



Framesoc User Guide v1.0.8

Generoso Pagano, Youenn Corre

**INRIA
REPORT**

July 2015

Project-Team MESCAL



Framesoc User Guide

v1.0.8

Generoso Pagano*, Youenn Corre*

Project-Team MESCAL

Inria Report — July 2015 — 29 pages

Abstract: This guide describes the Framesoc workbench, which is the desktop user environment provided by Framesoc infrastructure. The document targets the end-users.

* INRIA, firstname.lastname@inria.fr

**RESEARCH CENTRE
GRENOBLE – RHÔNE-ALPES**

Inovallée
655 avenue de l'Europe Montbonnot
38334 Saint Ismier Cedex

Table of contents

1	Introduction	4
2	Installation and Setup	5
3	Framesoc Perspective	5
4	Management Views	6
4.1	Trace Browser	6
4.1.1	Trace Filter	7
4.2	Trace Metadata Viewer/Editor	8
5	Analysis Views	10
5.1	General principles	10
5.2	Statistics Pie Chart	14
5.3	Event Density Chart	16
5.4	Event Table View	17
5.5	Gantt Chart View	18
6	Framesoc Menu and Toolbar	20
6.1	Preferences	21
6.1.1	Database Settings	21
6.1.2	GUI settings	24
6.1.3	Color Management	24
6.1.4	Tool Management	25
6.2	Trace Import	26
6.3	Launch Analysis Tool	26
6.4	Trace Export	26
A	Framesoc Configuration File	28

1 Introduction

Framesoc, the SoC-Trace project [1] trace management infrastructure, is composed by three main layers, as shown in Figure 1.

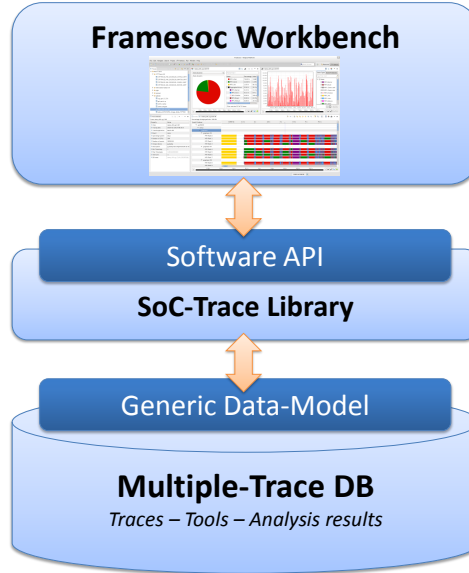


Figure 1: Framesoc Architecture

At the lowest level, there is a Multiple-Trace database, which stores all the data concerning traces, analysis tools and the analysis result produced by the tools, using a generic data-model. Above this layer, the SoC-Trace library provides a software API to facilitate the interaction with the data-model and help the implementation of trace tools. On top of this software library, Framesoc provides a graphical user environment, facilitating trace management, basic trace analysis and tool management.

This guide describes Framesoc from the user perspective (i.e., Framesoc graphical user environment). More information about the lowest layers of Framesoc infrastructure can be found in the technical report RT-427 [2] and the research report RR-LIG-046 [3].

2 Installation and Setup

Framesoc is an application built on top of the Eclipse framework. Therefore, the easiest way to install Framesoc is to use its Eclipse update site, as explained in the online wiki page: <https://github.com/soctrace-inria/framesoc/wiki/Install-and-setup-a-standalone-version-of-Framesoc-using-the-update-site>.

3 Framesoc Perspective

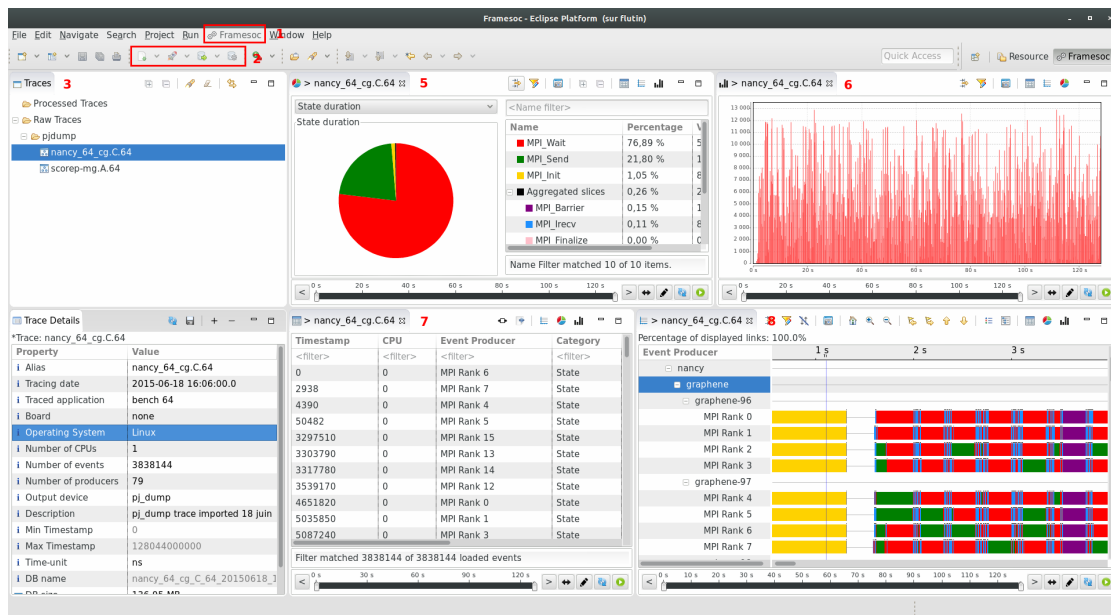


Figure 2: Framesoc Eclipse Perspective

Framesoc provides an Eclipse perspective¹ for trace management and analysis. The Framesoc perspective (Figure 2) contains the following elements:

- A Framesoc menu (1).
- A toolbar (2).
- A set of views for trace management and analysis. We find on the left the two *management views*: a trace browser (3) and a trace metadata viewer/editor (4). On the right, there are four *analysis views*: a statistics pie chart (5), an event density chart (6), an event table (7), and a Gantt chart (8).

The management views (trace browser and metadata viewer/editor) refer to the whole system and are typically the entry point for trace analysis. For this reason, they cannot be closed and there can be only a single instance of each of them. On the contrary, all the analysis views refer

¹Within an Eclipse application, a perspective defines an initial set and layout of views, menu and toolbars. The Framesoc Eclipse perspective can be activated by selecting it from the **Window > Open Perspective > Other... > Framesoc** menu.

to a single trace, so they can be opened and closed as needed. The alias of the trace currently shown within an analysis view is displayed as the view name, beside the view icon. For each trace, there can be an instance of each type of analysis view. The maximum number of open instances for a given type of analysis view is configurable (Appendix A). When a trace is selected in the trace browser, the corresponding metadata are shown in the metadata viewer and all the analysis views referring to that trace (if any) are highlighted. Namely, the view name (which corresponds to the trace alias) is preceded by an arrow (δ). In the example shown in Figure 2, all analysis views refer to the selected trace **nancy_64_cg.C.64**. On the other hand, when an analysis view is given focus, the trace shown in that view becomes the selected trace in the trace browser, so the metadata viewer is updated accordingly and all the analysis views are consistently highlighted or unhighlighted.

The following sections describe in detail Framesoc management and analysis views, as well as the functionalities accessible via Framesoc menu and toolbar.

4 Management Views

4.1 Trace Browser

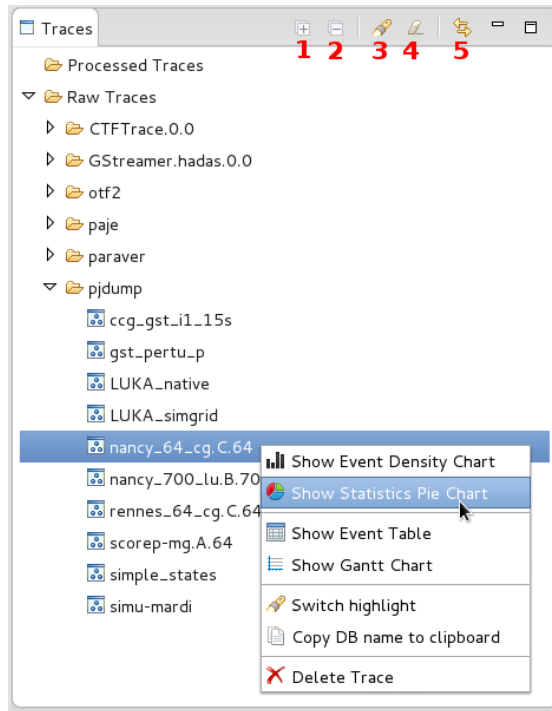


Figure 3: Traces view

The *Traces* view (Figure 3) is the Framesoc perspective trace browser. Traces are presented in a tree viewer with a two-level hierarchy. The first level distinguishes processed traces from raw traces ². Then, in each of these two categories, traces are grouped by type. The type may

²As described in the technical report RT-427 [2], a processed trace is a trace created by an analysis tool as the result of an analysis on another trace.

relate to the trace format or to the tool that created the trace. The viewer presents a trace alias for each trace.

