

## Analysis report examination with CUBE

---

Christian Feld  
Jülich Supercomputing Centre

---



# CUBE

---

Parallel program analysis report exploration tools

- Libraries for XML report reading & writing
- Algebra utilities for report processing
- GUI for interactive analysis exploration
  - requires Qt4.6 or later

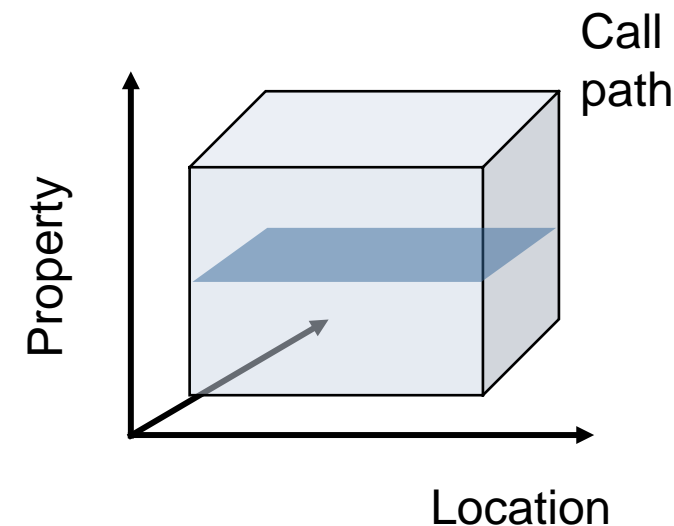
Originally developed as part of Scalasca toolset

Now available as a separate component

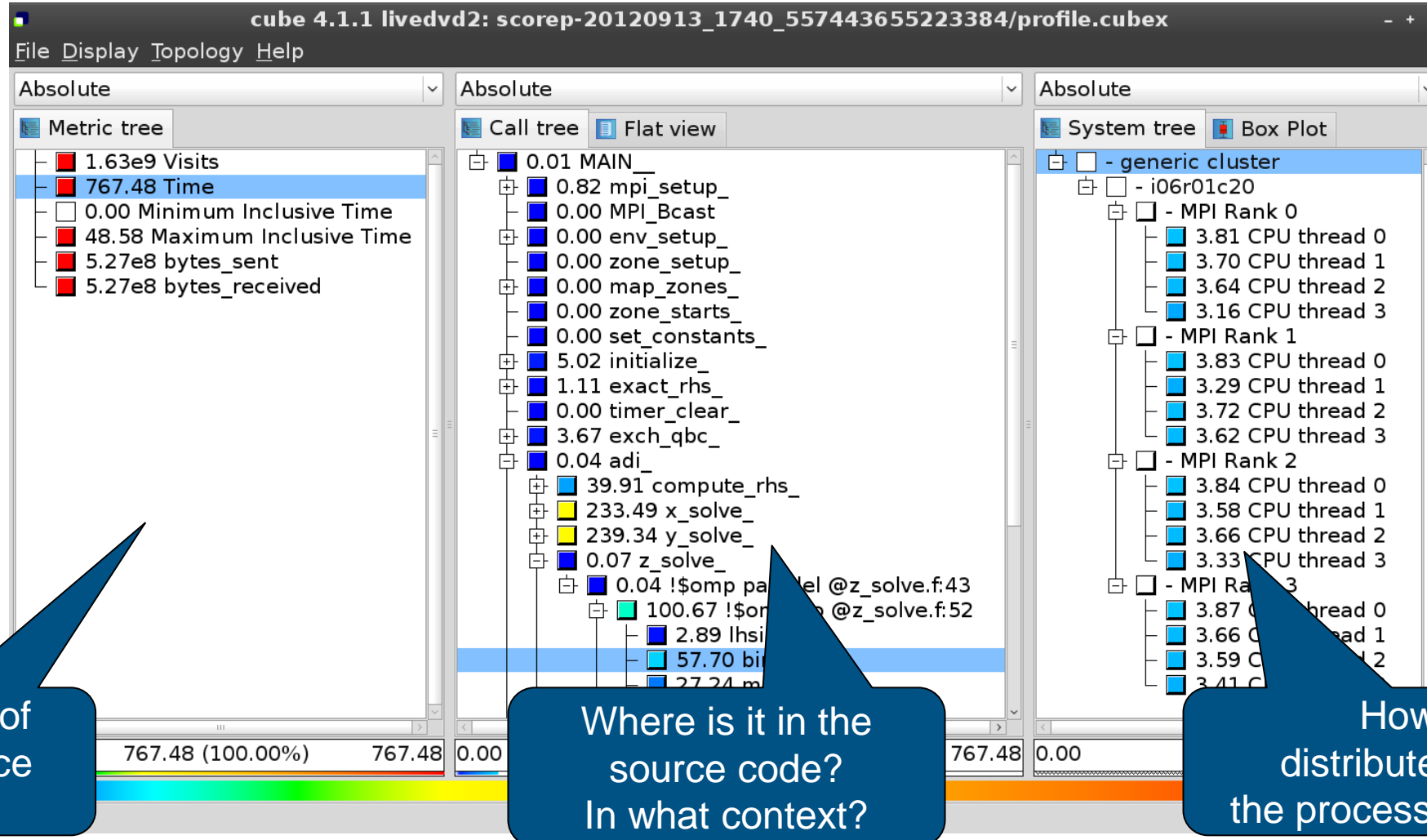
- Can be installed independently of Score-P, e.g., on laptop or desktop
- Latest release: CUBE 4.3.4 (April 2016)

# Analysis presentation and exploration

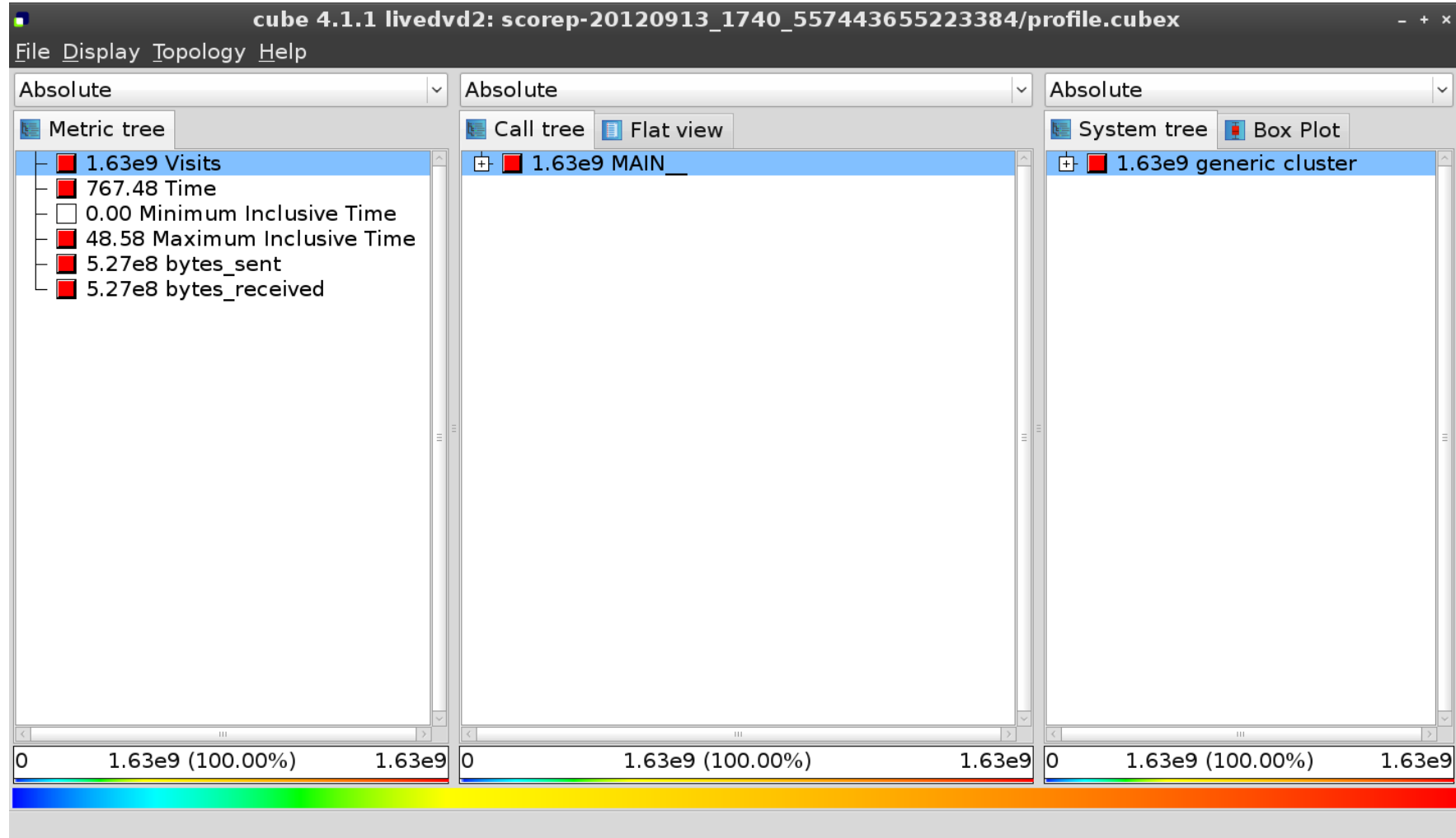
- Representation of values (severity matrix) on three hierarchical axes
  - Performance property (metric)
  - Call path (program location)
  - System location (process/thread)
- Three coupled tree browsers
- CUBE displays severities
  - As value: for precise comparison
  - As colour: for easy identification of hotspots
  - Inclusive value when closed & exclusive value when expanded
  - Customizable via display modes



