
#SBATCH --cpus-per-task=1
#SBATCH --time=00:10:00
#SBATCH --exclusive
#SBATCH --reservation=Training21

export OMP_NUM_THREADS=1
srun --cpu-bind=cores ./trace.sh .../bin/lulesh2.0_i
...
```

Run with Exrae

# Step 1: Adapt the job script to load Extrae

```
mn$ vi $HOME/tools-material/extrae/trace.sh
```

```
#!/usr/bin/env bash

#SBATCH --job-name=lulesh2.0_i_27p
#SBATCH --output=lulesh2.0_i_27p.out
#SBATCH --error=lulesh2.0_i_27p.err
#SBATCH --ntasks=27
#SBATCH --cpus-per-task=1
#SBATCH --time=00:10:00
#SBATCH --exclusive
#SBATCH --reservation=Training21

export OMP_NUM_THREADS=1

srun --cpu-bind=cores ./trace.sh ./bin/lulesh2.
...
```

```
#!/usr/bin/env bash
Load Extrae
module load gcc/7.2.0 EXTRAE/3.8.3
export TRACE_NAME=lulesh2.0_27p.prv

What to trace?
# Configure Extrae
export EXTRAE_CONFIG_FILE=extrae.xml

Type of application
# Load the tracing library (choose C/Fortran)
export
LD_PRELOAD=$EXTRAE_HOME/lib/libmpitrace.so
#export
LD_PRELOAD=$EXTRAE_HOME/li

# Run the program
$*
```

# Step 1: Which tracing library?

- Choose depending on the application type

Library	Serial	MPI	OpenMP	pthread	CUDA
libseqtrace	x				
libmpitrace[f] <sup>1</sup>		x			
libomptrace			x		
libpttrace				x	
libcudatrace					x
libompitrace[f] <sup>1</sup>		x	x		
libptmpitrace[f] <sup>1</sup>		x		x	
libcudampitrace[f] <sup>1</sup>		x			x

<sup>1</sup> include suffix “f” in Fortran codes

# Step 3: Run it!

- Submit your job

```
mn$ cd $HOME/tools-material/exrae  
mn$ sbatch job_27p.sh
```

- Easy!

# Step 2: Extrace XML configuration

```
mn$ vi $HOME/tools-material/extrace/extrace.xml
```

```
<mpi enabled="yes">
  <counters enabled="yes" />
</mpi>

<openmp enabled="no">
  <locks enabled="no" />
  <counters enabled="yes" />
</openmp>

<pthread enabled="no">
  <locks enabled="no" />
  <counters enabled="yes" />
</pthread>

<callers enabled="yes">
  <mpi enabled="yes">1-3</mpi>
  <sampling enabled="no">1-5</sampling>
</callers>
```

Trace the MPI calls  
(What's the program doing?)

Trace the call-stack  
(Where in my code?)

# Step 2: Extrae XML configuration (II)

```
mn$ vi $HOME/tools-material/extrae/extrae.xml
```

```
<counters enabled="yes">
  <cpu enabled="yes" starting-set-distribution="1">
    <set enabled="yes" domain="all" changeat-time="0">
      PAPI_TOT_INS,PAPI_TOT_CYC
    </set>
  </cpu>
  <network enabled="no" />
  <resource-usage enabled="no" />
  <memory-usage enabled="no" />
</counters>
```

Select which  
HW counters  
are measured  
(How's the machine doing?)

## Step 2: Extrae XML configuration (III)

```
mn$ vi $HOME/tools-material/extrae/extrae.xml
```

```
<buffer enabled="yes">  
  <size enabled="yes">500000</size>  
  <circular enabled="no" />  
</buffer>
```

Trace buffer size  
(Flush/memory trade-off)

```
<sampling enabled="no" type="default" period="50m" variability="10m" />
```

```
<merge enabled="yes"  
  synchronization="default"  
  tree-fan-out="16"  
  max-memory="512"  
  joint-states="yes"  
  keep-mpits="yes"  
  sort-addresses="yes"  
  overwrite="yes">  
  $TRACE_NAME$  
</merge>
```

Enable sampling  
(Want more details?)

Automatic  
post-processing  
to generate the  
Paraver trace

# All done! Check your resulting trace

- Once finished (check with “squeue”) you will have the trace (3 files):

```
mn$ ls -l $HOME/tools-material/extrac  
lulesh2.0_i_27p.pcf  
lulesh2.0_i_27p.prv  
lulesh2.0_i_27p.row
```

- Any trouble? Traces already generated here:

```
mn$ ls $HOME/tools-material/traces
```

- Now let's look into it !

# Install Paraver

- Download from <https://tools.bsc.es/downloads>

Pick your version

The screenshot shows a dark-themed web interface for downloading Paraver. On the left, a blue sidebar says "Pick your version". The main area lists download links for different operating systems:

- Windows: wxparaver-4.9.2-win.zip
- Mac OS X: wxparaver-4.9.2-mac.zip
- Linux (32-bit): wxparaver-4.9.2-Linux\_i686.tar.gz
- Linux (64-bit): wxparaver-4.9.2-Linux\_x86\_64.tar.gz

Below these, there's a command-line example:

```
laptop$ scp <USER>@mn1.bsc.es:tools-packages/<PKG>
```

At the bottom, there are sections for related tools:

- Get CLUSTERING**: Version 2.6.6 • 2 MB. Options: 101 RAW, 32, 64. +
- Get TRACKING**: Version 2.6.5 • 1.9 MB. Options: 101 RAW, 32, 64. +
- Get FOLDING**: Version 1.0.2 • 11.06 MB. Options: 101 RAW, 32, 64. +
- SPECTRAL**: Signal processing techniques to select representative regions from Paraver traces. Get SPECTRAL: Version 3.4.0 • 0.31 MB. Options: 101 RAW, 32, 64. +
- BASIC ANALYSIS**: Framework for automatic extraction of fundamental factors for Paraver traces. Get BASIC ANALYSIS: Version 0.2 • 10.89 MB. Options: 101 RAW. +

# Install Paraver (II)