Double-clicking on a trace opens the Gantt Chart for that trace (Subsection 5.5). Right-clicking on a single trace opens a context menu that gives access to the following functionalities:

- *Show Event Density Chart*: open the density chart view for this trace (see Subsection 5.3).
- *Show Statistics Pie Chart*: open the pie chart view for this trace (see Subsection 5.2).
- *Show Event Table*: open the event table view for this trace (see Subsection 5.4).
- *Show Gantt Chart*: open the Gantt chart view for this trace (see Subsection 5.5).
- *Switch highlight*: switch the highlight state of the trace in the browser (see Figure 5). A trace being highlighted in the browser is shown in bold.
- *Copy DB name to clipboard*: copy to the clipboard the name of the database containing the trace raw data.
- *Delete Trace*: delete the trace from the system.

Extra options in this menu may be available, depending on the tools currently installed in Framesoc. For instance, if Ocelotl is installed, then a new line is added to open the trace with Ocelotl (i.e. *Show Ocelotl*). If more than one trace is selected, only the *Switch highlight* and *Delete Trace* entries are displayed in this context menu. Note that if one or more traces are deleted, an update notification is sent to the other views. This allows a view to do the necessary actions if the trace it displays has been deleted. Note also that if a given type of analysis view (e.g., the Gantt chart view) is already open for a given trace, when right-clicking on that trace the context menu will propose to open a new instance of that type of view (e.g., *Show Another Gantt Chart* menu entry).

The view toolbar contains the following buttons: a couple of buttons to expand (1) and collapse (2) the trace hierarchy, a button to launch the *Trace Filter Dialog* (3, see Subsection 4.1.1), a button to clean the highlight state of all traces (4, see Subsection 4.1.1) and a button to manually resynchronize the displayed traces with the information contained in the Framesoc System DB³ (5).

4.1.1 Trace Filter

Pressing the button indicated as 3 in Figure 3, the *Trace Filter Dialog* is opened (Figure 4).

This dialog displays a table containing all the traces imported in the system. Each column of this table corresponds to a particular trace metadata. Clicking on the header of each column, it is possible to sort the table rows according to the column content. The first row of the table has editable fields, allowing for filtering on the corresponding column content. Filtering can be done using regular expressions. At the bottom of the table, the number of traces matched by the filter, over the totality of traces, is displayed. In the example shown in Figure 4, we filtered all the traces produced on “Linux”, containing the string “pj.dump” in the description.

The first column of each row has a checkbox, allowing to select/deselect the corresponding trace. The checkbox in the first row (filter row) select/deselect all shown traces. When pressing *OK*, all selected traces are highlighted (shown in bold) in the trace browser (Figure 5). This allows the analyst to immediately spot interesting traces in the browser.

³See the technical report RT-427 [2] for further details on the database architecture.

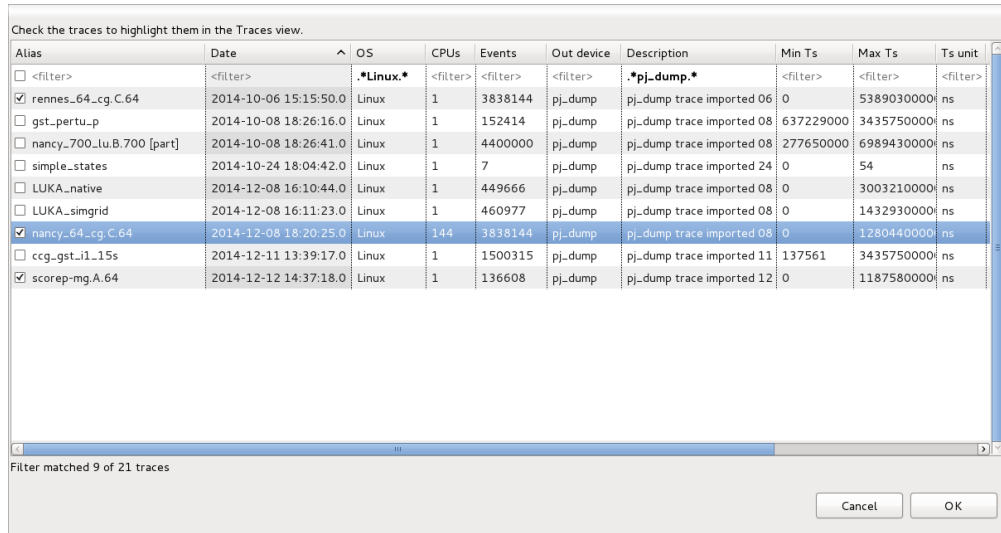


Figure 4: Trace Filter Dialog

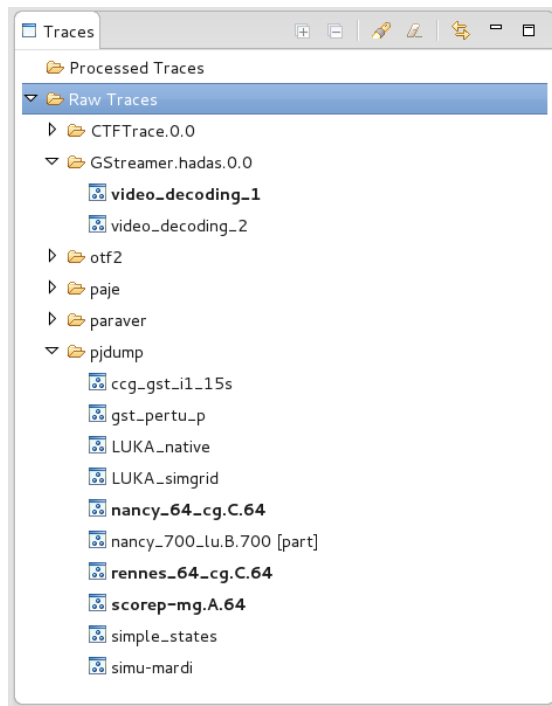


Figure 5: Highlighted traces in Traces view

4.2 Trace Metadata Viewer/Editor

The *Trace Details* view (Figure 6) is the Framesoc perspective trace metadata viewer and editor. When a trace is selected in the *Traces* view, the trace metadata are displayed in a table viewer

Property	Value
i Alias	nancy_64_cg.C.64
i Tracing date	2015-06-18 16:06:00.0
i Traced application	bench 64
i Board	none
i Operating System	Linux
i Number of CPUs	1
i Number of events	3838144
i Number of producers	79
i Output device	pj_dump
i Description	pj_dump trace imported 18 juin 2015 1
i Min Timestamp	0
i Max Timestamp	128044000000
i Time-unit	ns
i DB name	nancy_64_cg_C_64_20150618_160542
DB size	136,95 MB
Trace size	261,38 MB

Figure 6: Trace Details view

containing two columns: the property name and the property value. If more than one trace is selected in the *Traces* view, only the properties having the same name and the same values are shown. The different properties are grouped in two different categories: predefined properties (displayed first) and custom properties (displayed last). Those categories are identified by two different icons. The *Value* column is normally editable, with the exception of some properties that are read only. In order to edit an editable property value, the user simply has to click the value and modify it (as shown in Figure 6 for the *OS* property). When one or more values have been modified, a star (*) is displayed before the trace alias, on top of the table, and the two view toolbar buttons 1 and 2 are enabled. The *Reset changes* button (1) restores all non-saved edited properties to their previous value. The *Save changes* button (2) stores the changes. If the *Alias* predefined property of a trace is persistently modified, other views are notified in order to take the necessary actions (e.g., update their label for the trace).

The other two buttons in the toolbar (3 and 4) allows for adding and removing custom trace metadata to the currently selected traces. Pressing the button indicated with 3 in Figure 6 will display a dialog to add a new property (Figure 7).