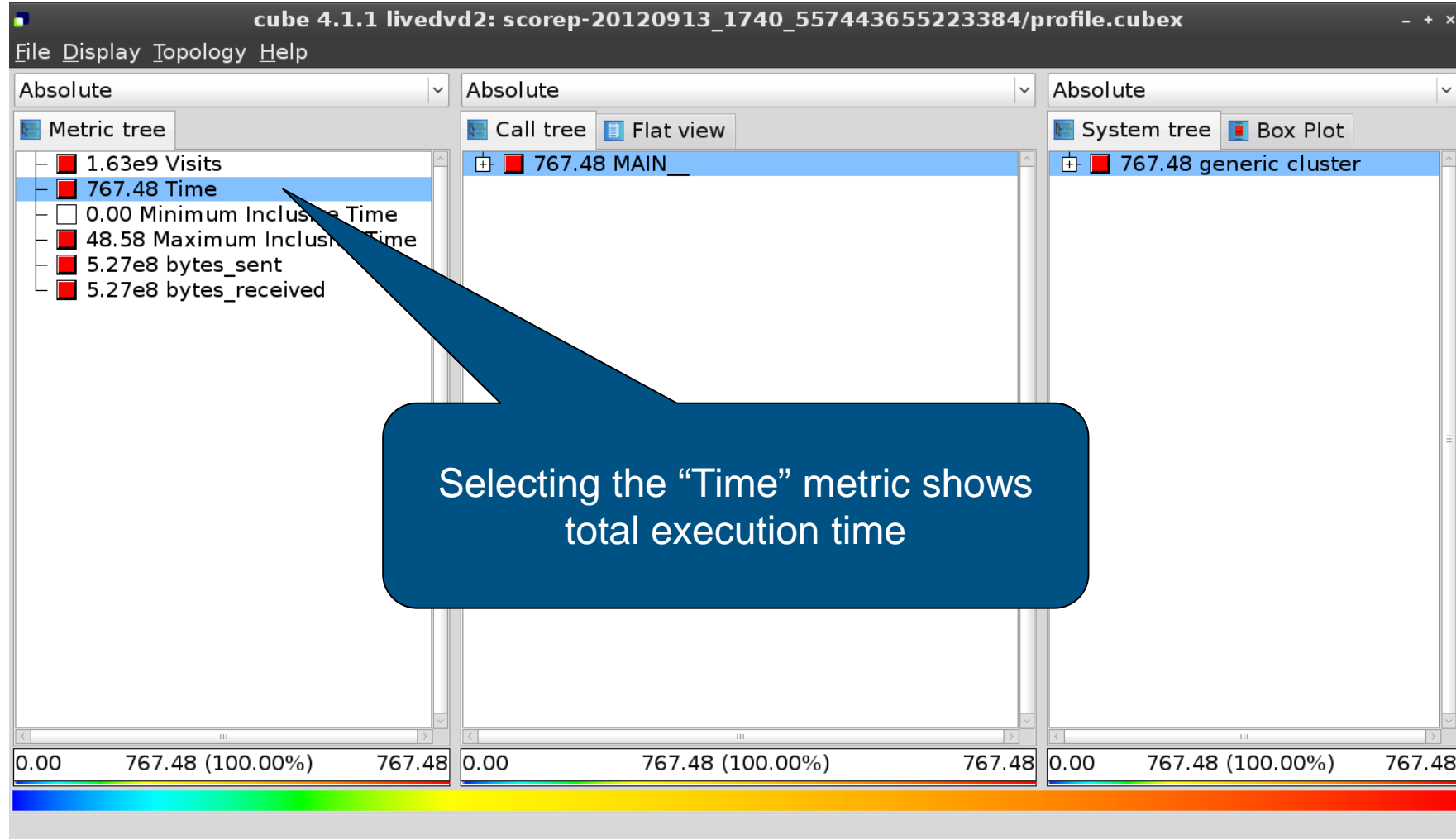
# Analysis presentation



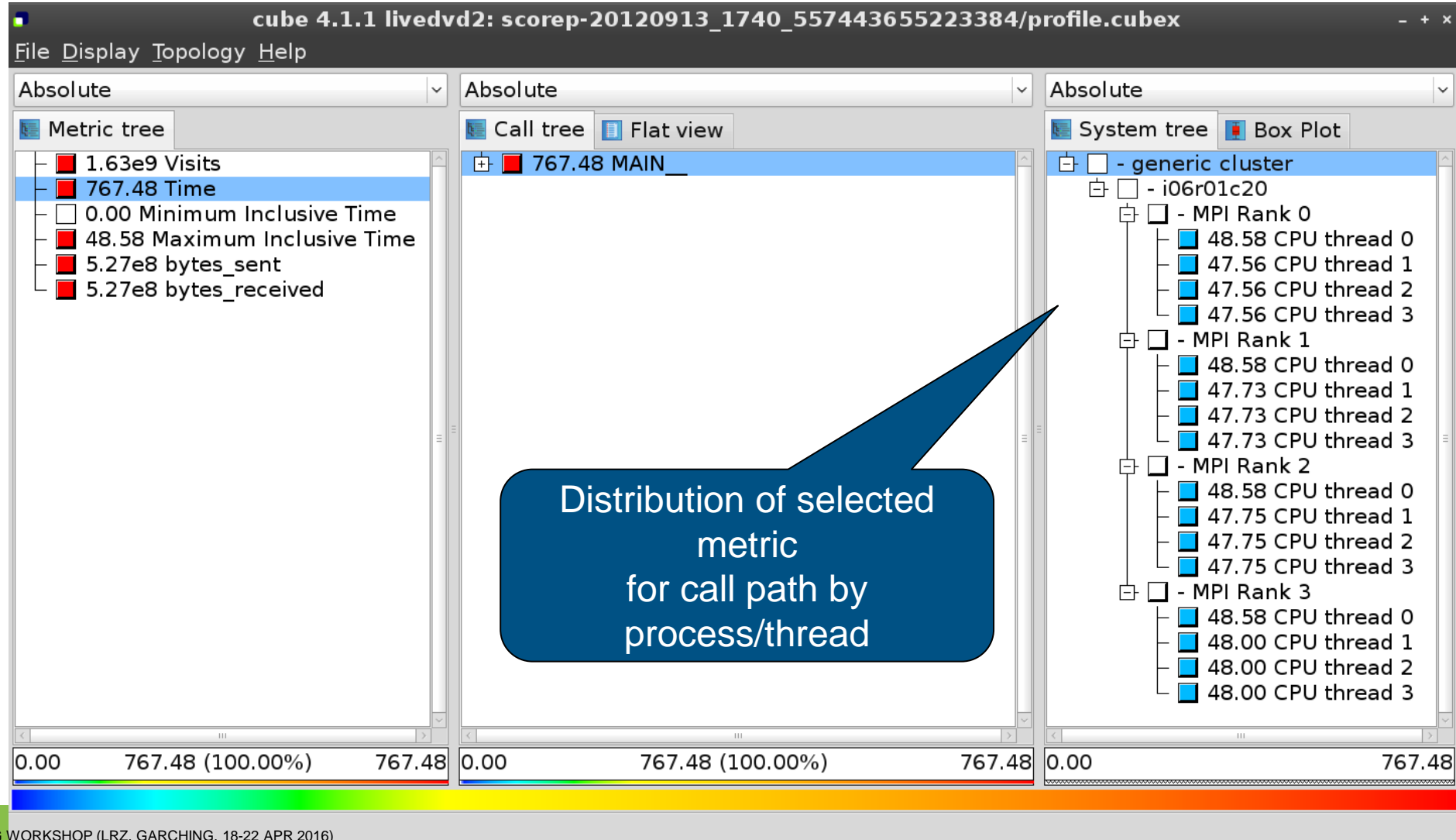
# Analysis report exploration (opening view)