```
laptop$ tar xf wxparaver-4.9.2-linux-x86_64.tar.gz  
laptop$ mv wxparaver-4.9.2-linux-x86_64 paraver
```

- Start Paraver

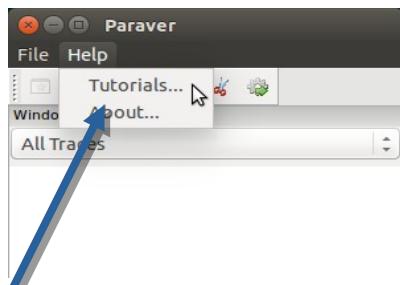
```
laptop$ paraver/bin/wxparaver
```

- Trouble installing locally? Remote open from MareNostrum4

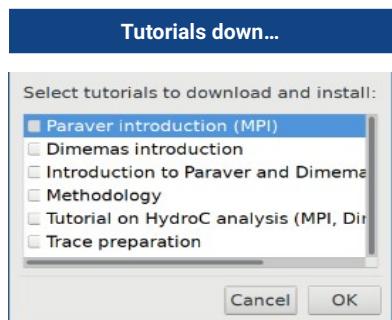
```
mn$ module load paraver  
mn$ wxparaver
```

# Install Paraver tutorials (I)

- Download tutorials:



Tutorials window  
will pop-up



**Tutorials**

 **Barcelona Supercomputing Center**  
Centro Nacional de Supercomputación

**No tutorials found!?**

**Install using the download dialog**

You can automatically download and install any of the available tutorials by clicking the "Download and Install" button.

Just check in the desired tutorials and press the "OK" button.

**Manual installation**

Please check that a **root directory** to tutorials is properly defined:

1. Open the [Preferences Window](#).
2. Select Global tab.
3. In the **Default directories** box, change the **Tutorials root directory**.
4. Save your new settings by clicking the **Ok** button in the [Preferences Window](#).
5. After that, we will automatically refresh the tutorials list.
6. If nothing happens, come back here and press the **Index** button (the first one at the bottom-left) to rebuild the tutorials list.

If the button **Index** doesn't seem to work (you're still reading this help!), please verify that:

- Every tutorial is **uncompressed**.
- Every tutorial is inside its own **subdirectory**.
- These subdirectories (or tutorials) are copied/linked into the root directory that you have selected before (i.e.: /home/myuser/mytutorials/tut1/, /home/myuser/mytutorials/tut2/, etc).
- Every tutorial has a main **index.html** (i.e.: /home/myuser/mytutorials/tut1/index.html ).

If you still get this help after checking these steps again, please contact us at [paraver@bsc.es](mailto:paraver@bsc.es).

**Latest tutorials**

Find them available at <https://tools.bsc.es/paraver-tutorials>

- As single **.tar.gz** package (127 MB).
- As single **.zip** package (127 MB).

Click on Help  
Tutorials

Follow these tutorials by clicking on the hyperlinks and reading the explanations. When you click on a link, multiple views will open.

# Install Paraver tutorials – alternative methods(II)

- Download tutorials archive
  - <https://tools.bsc.es/paraver-tutorials>

The screenshot shows a web browser displaying the BSC (Barcelona Supercomputing Center) website. The URL in the address bar is `news@tools:~ > Paraver 4.7.2 available`. The page title is "Paraver tutorials". The navigation menu includes Home, Paraver », Dimemas », Extrae, Research », Documentation » (which is active), Downloads, and Publications.

Below the title, the text reads: "These seven tutorials can be opened with wxParaver versions newer than 4.3.0, and you'll be able to follow the steps within the tool. To install them, download and untar the package and follow the instructions of the Help/Tutorial option on the Paraver main window. Following there is a list of available tutorials:"

<a href="#">Paraver introduction (MPI)</a>	Start here to familiarize with Paraver basic commands and the first steps of a performance analysis.
<a href="#">Dimemas introduction</a>	The basic steps to learn how to configure and run the Dimemas simulator and to start looking at the results.
<a href="#">Introduction to Paraver and Dimemas methodology</a>	This tutorial presents different ways to analyze a MPI application through well-known rules, their diagnosis and how they impact on your exploration (no traces included).
<a href="#">Methodology</a>	This tutorial shows some examples of the analysis that can be done using the provided configuration files.
<a href="#">Tutorial on HydroC analysis (MPI, Dimemas, CUDA)</a>	One example of performance analysis of the MPI application Hydro and further simulations with Dimemas.
<a href="#">Trace preparation</a>	Look at this tutorial to select a representative region for a large trace that cannot be loaded into memory.
<a href="#">Trace alignment tutorial.</a>	If you identify some unexpected unalignment or backwards communications, use this tutorial to learn how to correct shifts between processors.

If you prefer you can download all of them together in a single package:

[.tar.gz format \(127 Mb\)](#)

[.zip format \(127 Mb\)](#)

**All tutorials**

**paraver-tutorials-20150526.tar.gz**

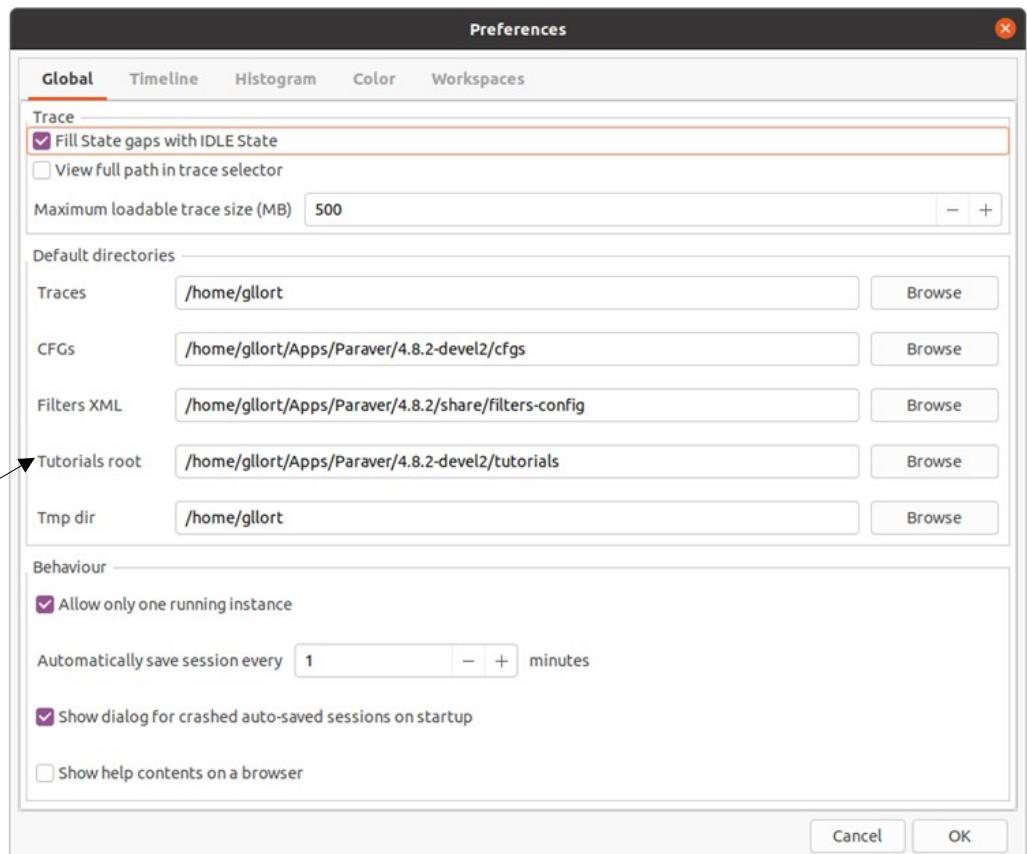
# Install Paraver tutorials – alternative methods(III)

- Start Paraver:
  - Linux: Run the command:
- Windows: Double-click on paraver/wxparaver.exe
- MAC: Double click on paraver/wxparaver.app

```
laptop$ paraver/bin/wxparaver
```

- Open File → Preferences  
Setup the “Tutorials root” pointing to your folder “tutorials”

**Click Browse and select your folder “tutorials”**



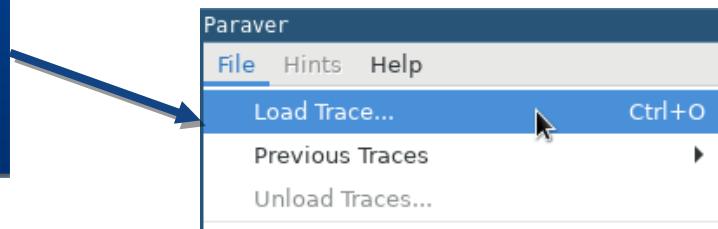
# First steps of analysis

- Copy the trace to your laptop