In the example shown in Figure 7, after pressing *OK*, a new trace property *ISSUE* of type *INTEGER* and value 11 is added to the selected trace. The trace metadata will therefore look as shown in Figure 8. In this figure it is possible to see that, when selecting a custom property, the *Remove property* button (indicated as 4 in Figure 6, where it is disabled) is enabled and can be used to remove the selected custom property.

Note that if more than one trace is selected in the *Traces* view, it is still possible to edit, add or remove properties.

New custom trace parameter

Name: ISSUE

Type: INTEGER

Value: 11

Cancel OK

Figure 7: Dialog to add a new custom property as trace metadata

Trace: nancy_64_cg.C.64

Property	Value
Alias	nancy_64_cg.C.64
Tracing date	2014-12-08 18:20:25.0
Traced application	bench_64
Board	none
Operating System	Linux
Number of CPUs	144
Number of events	3838144
Output device	pj_dump
Description	pj_dump trace imported 08 Dec 2014
Min Timestamp	0
Max Timestamp	128044000000
Time-unit	ns
DB name	nancy_64_cg_C_64_20141208_1720
ISSUE	11

Figure 8: Trace Details view after adding a custom property (ISSUE)

5 Analysis Views

5.1 General principles

Before describing in detail the different Framesoc analysis views, we present here some general principles that hold for all such views.

Data management All Framesoc analysis views try to maximize the interactivity and responsiveness of the system by pipelining data loading and visualization. This allows the user to start working with data (or at least visualize it), while the loading job is ongoing. For example, the Gantt chart allows for zooming in/out or panning the visualized part of the trace, while trace

events are being loaded.

The user has full control on the loading job, and can stop it when she wants. In this case, the data loaded so far are still available for analysis. To cancel a loading job, the user uses the *Progress* view (see for example Figure 9 for the *Progress* view showing the Gantt loading process.).

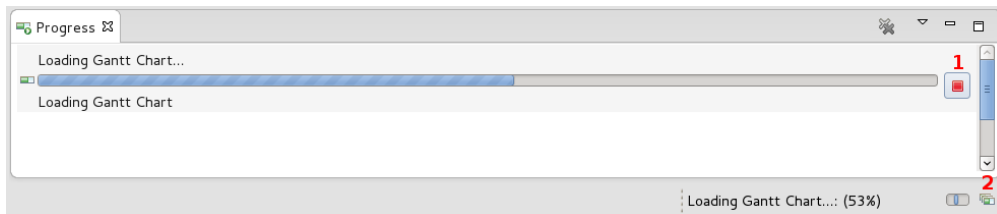


Figure 9: Progress view enabling to cancel a loading job

The button indicated as 1 requests for stopping the job. The *Progress* view is not shown by default in the Framesoc perspective. To access this view, the user has to click on the button indicated as 2, which is always visible in Eclipse when a job is in progress.

Time management Each Framesoc view is capable to load and display trace data related to whatever trace time interval: it is not compulsory to load in memory the entire trace to show only a given portion of it; it is possible, on the contrary, to load only a specific sub-interval. For example, the user has the flexibility to load only a specific time window of interest in the Gantt chart, avoiding charging into memory portions of the trace that are not interesting for the analysis.

The user controls the portion of the trace actually loaded into memory and displayed in the analysis view using a time management bar, located at the bottom of each Framesoc analysis view (Figure 10).



Figure 10: Time management bar present in all Framesoc analysis view

This bar contains a double range time slider (1) representing the whole trace duration, surrounded by two arrow buttons (2 and 3), and followed by four more buttons on the right (4, 5, 6 and 7). The solid part of the time bar (here colored in black) identifies the portion of the trace actually loaded in the view. The semi-transparent grey portion shows the currently selected part of the trace. Using this kind of representation, the user always keeps a global visibility on the whole trace, while loading only the information he is interested in. In order to change the time window loaded in the view, the user can move the knobs 8 and 9 of the double range slider. However, in order to avoid spurious and useless data transfers (from disk to memory), the new time interval is actually loaded only when the user presses the button indicated as 7. The button indicated as 6 resynchronizes the time bar with the time window actually loaded in the table, if they differ (i.e., the user modified the interval in the timebar without triggering a load with button 7). In order to modify the time window visualized in the double range slider the user has several possibilities.

- *Graphical selection*: it is the most intuitive way and it involves using the two knobs 8 and

9, in order to graphically set the two bounds of the time interval. Alternatively, the user can also click on the time bar which will move the closest knob to that position.

- *Time window navigation*: this possibility involves using the two arrow buttons 2 and 3, to select a time window that has the same size that the currently visualized one, but is located immediately before or immediately after the current one.
- *Manual selection*: by pressing the button indicated as 5, the user accesses a dialog (Figure 11) where it is possible to explicitly put the exact values of the time interval start and end timestamps. This dialog enables also a manual definition of the size of the time window to be used when performing the *time window navigation* described above.
- *Select all*: using the button indicated as 4, the user easily selects the whole trace duration.

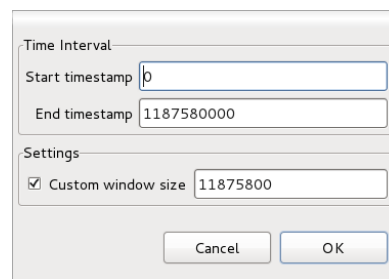


Figure 11: Time window dialog

Regarding timestamps representation in Framesoc, if no time unit is indicated beside a displayed timestamp, by default the time unit used is the one shown in the *Trace Details* (Section 4.2) view for that trace (**Time-Unit** metadata). This applies to the time management bar described above, as well as to any other Framesoc GUI place.

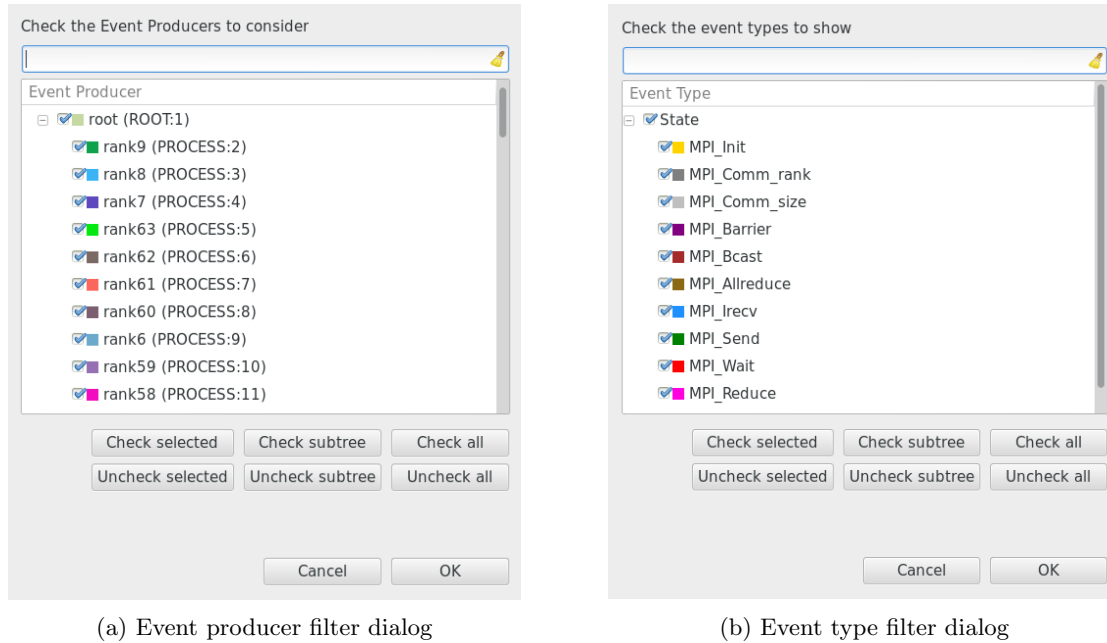
Event producer and event type filter The pie chart, the histogram and the Gantt chart views all have filter options in order to give the users more control over their analysis. These filters allow to select the event producers and the event types used when loading the view. These filters can be accessed through the buttons in Figure 12 that are in the top right corner of the views. Clicking on one of these buttons will open a new dialog.



Figure 12: Filter buttons

For the event producer filter dialog, shown in Figure 13a, all the event producers are displayed as a hierarchy tree, where each node and leaf is a checkbox that allow to select the event producer. At the bottom of the window, several buttons are available that allow to perform select operations. These buttons are:

- *Check/Uncheck Selected*: select/unselect the selected producer.
- *Check/Uncheck Subtree*: select/unselect the selected producer and all its children in the hierarchy.