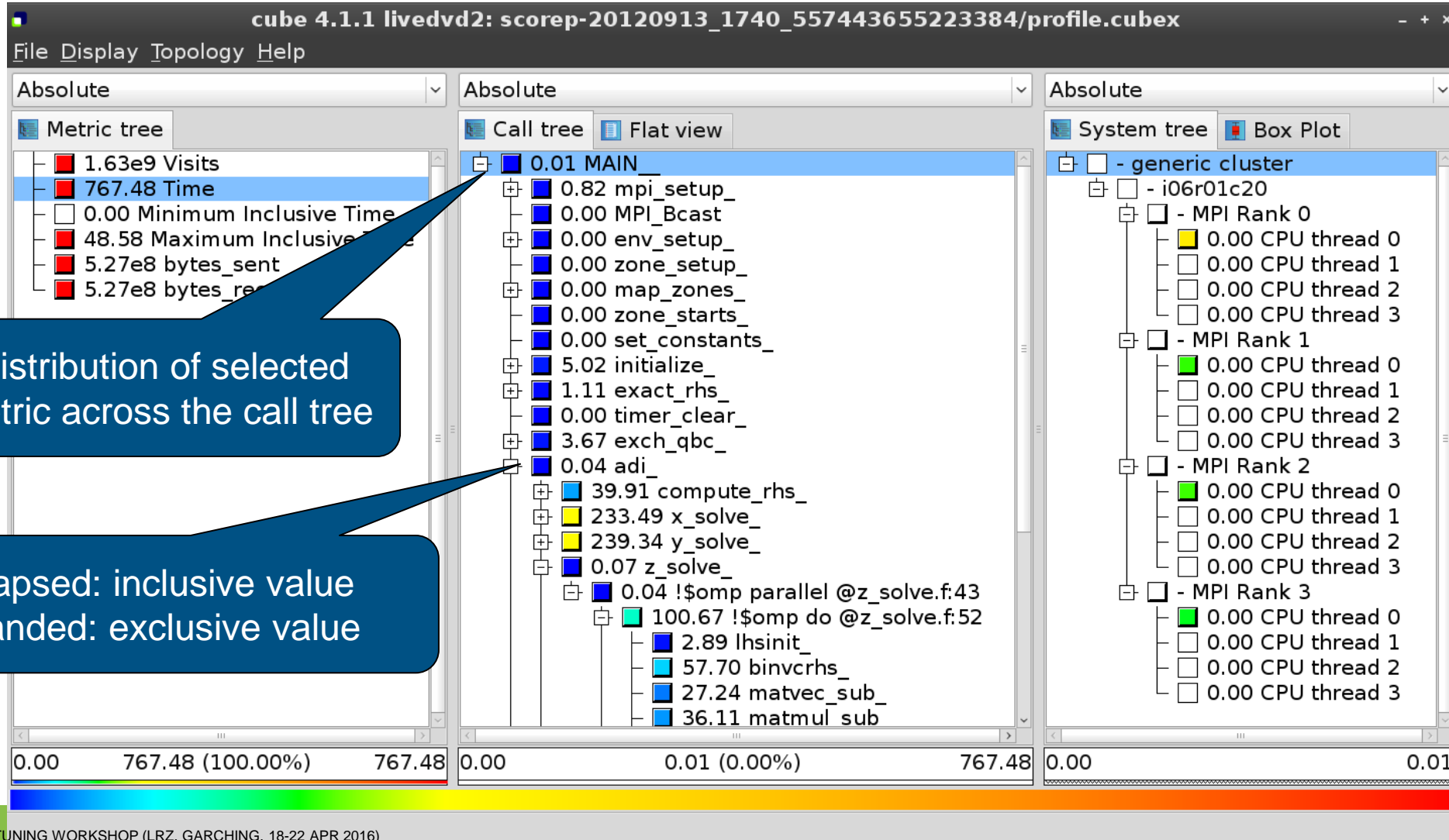
# Metric selection



# Expanding the system tree



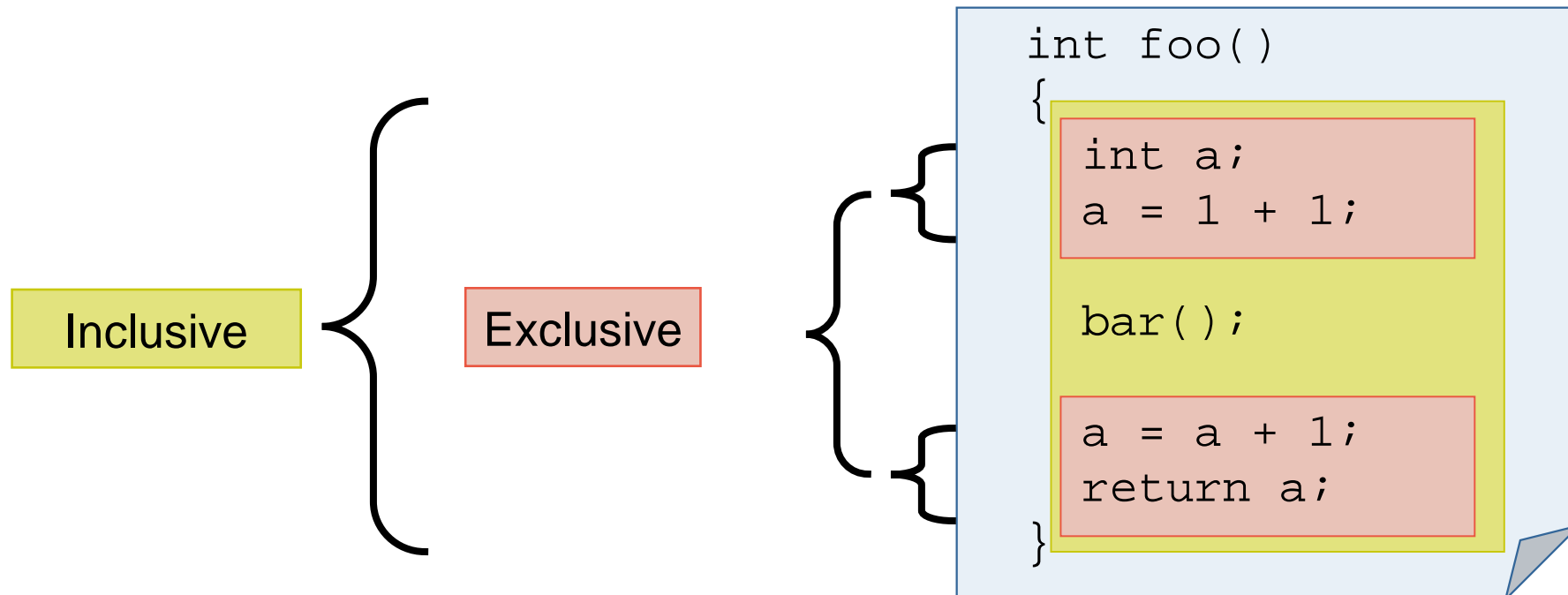
# Expanding the call tree



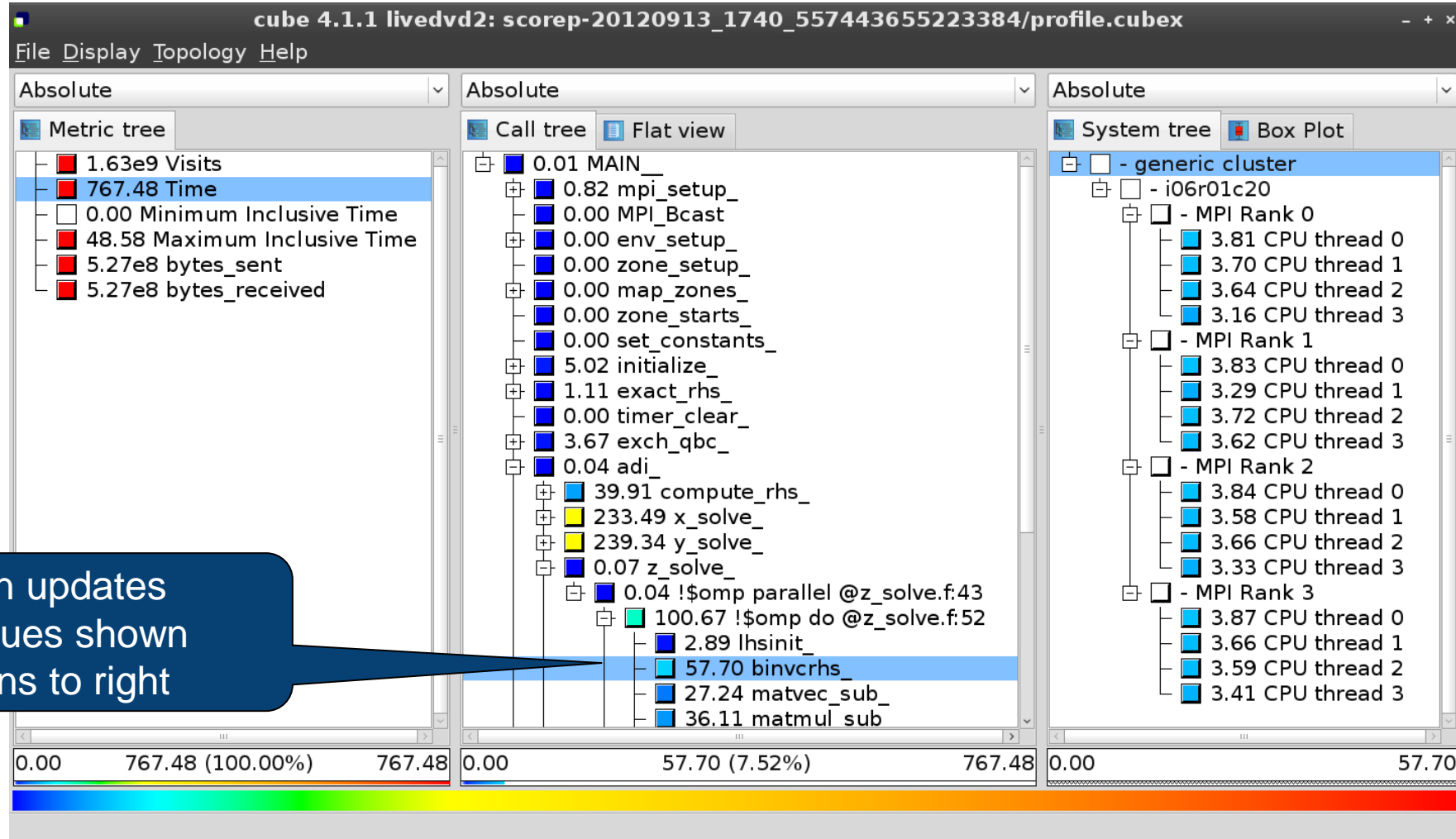


# Inclusive vs. Exclusive values

- Inclusive
  - Information of all sub-elements aggregated into single value
- Exclusive
  - Information cannot be subdivided further



# Selecting a call path

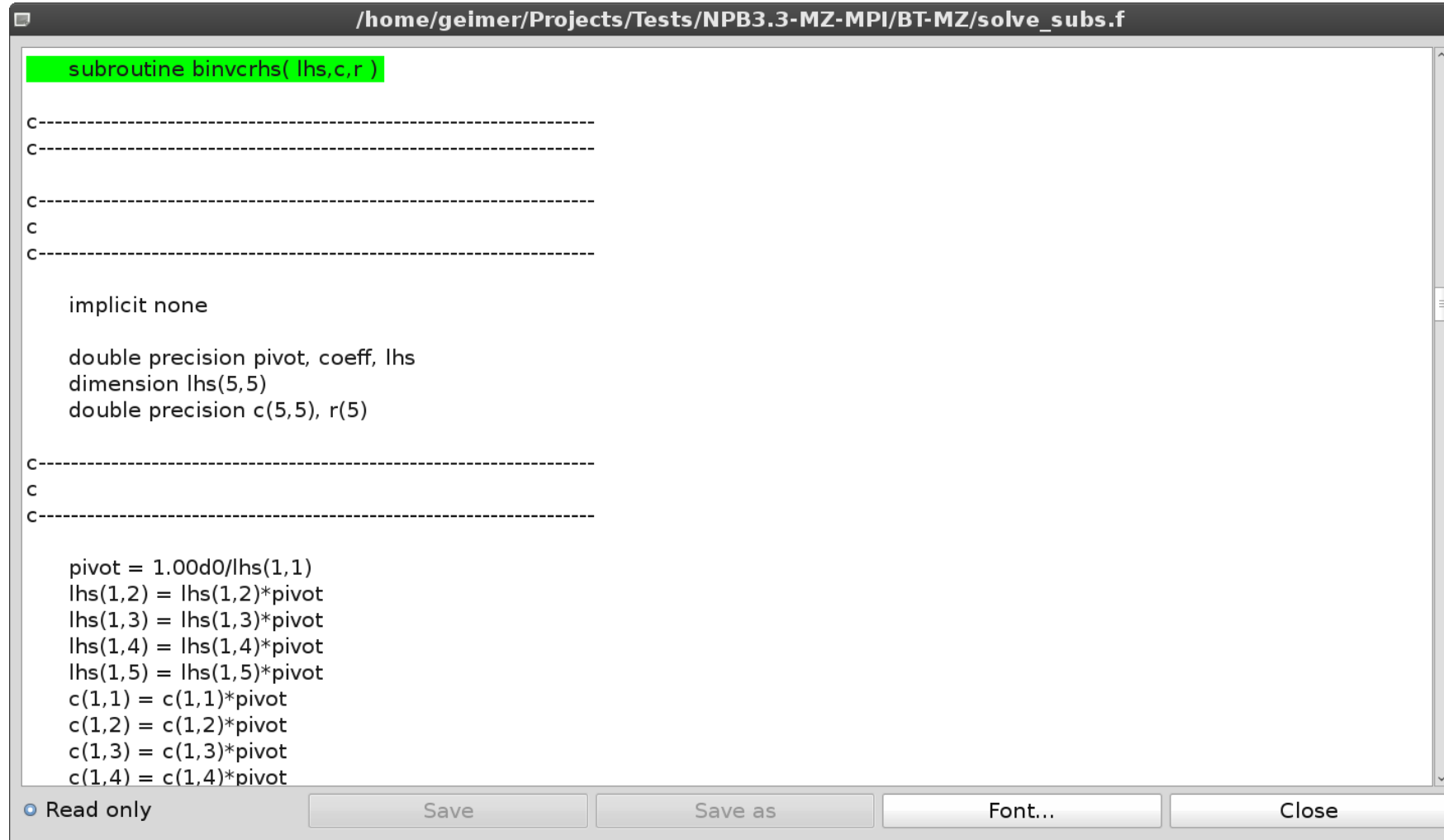


# Source-code view via context menu

The screenshot displays the 'cube 4.1.1 livedvd2: scorep-20120913\_1740\_557443655223384/profile.cubex' application. It features three main panels: 'Metric tree', 'Call tree', and 'System tree'. The 'Call tree' panel is in 'Flat view' and shows a hierarchical list of functions. A context menu is open over the '57.70 binvcrhs' node, listing options such as 'Call site', 'Called region', 'Expand/collapse', 'Hiding', 'Cut call tree', 'Find items', 'Find Next', 'Clear found items', 'Copy to clipboard', and 'Min/max values'. The 'Source code' option is highlighted. A blue callout box with a white border points to the context menu, containing the text 'Right-click opens context menu'. At the bottom of the application, a status bar shows the selected item's value and percentage: '57.70 (7.52%)' out of a total of '767.48'. A legend below the status bar indicates that this view 'Shows the source code of the clicked item'.

| Metric         | Value  | Percentage |
|----------------|--------|------------|
| 767.48 Time    | 767.48 | 100.00%    |
| 57.70 binvcrhs | 57.70  | 7.52%      |