```
laptop$ scp <USER>@mn1.bsc.es: \
tools-material/extrاء/lulesh2.0_i_27p.* $HOME
```

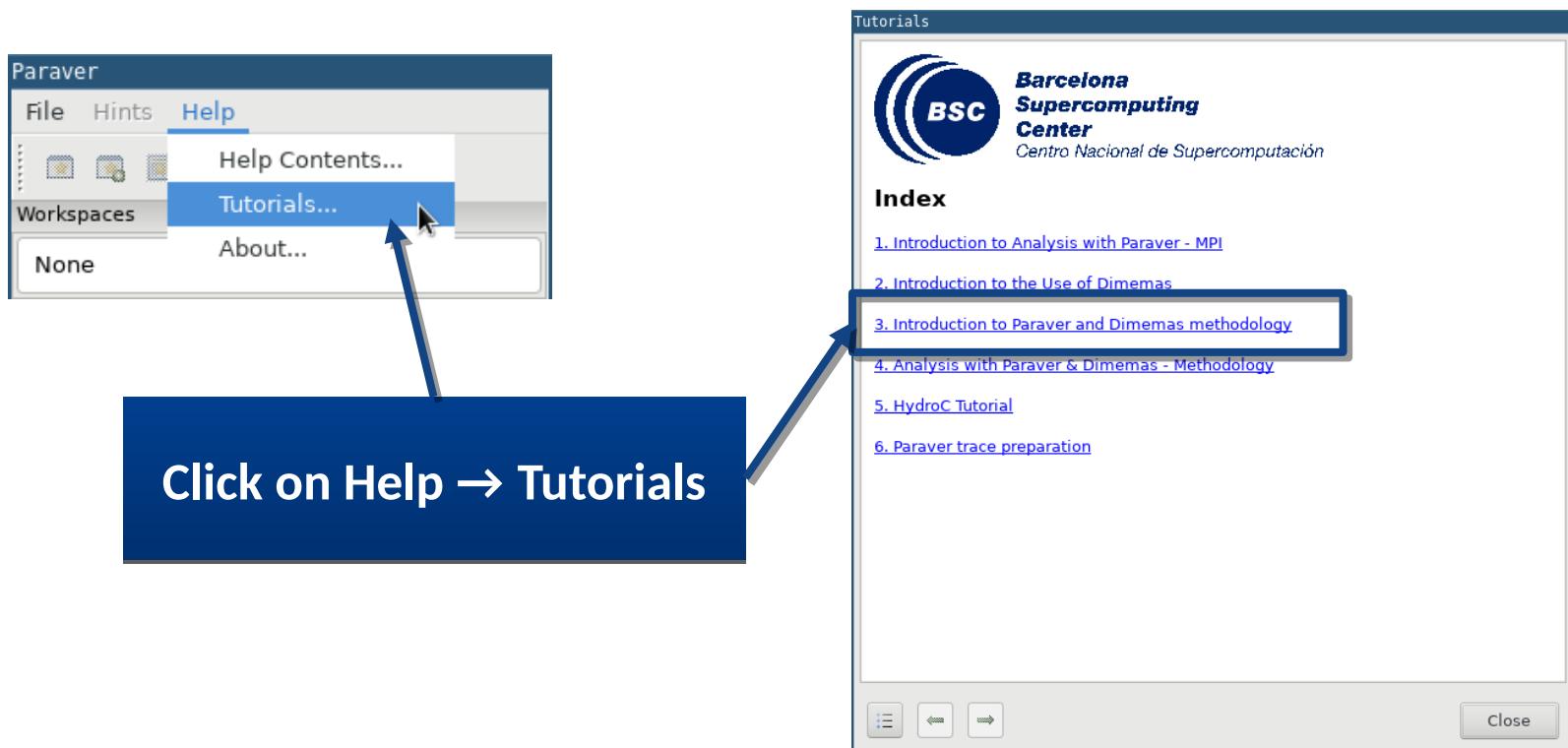
- Load the trace with Paraver

Click on File → Load Trace  
→ Browse to  
“lulesh2.0\_i\_27p.prv”



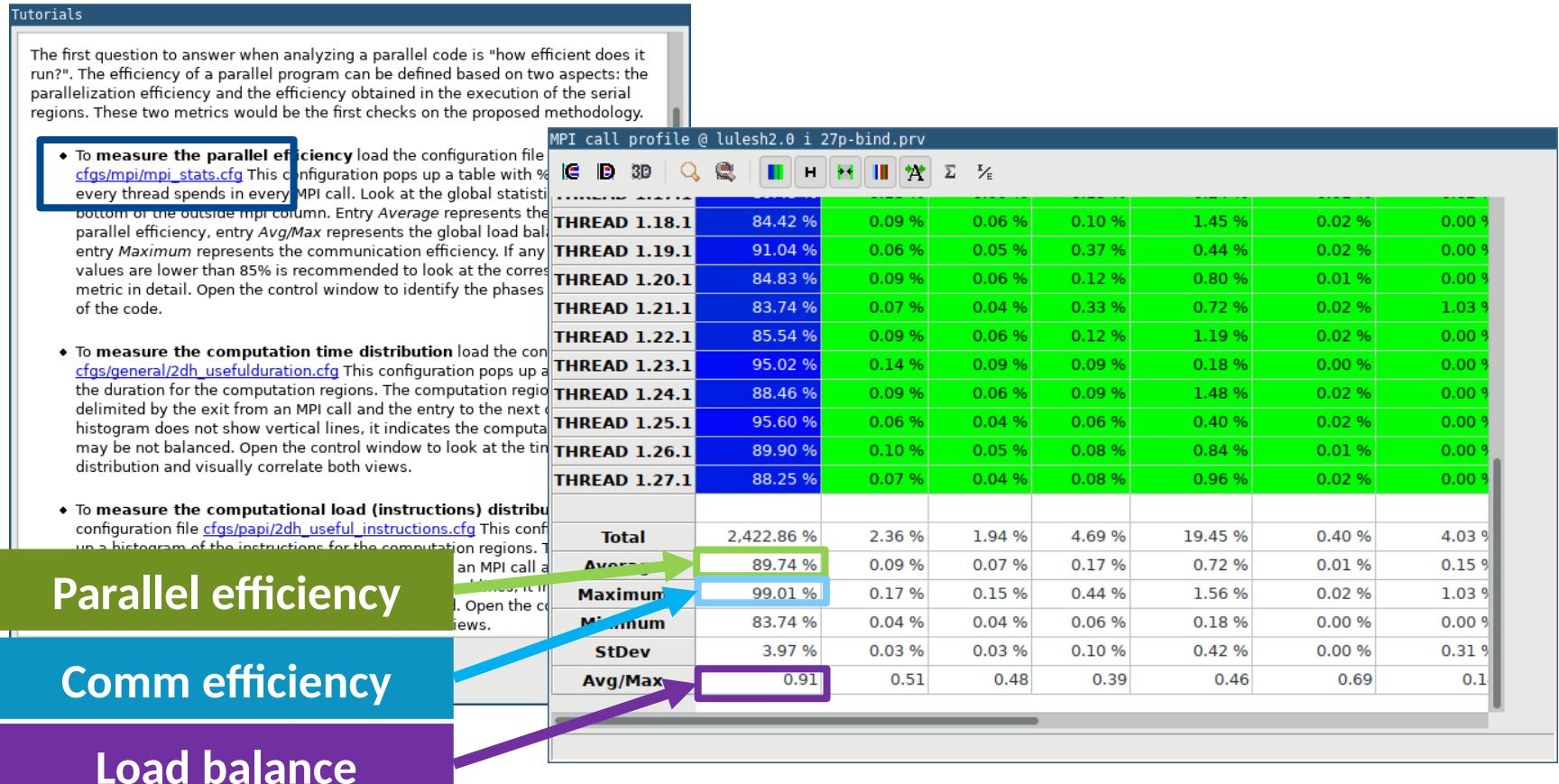
# First steps of analysis

- Follow Tutorial #3
  - Introduction to Paraver and Dimemas methodology



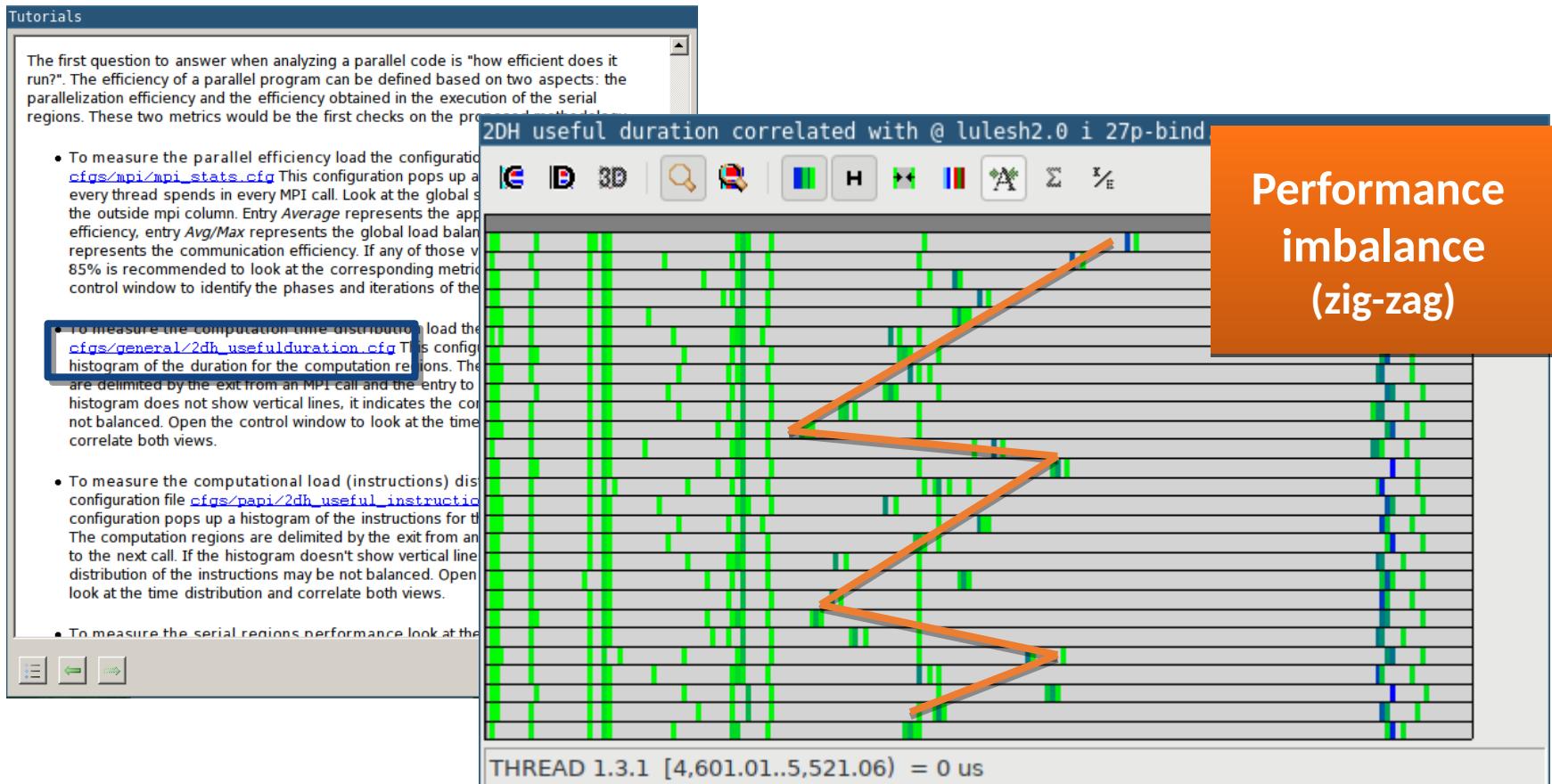
# Measure the parallel efficiency

- Click on “mpi\_stats.cfg”