(a) Event producer filter dialog

(b) Event type filter dialog

Figure 13: Framesoc filters

- *Check/Uncheck All*: select/unselect all the event producers.

At the top of the hierarchy tree, an editable text field can be used to search among the event producers.

The event type filter dialog, illustrated in Figure 13b, displayed all the event types present in the current trace. The event types are displayed as a hierarchy tree where they are grouped by their category (i.e. punctual event, state, link or variable). The dialog is similar to the event producer one, described in the previous paragraph, including the text field to search among the event types and the buttons to perform the select operations.

Note that as opposed to the time management bar, where in order to perform the filtering on time the user has to press a button, these filters are applied immediately after the user has pressed the *OK* button in the dialog.

View communication Using the Framesoc Bus (described in the RT-447 [4]), Framesoc views can communicate with each other. This mechanism is mostly used to synchronize the different views on time intervals, and to propagate to all the views changes in colors (see Subsection 6.1.3) or trace metadata (see Subsection 4.2). Regarding time interval synchronization, in the toolbar of each Framesoc analysis view, there are some *Show in...* buttons, allowing the user to pass from one view to another, still keeping the same trace and the same time interval. Figure 14 shows all the possible *Show in...* buttons.

Figure 14: *Show in...* toolbar buttons

Button **1** switches to the event table view (see Subsection 5.4). Button **2** switches to the Gantt chart view (see Subsection 5.5). Button **3** switches to the statistics pie chart view; in this case, a dialog is shown to the user in order to select one of the possible statistics operators (see Subsection 5.2). Button **4** switches to the event density chart view (Subsection 5.3).

Of course, a view toolbar does not have a button redirecting to the view itself, so only three out of four possible *Show in...* buttons are displayed in each view toolbar.

Group of view If the *Allow view replication* option in the preferences is enabled (see Subsection 6.1.2), it is possible to open several instances of the same type of view for the same trace. Indeed, when right-clicking on a trace in the Trace browser, if for that trace an instance of a given type of view is already open, the *Show Another...* menu item is proposed. Clicking on this option will open a new instance of the selected type of view for the selected trace and also creates a new *group of view*. A group defines a family of different types of views (i.e., Gantt, Pie, etc.) open with a given trace. Each group can contain one instance of each type of view. If the user performs a switch from a type of view (e.g., Gantt) to another (e.g., Pie), if the group of view does not yet contain an instance of the demanded view, a new instance of such a view will be created for the group. On the contrary, if the demanded type of view already exists in the group, this view is put forward and it is synchronized on time (as described in the previous paragraph). When there are at least two groups of view for the same trace, the name of the trace in the views is preceded by a number identifying the group to which it belongs.



Figure 15: Snapshot button

Snapshot All the analysis views can be exported into an external file(s), by clicking on the button shown in Figure 15. How the results are exported is explained for each analysis view in the following subsections. In every case, the results are placed in a directory created with the name of the current trace and a timestamp of the time and date.

5.2 Statistics Pie Chart

The *Statistics Pie Chart* view (Figure 16) is a Framesoc analysis view presenting several metrics in the form of a pie chart. The view is composed of two parts. On the left, there are the metric selector (**1**) and the actual pie chart (**2**). On the right, a table viewer (**4**) displays the same information as the pie chart. The table has three columns: the pie-slice name, the percentage value and the actual value. The name column cells contain a small square icon, filled with the color corresponding to the slice. Each column header, when clicked, triggers the sorting of the rows according to the values in the corresponding column. The content of this table viewer can be filtered by name using the text field on top of it (**3**). The number of items matched by the filter, over the totality of items, is shown in the status bar under the table viewer (**5**). At the bottom of the view, there is the time management bar described in Subsection 5.1.

For the time being, four metrics are available:

- Event Producer instances: each slice represents the number of events having a given event producer.
- Event Type instances: each slice represents the number of events having a given event type.

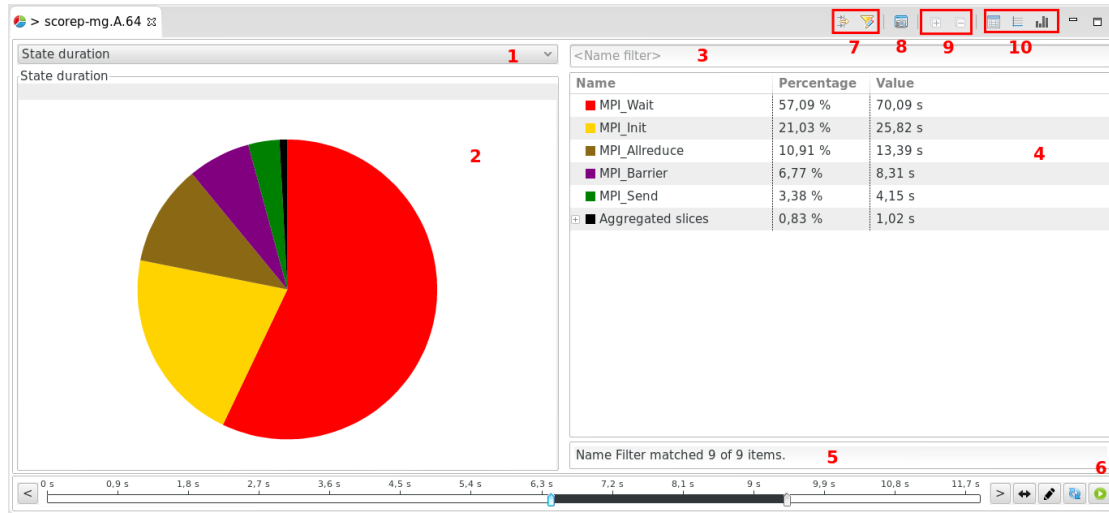


Figure 16: Statistics Pie Chart view

- Link duration: each slice represents the duration of a given type of links.
- State duration: each slice represents the duration of a given type of state.

Note that each metric is computed considering only the selected time interval.

From Figure 16 it is possible to note that there can be a special slice in the pie: the *Aggregated* slice. This slice aggregates all the slices whose value is smaller than a given threshold, being therefore difficult or impossible to see. This threshold has been empirically set to 1%, taking into account user ergonomics and screen limitations. In the table on the right, the *Aggregated* slice corresponds to a folder entry, whose sub-entries are the actual slices. All the detailed information is thus kept and made available in the tabular representation.

By right clicking on one or more of the table elements, a context menu is shown. The commands available in this menu vary according to the elements selected. If the *Aggregated* slice row is selected, no menu is shown. In any other case, the menu can contain one or more of the following commands:

- *Exclude Item from statistics / Exclude items from statistics*: shown if only leaf items are selected. This command excludes the selected leaf items from statistics computation. If there is at least one excluded item, the number of excluded items is shown in the status bar (5).
- *Merge Items*: shown only if two or more leaf items are selected. This command merges the items in a new folder item. The name and the color of the new folder item are chosen by the user with the dialog shown in Figure 17.
- *Unmerge Items*: shown if one or more folder items are selected. This command unmerges the items contained in the folder item.
- *Unmerge All Items*: shown if at least one folder item, created by merging two or more leaf items, exists. This command unmerges all the merged items.

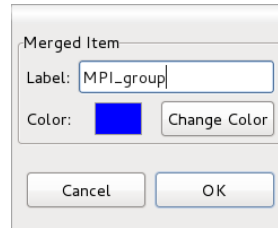


Figure 17: Merge Items Dialog

- *Restore Excluded Items*: shown if at least one leaf item has been excluded from statistics computation. This command restores all previously excluded items into statistics computation.

Note that when a *Statistics Pie Chart* view is opened for a given trace using the context menu in the trace browser, no pie chart is actually displayed, since the user has to select the metric of interest first, then press the load button (6) in the time management bar.

The view toolbar contains the *Show in...* buttons (10), the filter button (7) and a couple of buttons to expand/collapse the folder items in the table (9). The snapshot button (8) creates an image of the pie chart, creates a text file with the current parameters (trace name, statistics operator, etc.) and exports the value of the table in CSV format. An option dialog allows to select the size of the image and the destination directory.