## Source-code view



```
/home/geimer/Projects/Tests/NPB3.3-MZ-MPI/BT-MZ/solve_subs.f
subroutine binvcrhs( lhs,c,r )
C-----
C-----
C-----
C
C-----

implicit none

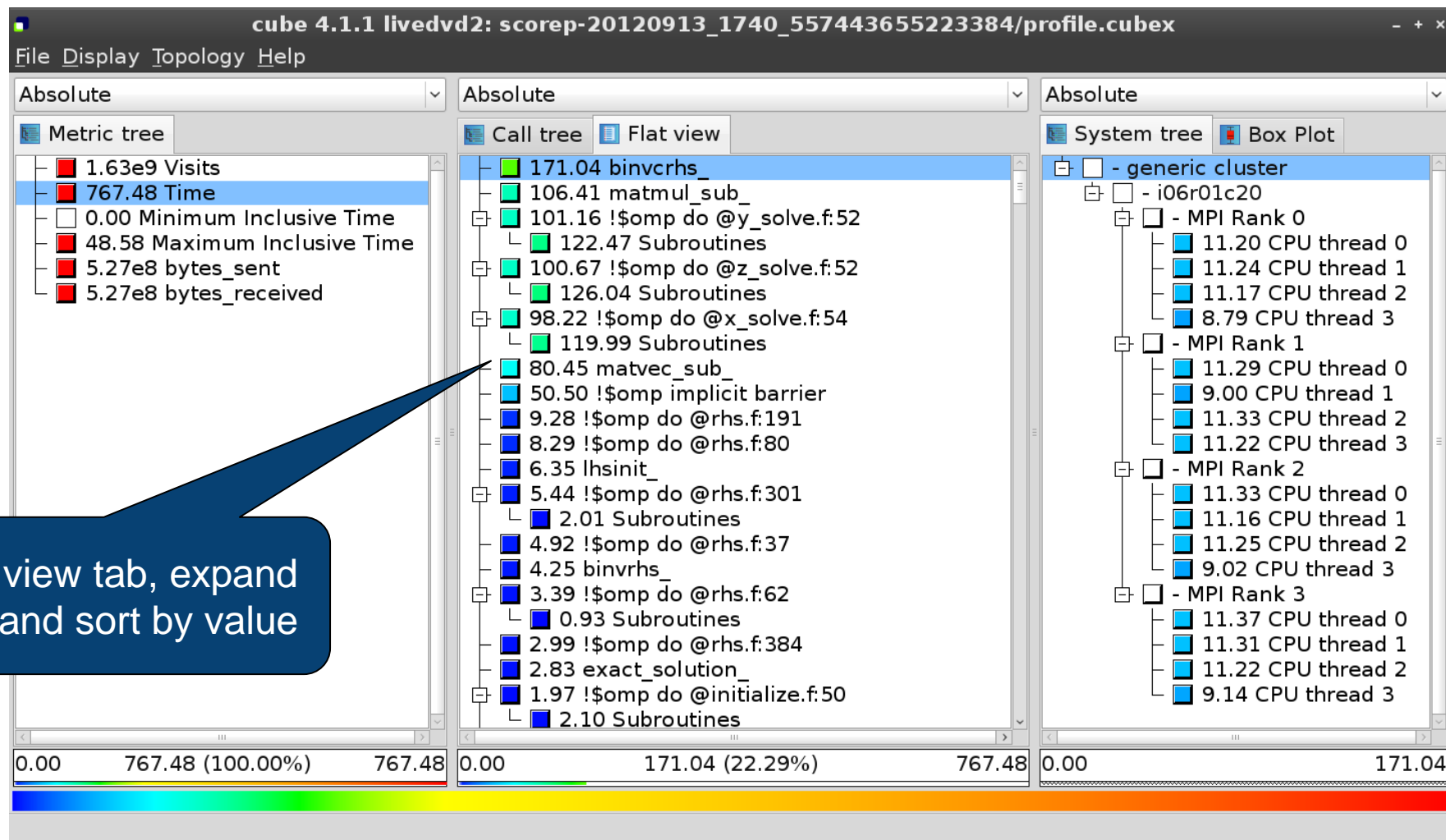
double precision pivot, coeff, lhs
dimension lhs(5,5)
double precision c(5,5), r(5)

C-----
C
C-----

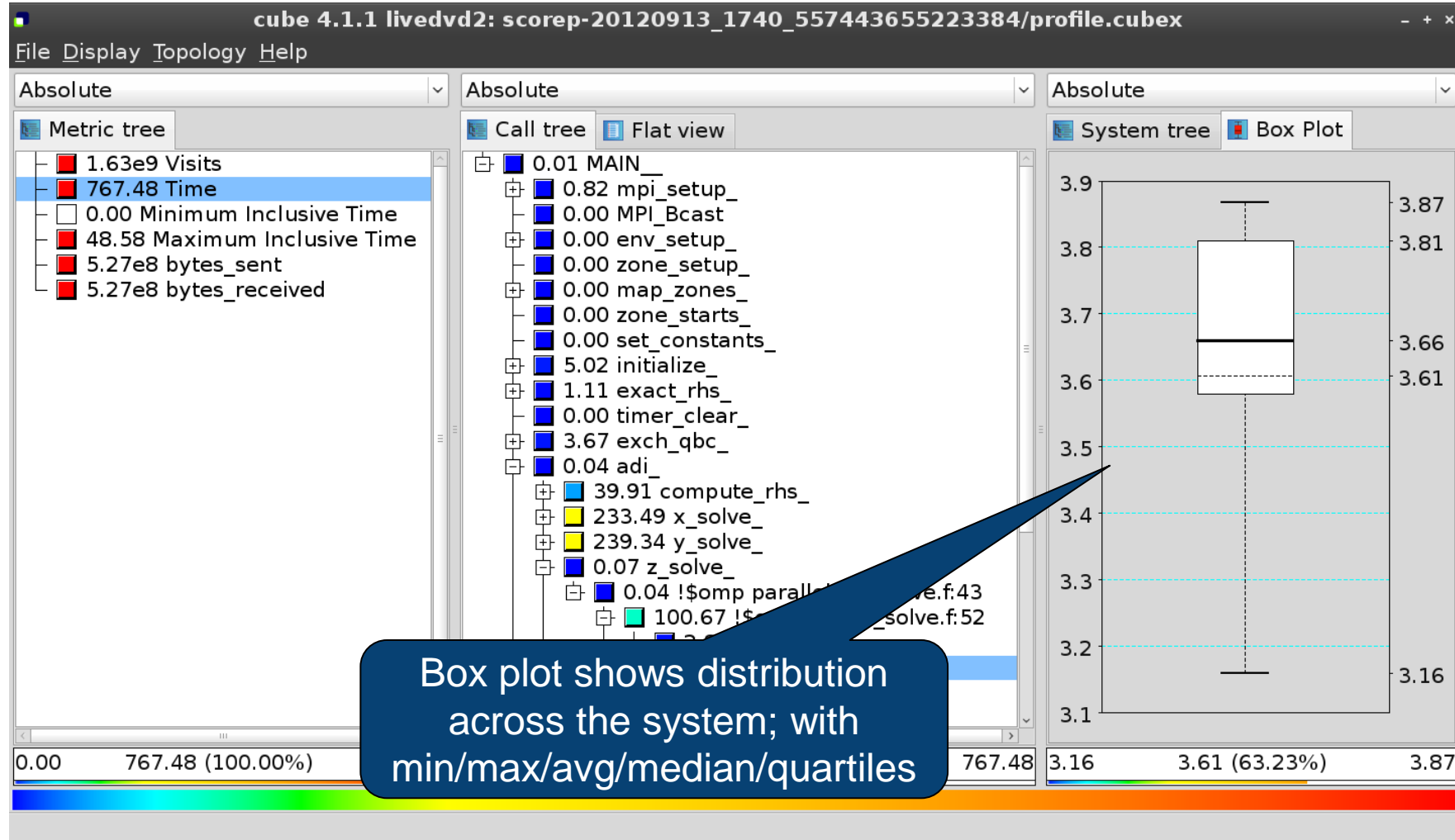
pivot = 1.00d0/lhs(1,1)
lhs(1,2) = lhs(1,2)*pivot
lhs(1,3) = lhs(1,3)*pivot
lhs(1,4) = lhs(1,4)*pivot
lhs(1,5) = lhs(1,5)*pivot
c(1,1) = c(1,1)*pivot
c(1,2) = c(1,2)*pivot
c(1,3) = c(1,3)*pivot
c(1,4) = c(1,4)*pivot
```