  - Check the Average for the column labeled “Outside MPI”



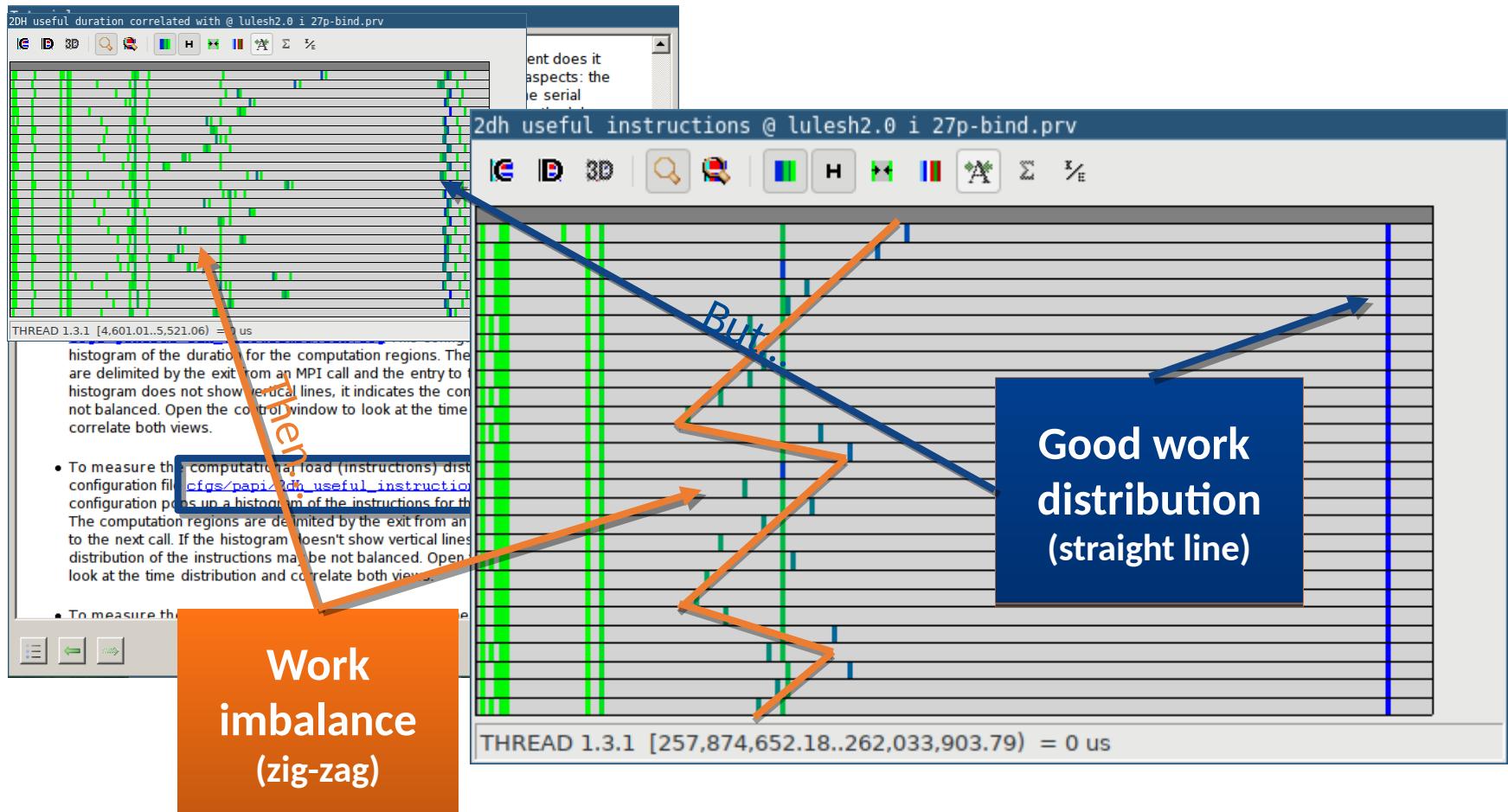
# Computation time distribution

- Click on “2dh\_usefulduration.cfg” (2nd link) ☺ Shows time computing



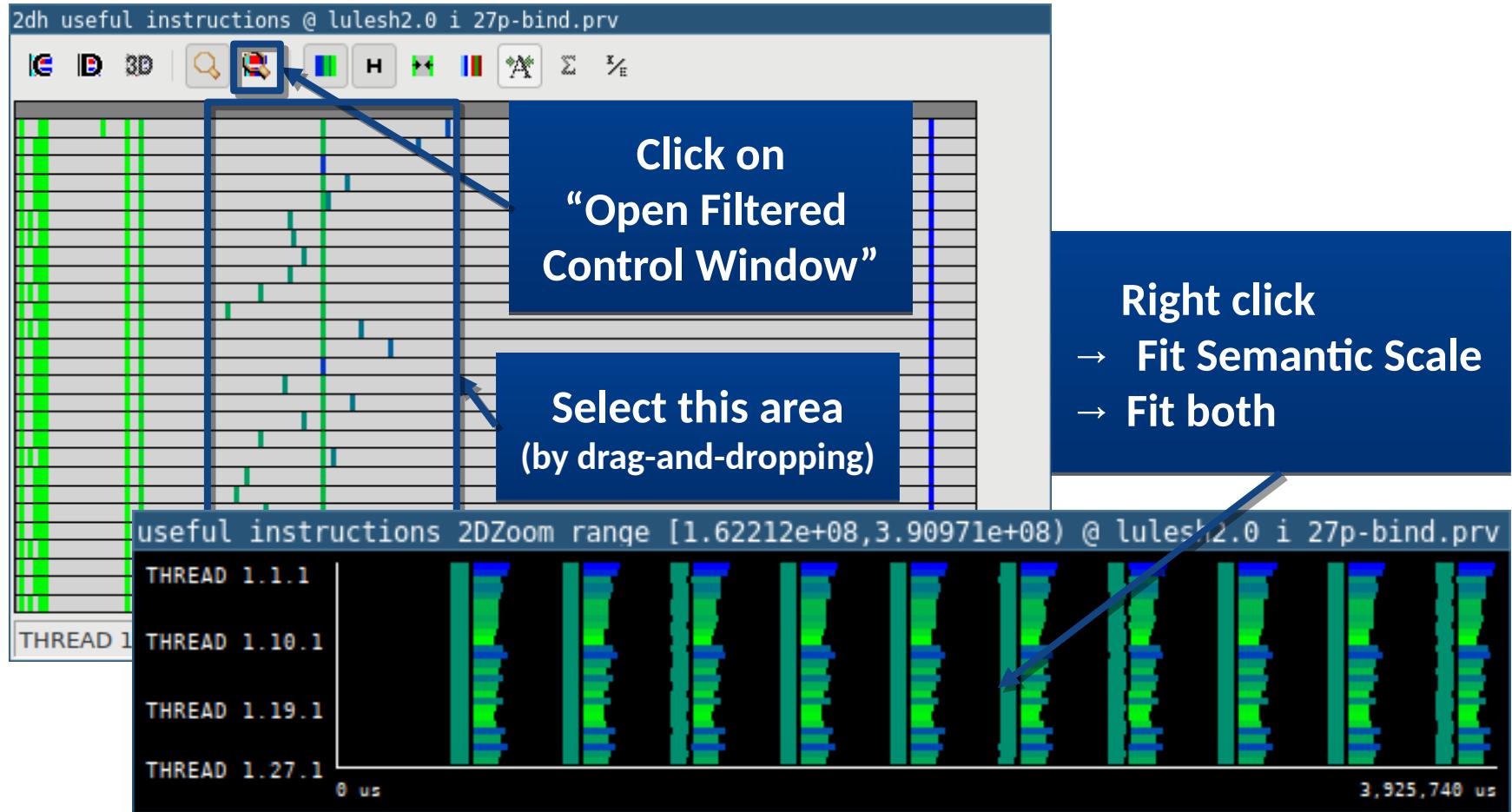
# Computation load distribution

- Click on “2dh\_useful\_instructions.cfg” (3rd link) ☐ Shows **amount of work**



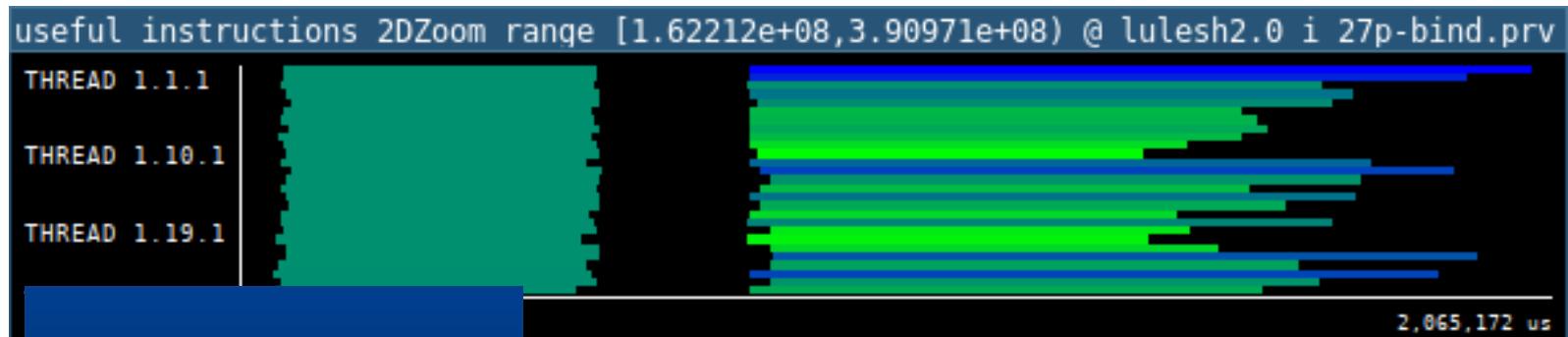
# Where does this happen?

- Go from the table to the timeline

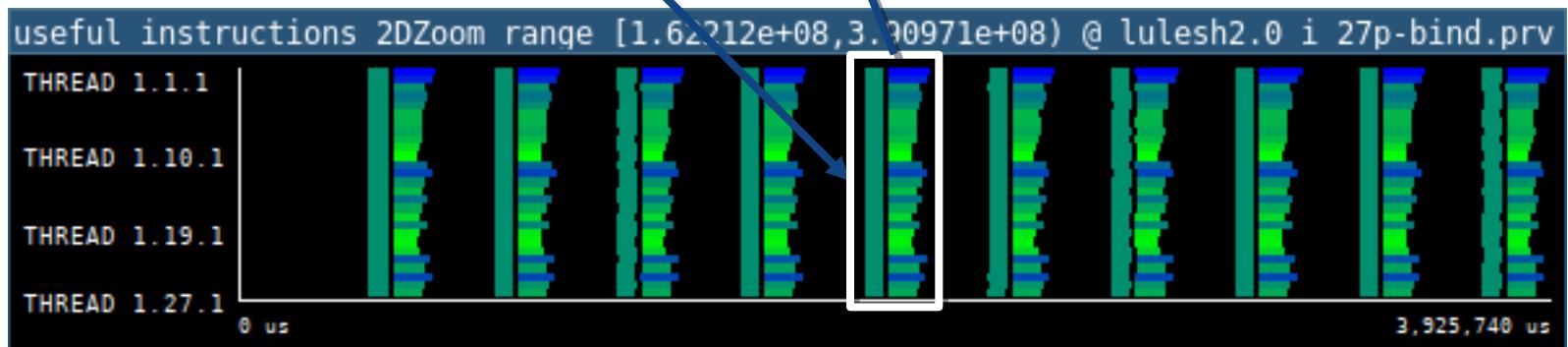