5.3 Event Density Chart

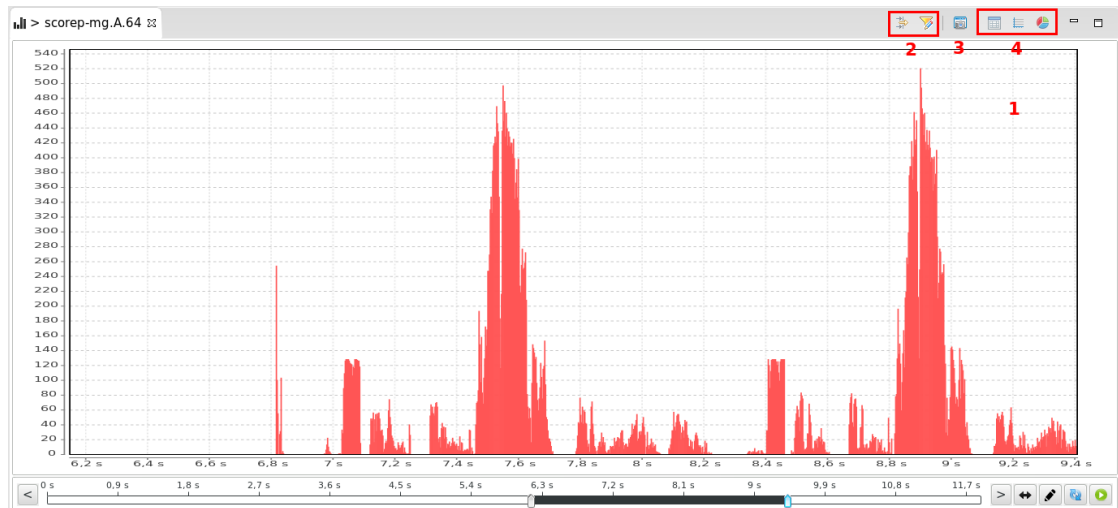


Figure 18: Event Density Chart view

The *Event Density Chart* view (Figure 18) is a Framesoc analysis view displaying the event density over time in the form of a histogram.

In the histogram (1), the x axis represents the time, while the y axis represents the number of events. The histogram number of bins is fixed and has been chosen taking into account the

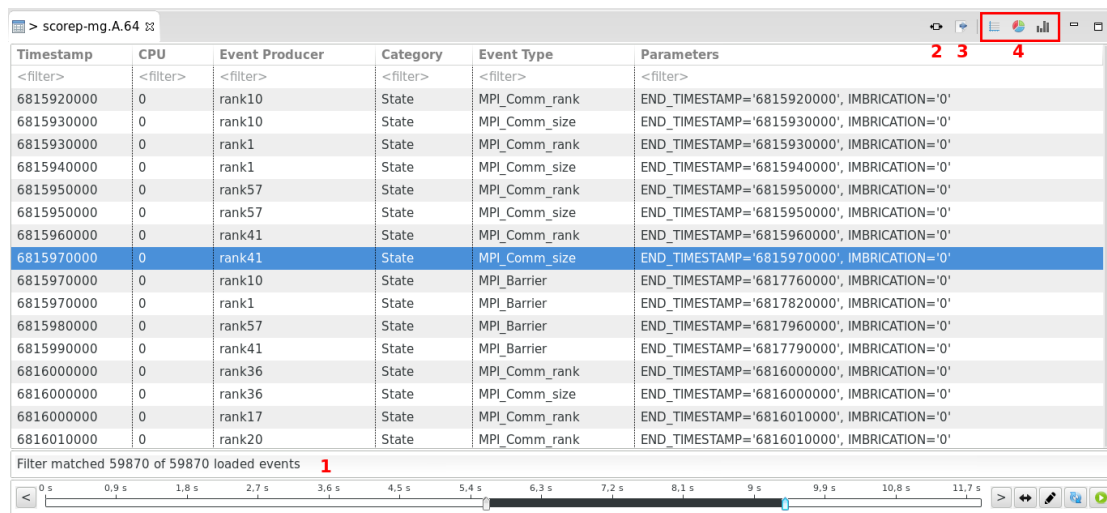
number of pixels actually present on a screen, to ensure a clear visualization. The user can zoom in and out on portions of the chart. In particular, to zoom in on a portion of the chart, the user has to click at the start of the portion, drag the mouse going to the right up to the end of the portion, then release the click. To zoom out completely, the user has to do the same as above, but dragging the mouse to the left this time. When hovering the mouse on a bin, some information about the bin is displayed: the central timestamp and the number of events.

In top right corner, there is the event type and event producer filter buttons (2), and the view switching buttons (4). The features of these buttons are described in Subsection 5.1.

At the bottom of the view, there is the time management bar described in Subsection 5.1. The event density is computed considering only the events in the selected time interval, for the selected types and the selected producers.

The snapshot button (3) creates an image of the event density chart and a text file containing the current context (trace name, filtered event types and event producers, etc.). An option dialog allows to select the size of the image and the destination directory.

5.4 Event Table View



Timestamp	CPU	Event Producer	Category	Event Type	Parameters
<filter>	<filter>	<filter>	<filter>	<filter>	<filter>
6815920000	0	rank10	State	MPI_Comm_rank	END_TIMESTAMP='6815920000', IMBRICATION='0'
6815930000	0	rank10	State	MPI_Comm_size	END_TIMESTAMP='6815930000', IMBRICATION='0'
6815930000	0	rank1	State	MPI_Comm_rank	END_TIMESTAMP='6815930000', IMBRICATION='0'
6815940000	0	rank1	State	MPI_Comm_size	END_TIMESTAMP='6815940000', IMBRICATION='0'
6815950000	0	rank57	State	MPI_Comm_rank	END_TIMESTAMP='6815950000', IMBRICATION='0'
6815950000	0	rank57	State	MPI_Comm_size	END_TIMESTAMP='6815950000', IMBRICATION='0'
6815960000	0	rank41	State	MPI_Comm_rank	END_TIMESTAMP='6815960000', IMBRICATION='0'
6815970000	0	rank41	State	MPI_Comm_size	END_TIMESTAMP='6815970000', IMBRICATION='0'
6815970000	0	rank10	State	MPI_Barrier	END_TIMESTAMP='6817760000', IMBRICATION='0'
6815970000	0	rank1	State	MPI_Barrier	END_TIMESTAMP='6817820000', IMBRICATION='0'
6815980000	0	rank57	State	MPI_Barrier	END_TIMESTAMP='6817960000', IMBRICATION='0'
6815990000	0	rank41	State	MPI_Barrier	END_TIMESTAMP='6817790000', IMBRICATION='0'
6816000000	0	rank36	State	MPI_Comm_rank	END_TIMESTAMP='6816000000', IMBRICATION='0'
6816000000	0	rank36	State	MPI_Comm_size	END_TIMESTAMP='6816000000', IMBRICATION='0'
6816000000	0	rank17	State	MPI_Comm_rank	END_TIMESTAMP='6816010000', IMBRICATION='0'
6816010000	0	rank20	State	MPI_Comm_rank	END_TIMESTAMP='6816010000', IMBRICATION='0'

Filter matched 59870 of 59870 loaded events

Figure 19: Events view

The *Events* view (Figure 19) is a Framesoc analysis view showing a tabular representation of trace events. The main element of this view is a table viewer, displaying a distinct row for each event of the trace. This table has the following columns:

- Timestamp: the event timestamp.
- CPU: the number of the CPU on which the event has been produced.
- Event Producer: the name of the entity producing the event.
- Category: the event category (State, Link, Punctual Event, Variable).
- Event Type: the event type name.
- Parameters: the list of the event custom parameters, with the format NAME='VALUE'.

The first row of the table contains an editable filter for each column. These filters accept regular expressions. The number of events matched by the filter, over the totality of loaded events, is shown in the status bar under the table viewer (1). Filtering takes place in a separate thread and can be stopped pressing **ESC**. All filters can be cleaned contextually pressing **DEL**.

At the bottom of the view, there is the time management bar described in Subsection 5.1.

Beside the three *Show in...* buttons indicated as 4 (see Subsection 5.1), the view toolbar contains a button (2) that resizes column width to fit the actual content. There is also a button (3) to export the table values into a CSV file. When exporting to CSV, an option dialog allows to select the columns that will be exported in the CSV file, along with specifying the destination directory. The export tool only exports the currently loaded values in the table, i.e. if a filter is active, then only the filtered elements are exported. Note that for the custom parameters, since not all event types have the same parameters, the export contains empty values, i.e. when an event type does not have a custom parameter then the corresponding CSV value is left blank⁴.

5.5 Gantt Chart View



Figure 20: Gantt Chart view

This view (Figure 20) shows a Gantt chart representation of trace events. This kind of representation is pretty common in trace analysis frameworks (e.g., Ltng Eclipse Viewer [5]) and is classically used to visualize application behavior over time, thanks to its ability to represent causality relations [6].

The main element of this view is a Gantt viewer⁵, which is composed of two parts.