Read only    Save    Save as    Font...    Close

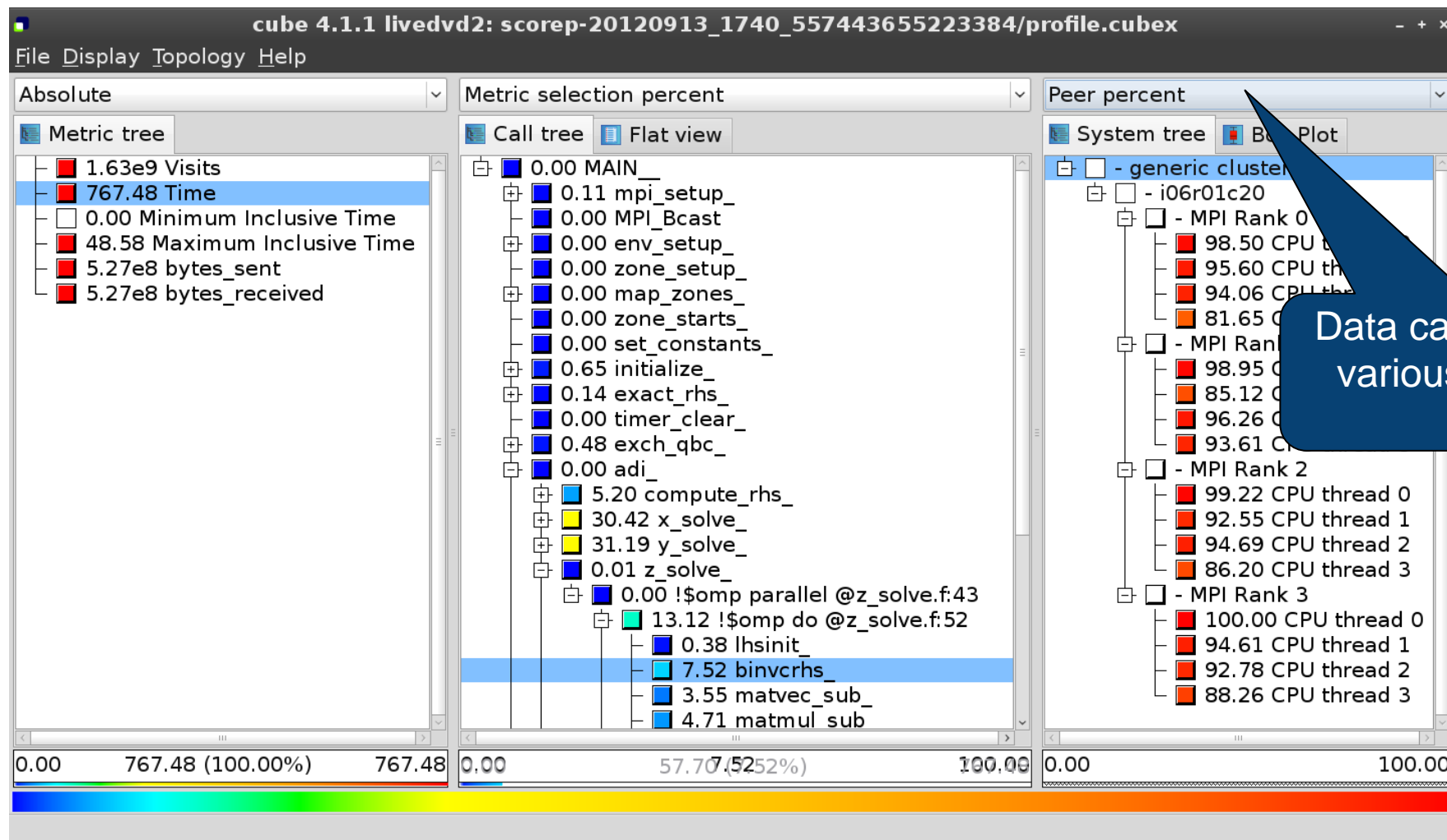
# Flat profile view



# Box plot view



# Alternative display modes



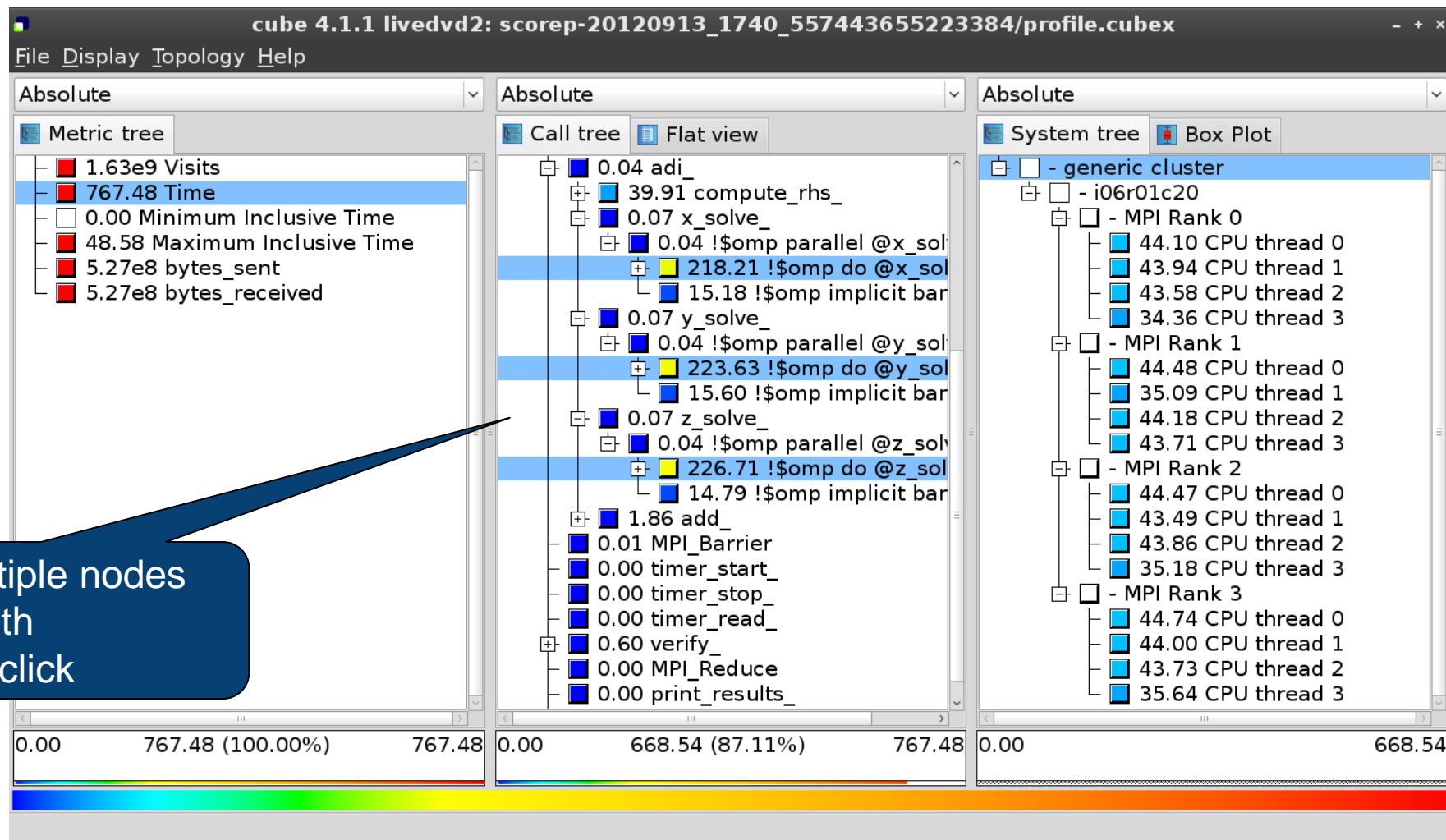
# Important display modes

---

- Absolute
  - Absolute value shown in seconds/bytes/counts
- Selection percent
  - Value shown as percentage w.r.t. the selected node “on the left” (metric/call path)
- Peer percent (system tree only)
  - Value shown as percentage relative to the maximum peer value