# Where does this happen?

- Slow & Fast at the same time? ☒ Imbalance

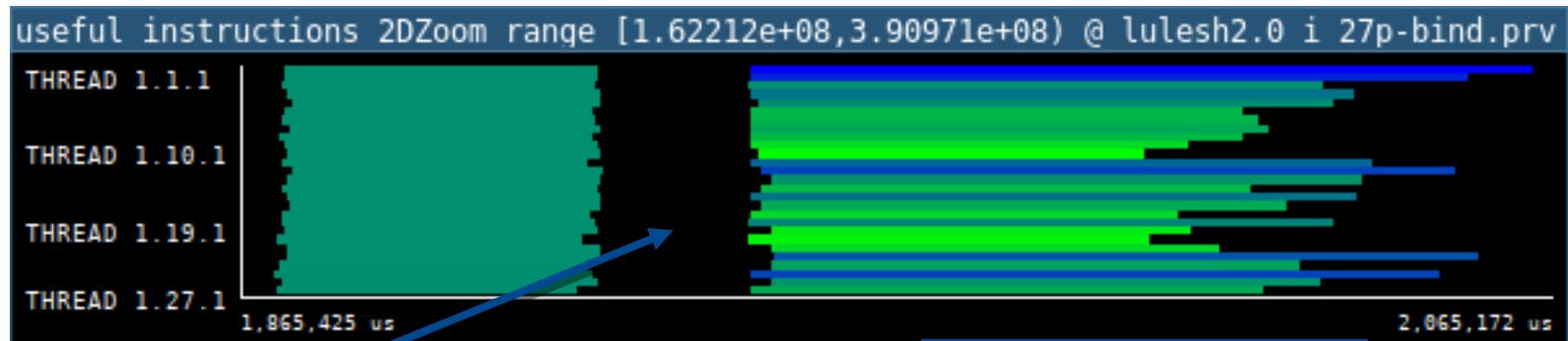


Zoom into  
1 of the iterations  
(by drag-and-dropping)



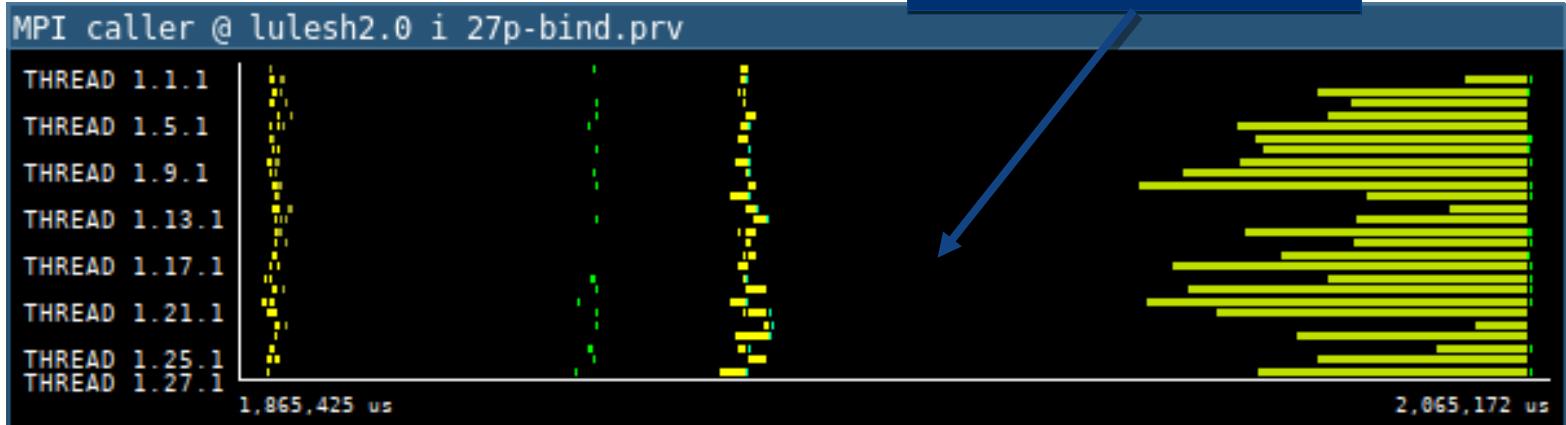
# Where does this happen?

- Hints → Call stack references → Caller function



Right click → Copy

Right click  
→ Paste → Time



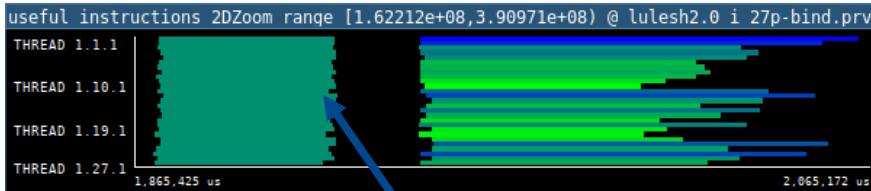
CommRecv

CommSend

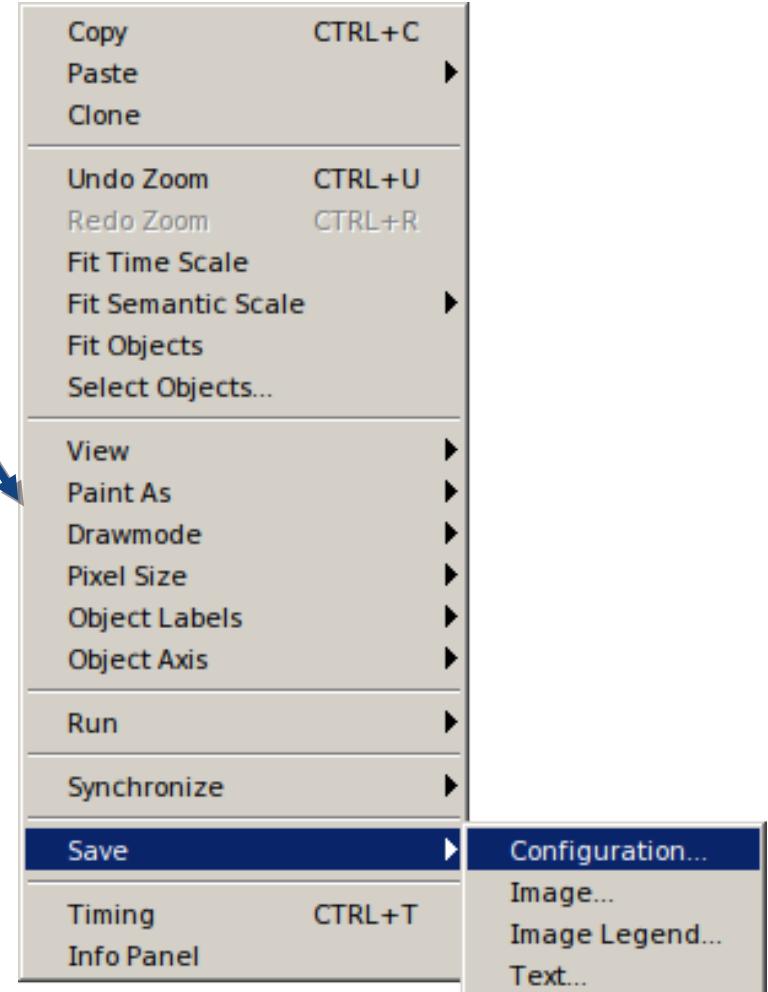
CommMonoQ

TimeIncrement

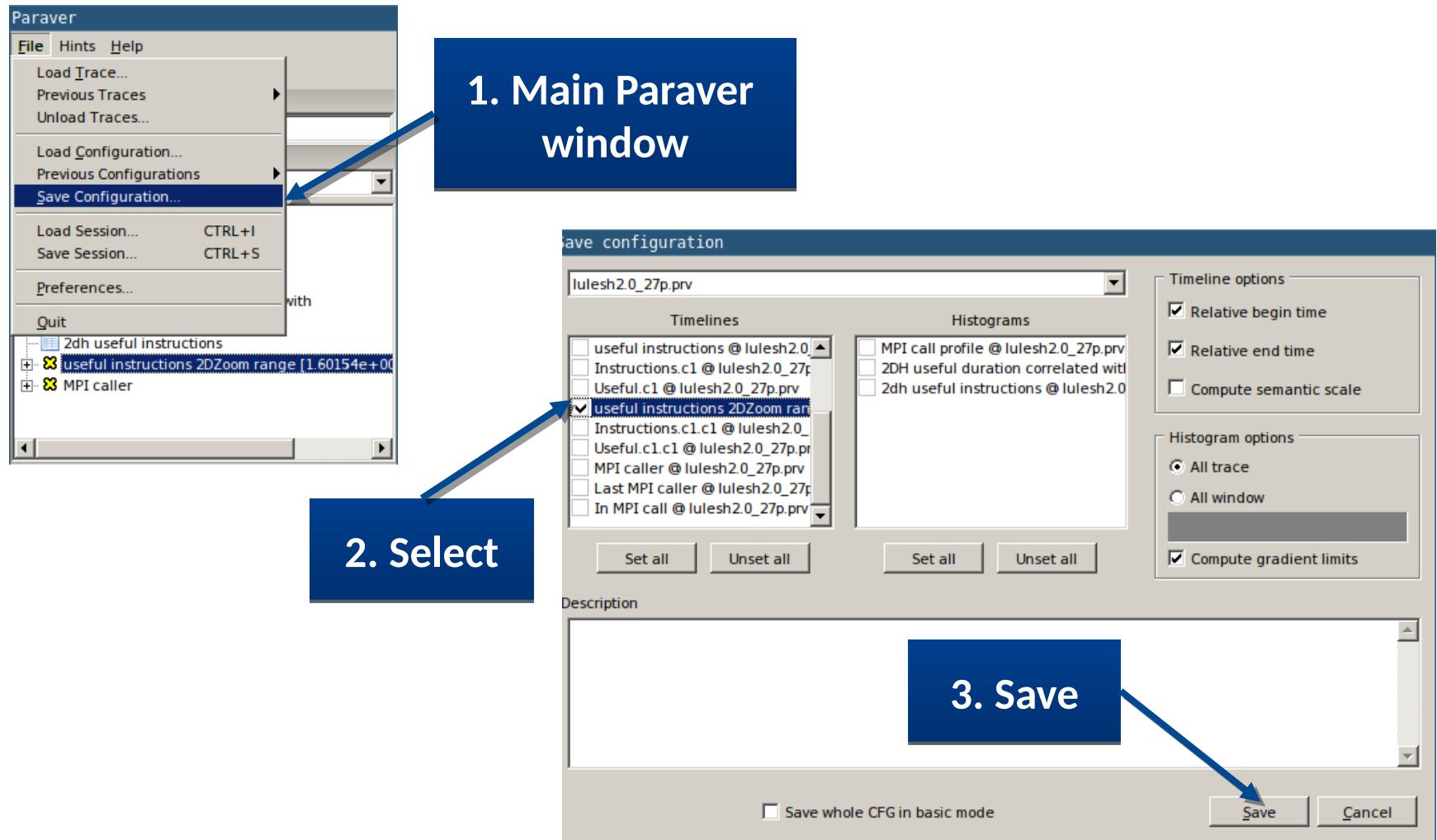