On the left there is the hierarchy of event producers. Double clicking on a non-leaf node of the tree, the node is expanded/collapsed. Right clicking on a node, a contextual menu is shown. For leaf nodes, the menu allows the user to hide them. For non-leaf nodes, the menu allows the user to hide them and their sons, or to expand/collapse them.

On the right there is the actual time chart, showing for each producer all the punctual events it generates (represented as simple vertical lines) and all the states it spends time in (represented

⁴These blank values can be filled automatically by external tools. For example, R has a "fill" option when importing a CSV that replaces blank values with "NA".

⁵The viewer is based on the one available in the Ltng Eclipse Viewer [5]

as colored rectangles). If there are links in the trace (e.g., communications), they are represented as oriented arrows. The Gantt chart does not currently consider variables. The colors used for the different entities depend on the event type. When hovering the mouse on a given entity, a tooltip with detailed information is shown. For each entity the tooltip contains specific information:

- for a state: the producer, the type, the start/end timestamps and the duration.
- for a link: the source/target producers, the start/end timestamps and the duration.
- for a punctual event: the producer, the type and the timestamp.

Zooming in/out on a given point of the Gantt chart, is done putting the mouse on the point of interest, then moving the mouse wheel while pressing **CTRL**. When zooming in/out the following functionalities can be triggered:

- State aggregation: if there are not enough pixels to show all the states in a given time region, these states are aggregated and a black point is displayed above the point where the aggregation took place.
- Link filtering: if there are not enough pixels to show all the links in a given time region, only the visible links are shown. The percentage of shown links (computed over the total of links in the time region currently visible) is displayed above the Gantt chart (15). When there exist no links in the trace (in the time region currently visible) the percentage value is 100%. This state of this option is global, i.e. the future instances of the Gantt chart view will use the value of this option when it was last modified (e.g. if link display is turned off in the Gantt, all the next instances of the Gantt view will not display the links).

These two functionalities keep the user aware of what is going on (information loss), thus reducing visual artifacts and possible analysis pitfalls.

The user can select a time region in the Gantt using the mouse (click on the start point, then drag and release on the end point). While selecting a given region, the same selection is applied to the time management bar present at the bottom of the view (see Subsection 5.1). Similarly, changing the selection in the time management bar, changes the selection in the Gantt chart. When a time region is selected, clicking on a given point of the Gantt chart removes the selection and restore the time management bar selection as it was before.

Beside the three *Show in...* buttons indicated as 14 (see Subsection 5.1), the view toolbar contains the following buttons:

- Button 1: display a dialog where it is possible to select the event producers to display.
- Button 2: display a dialog where it is possible to select the types to display.
- Button 3: hide/display links.
- Button 4: take a snapshot. A more detailed description is given below.
- Button 5: fit the zoom to the available screen space, to display all the loaded portion of the trace in the view.
- Button 6: zoom in.
- Button 7: zoom out.
- Button 8: select the previous event.

- Button **9**: select the next event.
- Button **10**: select the previous event producer line.
- Button **11**: select the next event producer line.
- Button **12**: display a dialog showing the color legend.
- Button **13**: use the CPU drawer; the trace is reloaded using a different event producer hierarchy where CPUs are the roots.

The snapshot button creates an image of the Gantt chart and a text file containing the context (trace name, time interval, filtered elements, etc.). When performing a snapshot, an option dialog allows to select the destination directory, the size of the snapshot, if the snapshot should include the time scale and if the whole Gantt height should be included. The snapshot only includes the current display of the Gantt, which means that:

- if the currently shown part do not show the whole loaded time interval (i.e. there is a zoom in), then only the currently shown part will be captured;
- if links are displayed they will also be captured;
- if part of the hierarchy is collapsed or hidden, then the capture of the Gantt will not show the hidden elements;
- if a selection is displayed, it will also be captured.

The graph is redrawn according to the given width, meaning that giving a bigger width than the current display, can alter the chart by drawing more details, i.e. by drawing events or states that were previously aggregated. If the whole Gantt height option is selected then the height of the final picture is constrained by the height of the Gantt. If the option is not selected, then the capture will start at the currently displayed starting point and will stop at the specified height, or at the bottom of the graph (whichever comes first).

6 Framesoc Menu and Toolbar

When the Framesoc perspective is activated, the Framesoc menu (Figure 21) and the corresponding toolbar (Figure 22) are visible.

In the menu, there are two items: the trace analysis menu and preferences. The preferences allows the user to access database system configuration, GUI configuration, color management and tool management. The trace analysis menu allows the user to launch importers, analysis tools and exporter tools.

In the toolbar, the different buttons are simply shortcuts for the above functionalities, where corresponding icons mean corresponding functionalities. The added value of the toolbar is that, beside each of the three buttons used to launch the tools of the various category (import, analysis, export), there is a drop down menu containing the list of the tools of that category, useful to directly launch a tool.

In the following subsections, the functionalities accessible from the menu or the toolbar are explained in detail.

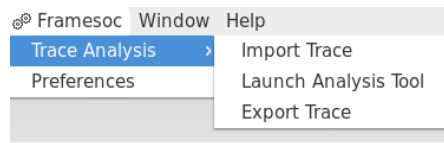


Figure 21: Framesoc Menu



Figure 22: Framesoc Toolbar

6.1 Preferences

The preferences dialog allows the user to configure various settings of Framesoc elements. There are four tabs in the preferences dialog: database, GUI, color management and tool management. The settings modified in this dialog are saved in the Framesoc configuration file (Appendix A).

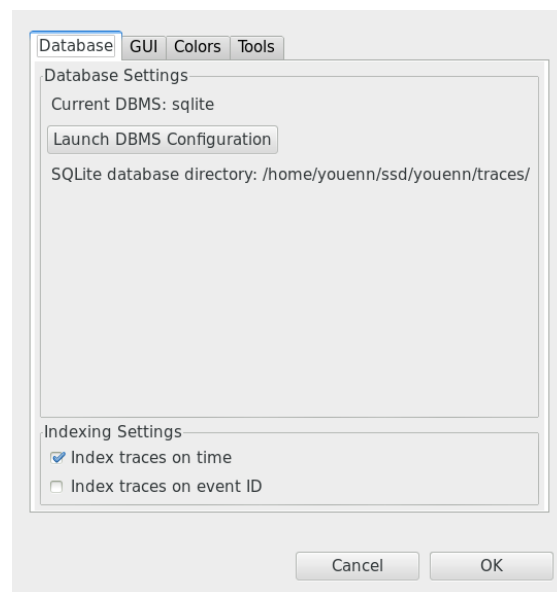


Figure 23: Preferences dialog – Database settings tab

6.1.1 Database Settings

The database settings tab, shown in Figure 23, is divided in two groups: database settings and indexing settings.

The indexing settings allow the user to choose if the traces imported with any of the importers in Framesoc, will be indexed on time, on event ID or on both.

The database settings group displays information about the currently used DBMS and its settings. If the current DBMS is SQLite, then the path to the current database is displayed. If the

current DBMS is MySQL, then the user name and the URL of the database are shown. In order to modify the database settings, you have to click on the button *Launch DBMS configuration*. This will open the DBMS configuration wizard described below.

DataBase Management System Configuration In the DataBase Management System (DBMS) configuration, the user accesses a configuration wizard, whose first page is shown in Figure 24. This page allows the user to select the DBMS to be used for trace storage. In fact, as described in the technical report RT-435 [7], Framesoc can work with several DBMS. At the moment, the support has been implemented for SQLite (recommended) and MySQL.

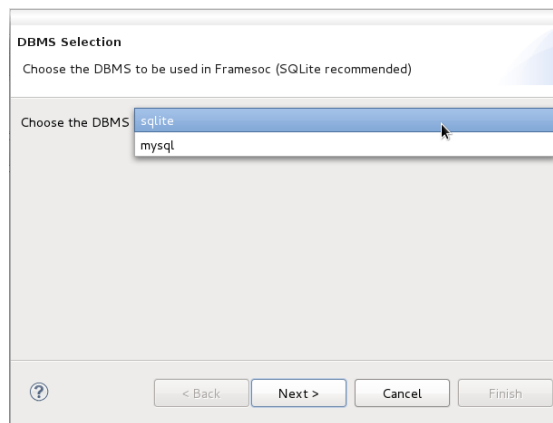


Figure 24: System Initialization: DBMS selection dialog

Once the user has chosen the DBMS and pressed *Next*, he comes to the DBMS configuration page (Figure 25), which is different for each DBMS. If SQLite is selected (Figure 25a), the user simply has to enter the directory where he wants the database files to be kept. Otherwise, if MySQL is chosen (Figure 25b), the user has to specify some connection parameters (user-name, password, URL).