# Multiple selection



## Derived metrics in Cube

---

- Value of the derived metric is not stored, but **calculated** on-the-fly
- One defines an CubePL expression, e.g.:  
**metric::time(i)/metric::visits(e)**
- Types of derived metrics:
  - **Prederived**: evaluation of the CubePL expression is done before the aggregation
  - **Postderived**: evaluation of the CubePL expression is performed after the aggregation
- Examples:
  - “Average execution time” Postderived metric with an expression:  
**metric::time(i)/metric::visits(e)**
  - “Number of FLOP per second” Postderived metric with an expression:  
**metric::FLOP()/metric::time()**

# Derived metrics in Cube GUI

Collection of derived metrics

Parameters of the derived metric

CubePL expression

1.09e8 Visits (occ)  
1.01e6 Time (sec)  
0.00 Minimum Inclusive Time (sec)  
246.14 Maximum Inclusive Time (sec)  
7.18e12 bytes\_sent  
7.18e12 bytes\_received

0.35 main(int, char \*D)  
2512.10 ughellinit  
1.01e6 ug::script::LoadUGScript(const char \*, bool)  
2.11 ug::script::ParseBuffer(const char \*, const char \*)  
0.04 ughellFinalize  
94.31 MPI\_Finalize

- machine Blue Gene/Q  
- rack 11  
- midplane 1  
- nodeboard 8  
- nodecard 4  
0.65 MPI Rank 0  
0.64 MPI Rank 1

Select metric from collection: Average execution time (kenobi)

Derived metric type: Postderived metric

Display name: Average visit time

Unique name: avg\_visit\_time

Data type: DOUBLE

Unit of measurement: sec

URL:

Description:  
Calculates average time of region execution per visit.  
Autor is Michael Knobloch.

Calculation Calculation Init Aggregation "+" Aggregation "-"

metric::time(i)metric::visits(e)

Create metric Cancel

Share this metric with SCALASCA group

# Example derived metric FLOPS based on PAPI\_FP\_OPS and time

The image displays the Cube-4.3.1 interface for configuring and visualizing a derived metric. On the left, the 'Edit metric FLOPS (on froggy1)' dialog is open, showing the configuration for a post-derived metric named 'FLOPS' with a unique name 'flops' and a data type of 'DOUBLE'. The metric is calculated as `metric::PAPI_FP_OPS()/metric::time()`. The main window shows three views: 'Metric tree', 'Call tree', and 'System tree'. The 'Metric tree' view shows a list of metrics, with '1.84e9 FLOPS' highlighted. The 'Call tree' view shows a hierarchical breakdown of the metric, with '9.65e8 !\$omp do @exact\_r...' highlighted. The 'System tree' view shows the system hierarchy, with 'machine Linux' highlighted. The bottom of the interface shows a color bar and a selected element: 'Selected !\$omp do @exact\_rhs.f:46'.

**Edit metric FLOPS (on froggy1)**

Select metric from collection: --- please select ---

Derived metric type: Postderived metric

Display name: FLOPS

Unique name: flops

Data type: DOUBLE

Unit of measurement:

URL:

Description:

Calculation: `metric::PAPI_FP_OPS()/metric::time()`

Share this metric with SCALASCA group

**Cube-4.3.1: scorep\_8x4\_sum/profile.cubex (on froggy1)**

File Display Plugins Help

Restore Setting Save Settings

**Metric tree**

- 1.17e7 Visits (occ)
- 1148.49 Time (sec)
- 0.00 Minimum Inclusive Time (sec)
- 41.57 Maximum Inclusive Time (...)
- 0 bytes\_put (bytes)
- 0 bytes\_get (bytes)
- 5.75e12 PAPI\_TOT\_INS (#)
- 2.69e12 PAPI\_TOT\_CYC (#)
- 2.12e12 PAPI\_FP\_OPS (#)
- 3.12e9 bytes\_sent (bytes)
- 3.12e9 bytes\_received (bytes)
- 1.84e9 FLOPS**

**Call tree**

- 3.17e5 MAIN\_
  - 7.04e5 mpi\_setup\_
    - 6.34e4 MPI\_Bcast
    - 2.05e5 env\_setup\_
      - 7.39e5 zone\_setup\_
        - 9.31e5 map\_zones\_
          - 9.39e4 zone\_starts\_
            - 6.16e5 set\_constants\_
              - 5.91e8 initialize\_
                - 0.00 exact\_rhs\_
                  - 145.62 !\$omp parallel @exac...
                    - 2.54e4 !\$omp do @exact\_r...
                      - 9.65e8 !\$omp do @exact\_r...**
                      - 9.62e8 !\$omp do @exact\_r...
                      - 8.14e8 !\$omp do @exact\_r...
                      - 1.21e5 !\$omp do @exact\_r...
                      - 0.00 !\$omp implicit barrier...
                    - 6.23e4 exch\_qbc\_
                      - 1.94e9 adi\_
                        - 2.19e5 MPI\_Barrier
                        - 1.92e9 <<bt\_iter>> (200 itera...
                        - 1.98e8 verify\_
                          - 1.05e5 MPI\_Reduce

**System tree**

      - machine Linux
        - node frog6
          - MPI Rank 0
            - 1.17e9 Master thread
            - 9.43e8 OMP thread 1
            - 9.47e8 OMP thread 2
            - 9.47e8 OMP thread 3
          - MPI Rank 1
            - 1.17e9 Master thread
            - 9.87e8 OMP thread 1
            - 9.68e8 OMP thread 2
            - 9.72e8 OMP thread 3
          - MPI Rank 2
            - 1.10e9 Master thread
            - 8.97e8 OMP thread 1
            - 8.77e8 OMP thread 2
            - 8.76e8 OMP thread 3
          - MPI Rank 3
            - 1.09e9 Master thread
            - 9.06e8 OMP thread 1
            - 9.04e8 OMP thread 2
            - 9.02e8 OMP thread 3

All (32 elements)

0.00 1.84e9 (100.00%) 1.84e9 0.00 9.65e8 (-0.00%) -12858016489314434.00 0.00... -179769313486231570814527423731704356798070...

Selected !\$omp do @exact\_rhs.f:46

# Context-sensitive help

The screenshot displays the 'cube 4.1.1' application window with the title bar 'cube 4.1.1 livedvd2: scorep-20120913\_1740\_557443655223384/profile.cubex'. The 'Help' menu is open, showing options: 'Getting started', 'Mouse and keyboard control', 'What's This? (Shift+F1)', and 'About'. The 'What's This?' option is selected, and a tooltip is visible over the 'Metric tree' panel. The tooltip contains the following text: 'Selected metrics description' and 'Selected regions description'. The 'Metric tree' panel shows a list of metrics, with '767.48 Time' selected. The 'System tree' panel shows a hierarchical view of the system, including 'generic cluster', 'i06r01c20', and four MPI Ranks, each with four CPU threads. The status bar at the bottom shows 'Change into help mode for display components'.

Context-sensitive help available for all GUI items

## CUBE algebra utilities

---

- Extracting solver call-tree from analysis report

```
% cube_cut -r '<<ITERATION>>' scorep_bt-mz_B_8x8_sum/profile.cubex  
Writing cut.cubex... done.
```

- Extracting partition of system-tree processes from analysis report

```
% cube_part -R 0-2,4,6- scorep_bt-mz_B_8x8_sum/profile.cubex  
Writing part.cubex... done.
```

- Calculating difference of two reports

```
% cube_diff scorep_bt-mz_B_8x8_sum/profile.cubex cut.cubex  
Writing diff.cubex... done.
```

- Additional utilities for merging, calculating mean, etc.
- Default output of `cube_utility` is a new report `utility.cubex`
- Further utilities for report scoring & statistics
- Run utility with “-h” (or no arguments) for brief usage info

# Loop Unrolling

---

- Show time dependent behavior by unrolling iterations

- Preparations:

- Mark loops by using Score-P user instrumentation in your source code

```
SCOREP_USER_REGION_BEGIN( scorep_bt_loop, "<<bt_iter>>", SCOREP_USER_REGION_TYPE_DYNAMIC )
```