# Save CFG's (2 methods)



Right click on  
timeline

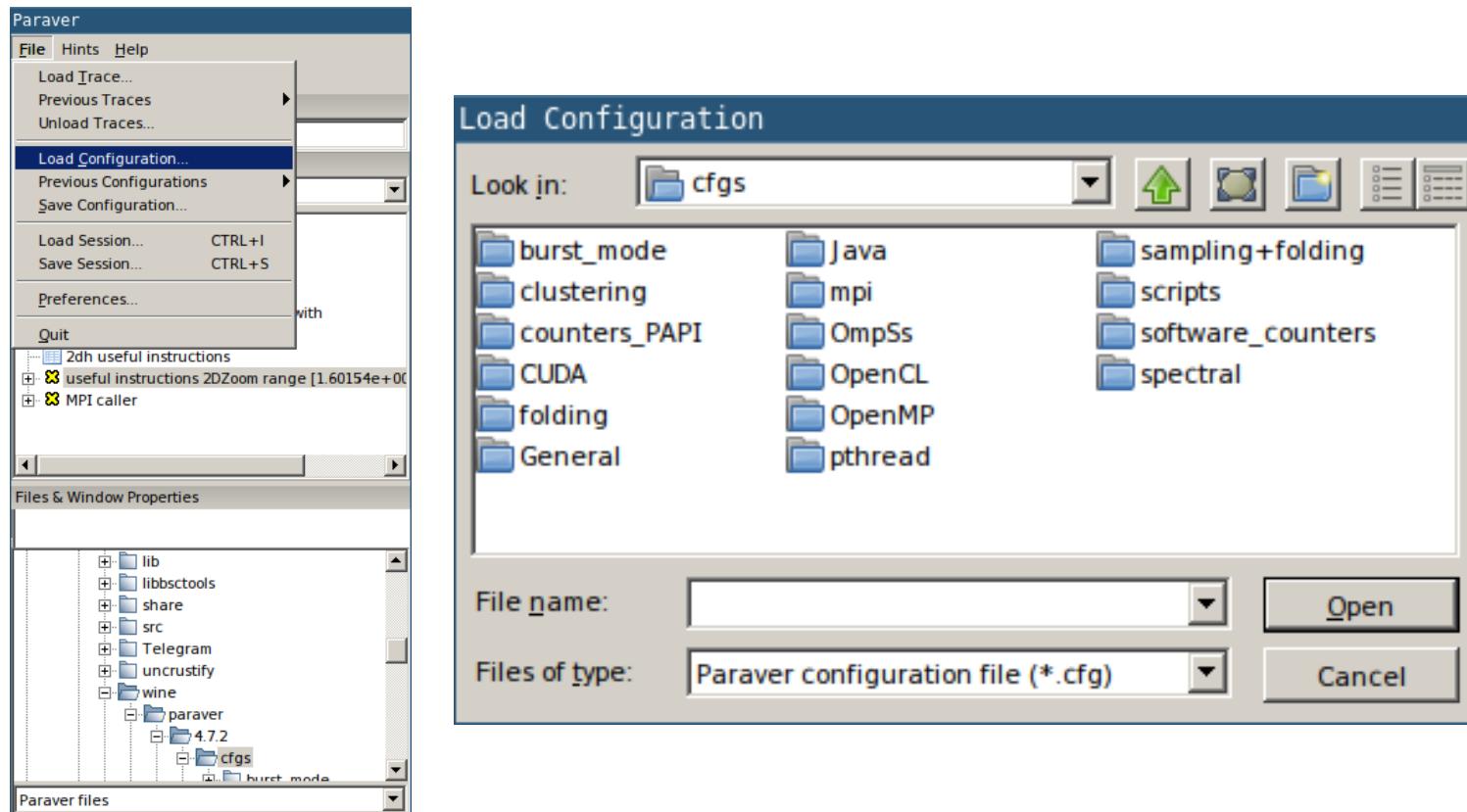


# Save CFG's (2 methods)



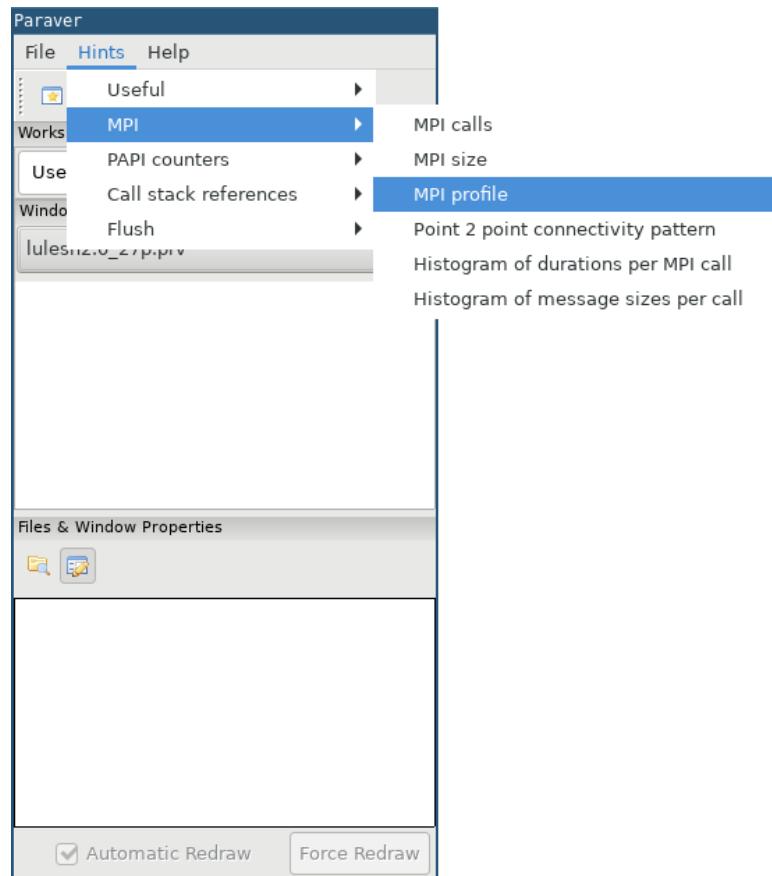
# CFG's distribution

- Paraver comes with many more included CFG's



# Hints: a good place to start!

- Paraver suggests CFG's based on the contents of the trace





**Barcelona  
Supercomputing  
Center**  
*Centro Nacional de Supercomputación*



EXCELENCIA  
SEVERO  
OCHOA

# Takeaway: Analyse efficiencies and unbalances

Judit Giménez, Germán Llort, Lau Mercadal,  
Sandra Méndez

✉ [tools@bsc.es](mailto:tools@bsc.es)

19/04/2021

POP Performance Analysis



**Barcelona  
Supercomputing  
Center**  
*Centro Nacional de Supercomputación*



EXCELENCIA  
SEVERO  
OCHOA

# Clustering Hands-On

Judit Giménez, Germán Llort, Lau Mercadal,  
Sandra Méndez

✉ [tools@bsc.es](mailto:tools@bsc.es)

19/04/2021

POP Performance Analysis

# Cluster-based analysis (I)

- Run the clustering tool on the trace you generated