In both cases, after successfully pressing *Finish*, the Framesoc storage subsystem is correctly configured. If a System DB already exists, it is reused, otherwise a new one is created. This configuration is saved in the Framesoc configuration file (Appendix A).

Note that after a modification of the database settings (at initialization or later), in the case of existing System DB, if there is a mismatch between the tools registered in this System DB and the tools actually present in the Framesoc Eclipse runtime⁶, this is automatically fixed:

- If a tool exists in the DB but not in the runtime, the tool is removed with its results (if any)⁷.
- If a tool exists in the runtime but not in the DB, the tool is automatically registered.

The first time Framesoc is launched, the database configuration wizard is automatically launched in order to set up the initial configuration of the database. Note also that at each

⁶As described in the technical report RT-435 [7] the preferred way to add tools to Framesoc is to create an Eclipse plugin, extending a specific extension point defined by Framesoc. For this reason a tool is typically a plugin in the Framesoc Eclipse runtime.

⁷The user is asked for confirmation if the `ask_for_tool_removal` variable is set to `true` in Framesoc configuration file (see Appendix A).

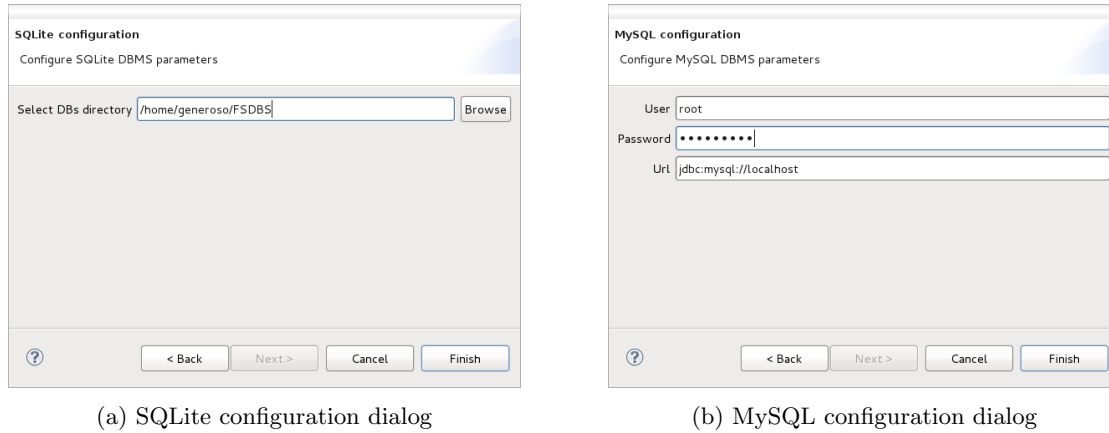


Figure 25: System Initialization: DBMS configuration

Framesoc startup, the system automatically checks that the storage configuration is good. If it is not the case, the database configuration wizard is automatically launched.

A control for mismatch between runtime tools and System DB tools is also done at each startup, with the same policy as described above.

Database Update At each startup of Framesoc or each time a new database is specified, a control is performed to check that the current database is conform to the current model. If Framesoc detects that it is not the case, then it warns the user that the database does not match with the current model, and asks the user if it wants to try to perform an update (Figure 26). During the update, a progress bar showing the progress of the updating process is displayed. If the process fails, a dialog stating the error will be displayed and it might then be necessary to change the repository of the database (*SQLite*) or delete it (*MySQL*). If the process succeeds, a dialog informs the user (Figure 27) and Framesoc can then be used normally.

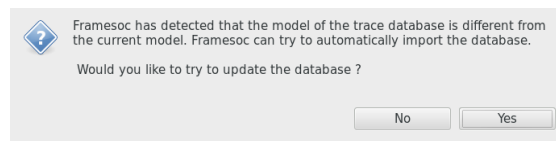


Figure 26: Update Database Question Dialog

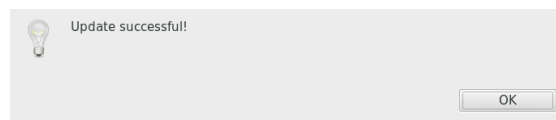


Figure 27: Successful Database Update Dialog

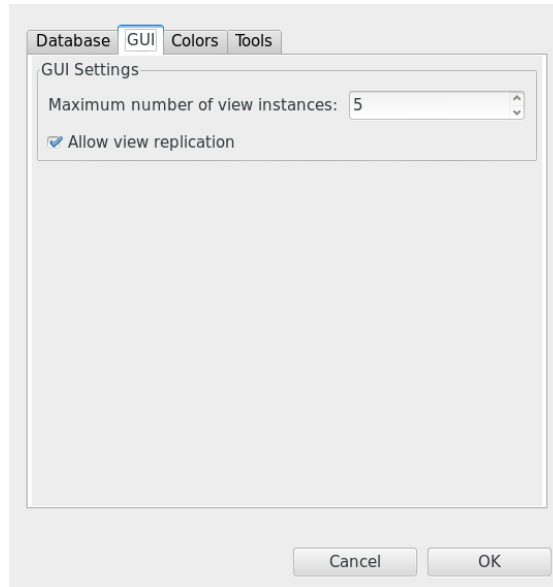


Figure 28: Preferences dialog – GUI settings tab

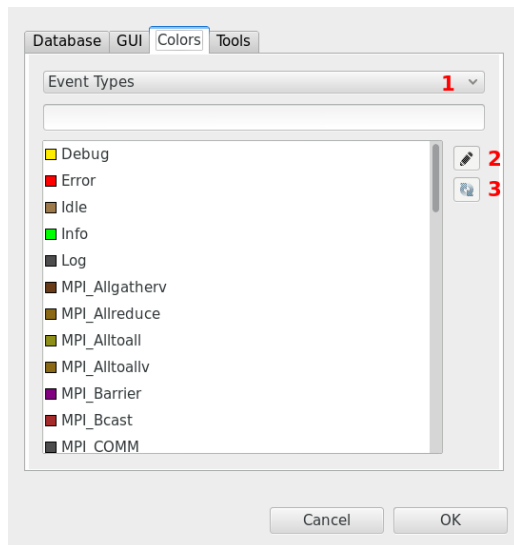
6.1.2 GUI settings

In the GUI settings tab (Figure 28), it is possible to configure two options for Framesoc analysis view management. The first option specifies the maximum number of instances of a type of view (e.g., the Gantt chart) that can be opened simultaneously. The value can be set to -1, meaning that there will be no limit on the number of simultaneous instances. If the value is set to 0, then it is equivalent to a value of 1. The second option is a checkbox that specifies if it is possible to open simultaneously several instances of the view on the same trace.

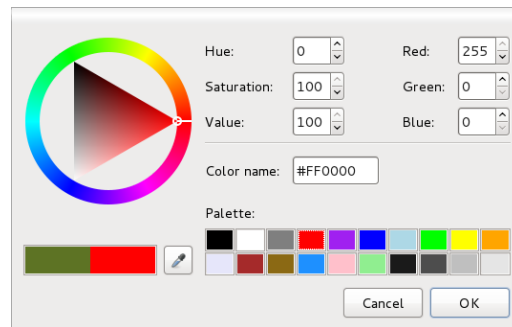
6.1.3 Color Management

The color management tab (Figure 29a) allows the user to modify the colors associated to event producers and event types in a centralized way for the whole workbench. The combo box at the top (1) enables the selection of the entity (event producer or event type). The list below enumerates all instances for a given entity, preceded by a small square icon filled with the instance color. The editable text field on top of this list can be used as a filter on the list. When an entity instance is selected, pressing the *edit* button (2) gives access to the color edition dialog (Figure 29b). Pressing *OK* saves all the changes. Pressing the *reset* button (3) reverts unsaved changes for the current entity. The color configuration for a given entity is physically stored in a configuration file located in the `configuration/fr.inria.soctrace.framesoc.ui` sub-folder of the Eclipse install directory. For example, the relative path to Eclipse install directory of the event type color configuration file is: `configuration/fr.inria.soctrace.framesoc.ui/event_type_colors`.

Note that when *OK* is pressed after changing some colors, the workbench modules are notified and the views may react by updating their colors in real-time.



(a) Color management tab



(b) Color editing dialog

Figure 29: Framesoc Color Management

6.1.4 Tool Management

The tool management tab (Figure 30) simply displays the list of tools registered in the system, enabling the installation, modification and removal of external *black-box* tools (tools that are not Eclipse plugins).