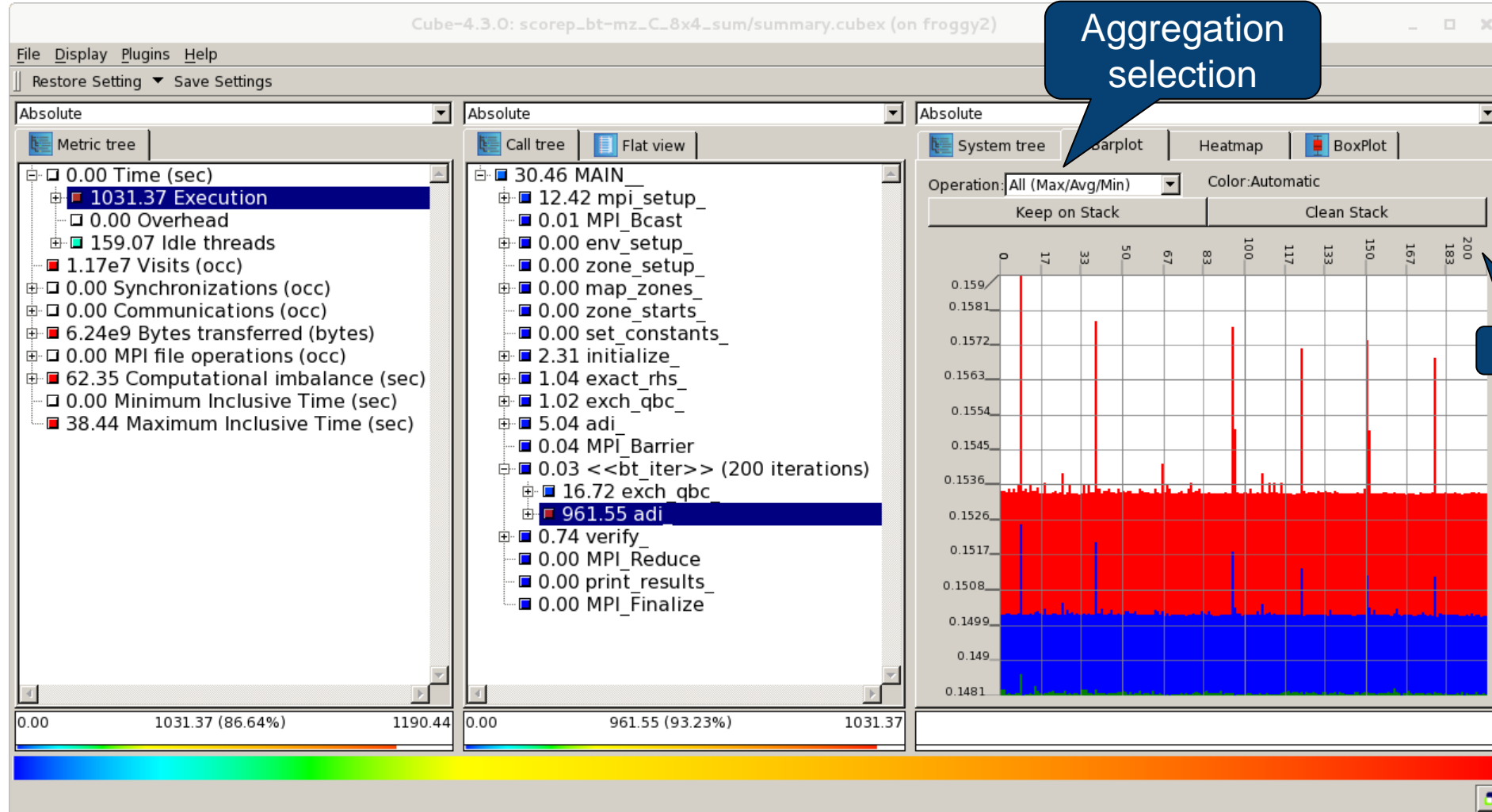
- Result in the CUBE profile:

- Iterations shown as separate call trees
  - Useful for checking results for specific iterations

or

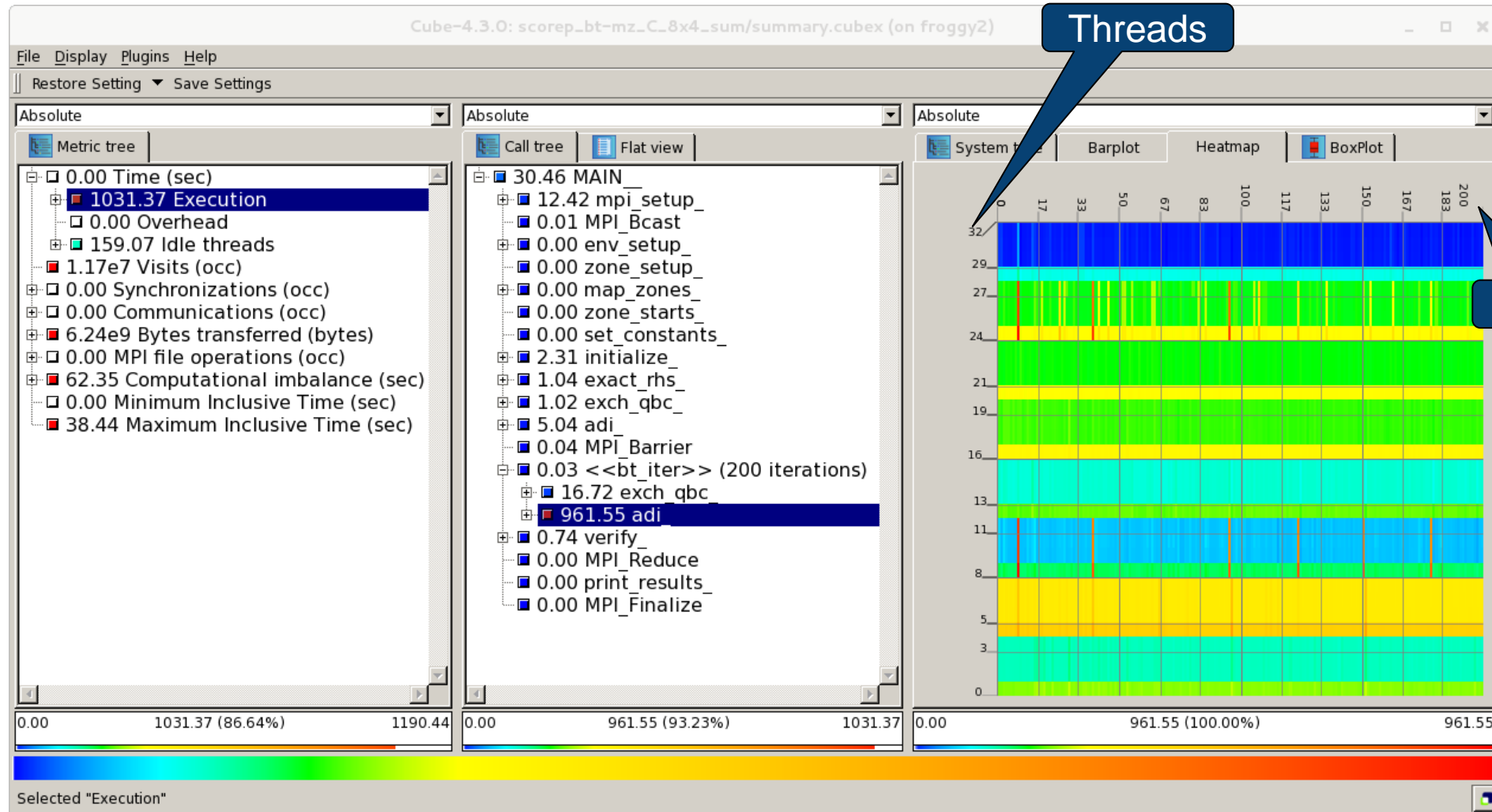
- Select your user instrumented region and mark it as loop
  - Choose hide iterations
  - View the Barplot statistics or the (thread x iterations) Heatmap

# Loop Unrolling - Barplot





# Loop Unrolling – Heatmap



## Further information

---

### CUBE

- Parallel program analysis report exploration tools
  - Libraries for XML report reading & writing
  - Algebra utilities for report processing
  - GUI for interactive analysis exploration
- Available under New BSD open-source license
- Documentation & sources:
  - <http://www.scalasca.org>
- User guide also part of installation:
  - ``cube-config --cube-dir` /share/doc/CubeGuide.pdf`
- Contact:
  - mailto: [scalasca@fz-juelich.de](mailto:scalasca@fz-juelich.de)