```
mn$ module load clustering_suite
mn$ cd $HOME/tools-material/clustering
mn$ BurstClustering \
-d cluster.xml \
-i ../extrae/lulesh2.0_i_27p.prv \
-o lulesh2.0_i_27p_clustered.prv
```

- If you didn't get your own trace, use a prepared one from:

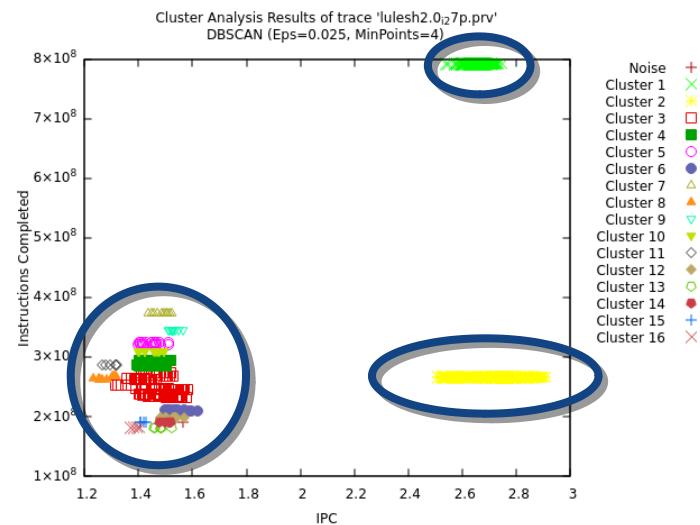
```
mn$ ls
$HOME/tools-material/traces/lulesh2.0_i_27p.prv
```

# Cluster-based analysis (II)

- Check the clustering scatter plot

```
mn$ gnuplot \
lulesh2.0_i_27p_clustered.IPC.PAPI_TOT_INS.gnuplo
t
```

- Identify main computing trends
- Work (Y) vs. Performance (X)
- Look at the clusters shape
  - Variability in both axes indicate **potential imbalances**



# Cluster-based analysis (III)

- Check the clustered trace
  - Copy the trace to your laptop

```
mn$ scp <USER>@mn1.bsc.es: \
tools-material/clustering/*.pcf,prv,row} $HOME
```

- Load with Paraver

```
laptop$ paraver/bin/wxparaver \
$HOME/lulesh2.0_i_27p_clustered.prv
```

- File ☰ Load configuration ☰ paraver/cfgs/clustering/clusterID\_window.cfg



# Cluster-based analysis (III)

- Correlate scatter plots & timelines to detect imbalances



Variable work  
and/or  
Variable speed  
+  
Simultaneously @ different processes  
=  
Imbalances