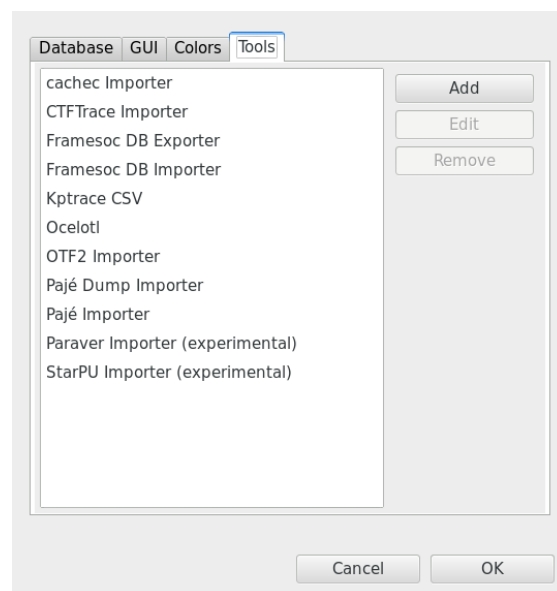


Figure 30: Tool manager tab

Adding or modifying a *black-box* tool requires the user to edit the different fields of the dialog

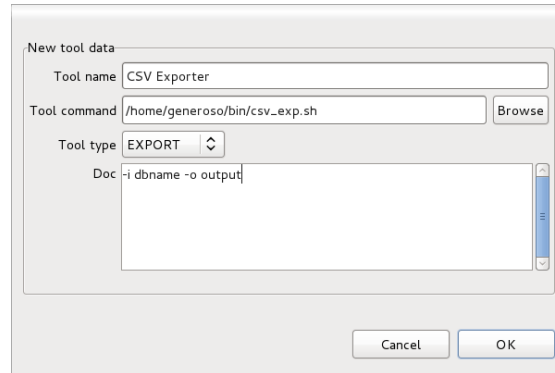


Figure 31: Black-box tool dialog

displayed in Figure 31. In particular, the user has to pick a unique tool name, specify the launch command, select the tool type (import, analysis, export) and write the launch documentation.

Note that the use of external *black-box* tools is discouraged, since most of the functionalities available using Eclipse plugins are not available.

Note also that plugin tools cannot be added, edited or removed via the manage tool dialog, since they are managed as standard Eclipse plugins. Plugin tools are installed and removed using the normal Eclipse procedure (**Help** > **Install New Software...** menu).

6.2 Trace Import

The trace import dialog (Figure 32) allows the user to import new traces into the system, by using one of the registered importer tools. The combo box on top of the dialog enables the selection of the importer. The controls displayed below the combo box change according to the selected importer. These controls allow the user to configure the importer and pass the needed input. For example, in Figure 32 we can see that the *Pajé Dump Importer* has a checkbox to enable double precision and a text field for passing trace files. Any tool specific message, triggered by a change in the input for example, is displayed in the text field at the bottom of the dialog (*Tool Message* group). The *OK* button becomes active only if the input specified is considered valid by the tool. Pressing *OK*, the import process is launched with the provided input. At the end of this process, the trace browser view is automatically updated with the new trace information.

6.3 Launch Analysis Tool

The launch analysis tool dialog (Figure 33) allows the user to launch one of the analysis tools registered in the system. This dialog has the same structure than the Import Trace dialog (see Subsection 6.2). Pressing *OK*, the selected analysis tool is launched with the provided input.

6.4 Trace Export

The launch export tool dialog (Figure 33) allows the user to launch one of the export tools registered in the system. This dialog has the same structure than the Import Trace dialog (see Subsection 6.2). Pressing *OK*, the selected export tool is launched with the provided input.

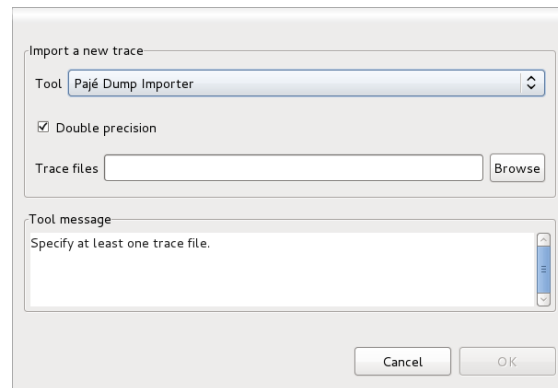


Figure 32: Import trace dialog

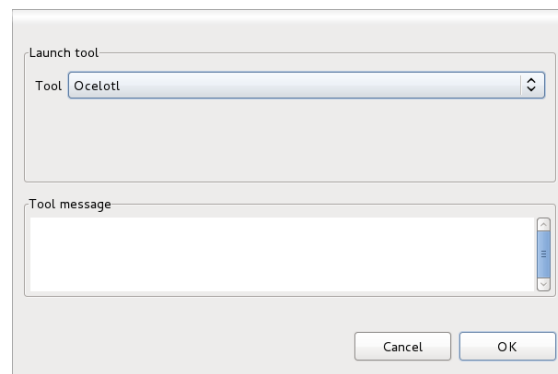


Figure 33: Launch analysis dialog

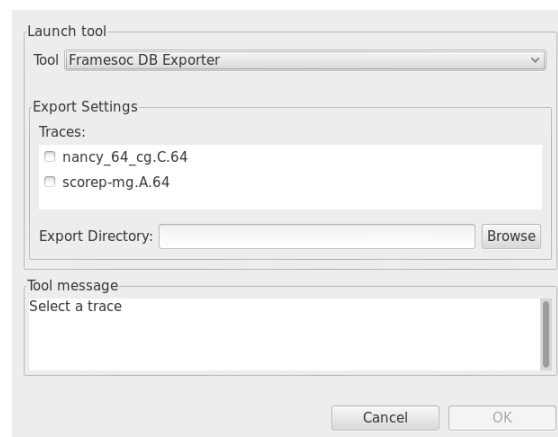


Figure 34: Export trace dialog

A Framesoc Configuration File

The Framesoc configuration file is located within the Eclipse installation directory. Its relative path to this directory is: `./configuration/fr.inria.soctrace.lib.utils/soctrace.conf`. Note that if the Eclipse directory does not have the write permission, the configuration file will be created in the user home directory with the same relative path as above. Please note that it is not recommended to modify this file manually, but preferably by using the *Preferences* dialog described in Subsection 6.1.

This file is automatically generated and mostly configurable using the GUI. It contains the following parameters:

soctrace_dbms : DBMS used by Framesoc. The accepted values are: `sqlite`, `mysql`.

mysql_db_user : MySQL database user name.

mysql_db_password : MySQL database password.

mysql_base_db_jdbc_url : MySQL database connection URL.

sqlite_db_directory : Directory containing the SQLite database files.

soctrace_db_name : Name of the Framesoc System DB.

max_view_instances : Maximum number of instances for a given type of analysis views.

trace_db_ts_indexing : Boolean stating if automatic time indexing is done at trace import.

trace_db_eid_indexing : Boolean stating if automatic indexing on event ID in `EVENT_PARAM` table

ask_for_tool_removal : Boolean stating if user confirmation must be asked when removing a tool and its results.

allow_view_replication : Boolean stating if it is possible to open more instances of the same type of view for a given trace.

References

- [1] SoC-TRACE project. <http://tinyurl.com/minalogic-soc-trace>.
- [2] Generoso Pagano and Vania Marangozova-Martin. SoC-Trace Infrastructure. Rapport Technique RT-0427, INRIA, November 2012.
- [3] Vania Marangozova-Martin and Generoso Pagano. Gestion de traces d'exécution pour le systèmes embarqués : contenu et stockage. Research Report RR-LIG-046, LIG, Grenoble, France, 2013.
- [4] Generoso Pagano and Vania Marangozova-Martin. FrameSoC Workbench: Facilitating Trace Analysis through a Consistent User Interface. Technical Report RT-0447, Inria, April 2014.
- [5] Lttng Eclipse Viewer. <http://lttng.org/eclipse>.
- [6] J.M. Wilson. Gantt Charts: A Centenary Appreciation. *European Journal of Operational Research*, 149(2):430–437, 2003.
- [7] Generoso Pagano and Vania Marangozova-Martin. SoC-Trace Infrastructure Benchmark. Rapport Technique RT-0435, INRIA, June 2013.



**RESEARCH CENTRE
GRENOBLE – RHÔNE-ALPES**

Inovallée
655 avenue de l'Europe Montbonnot
38334 Saint Ismier Cedex