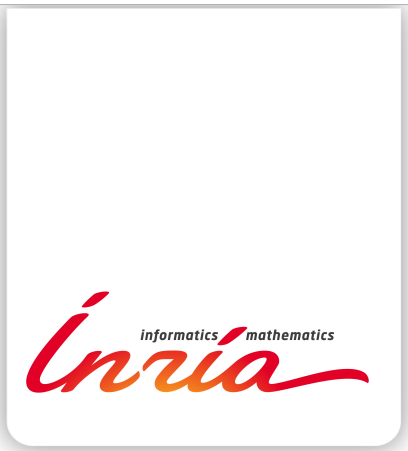
Framesoc User Guide [DRAFT]

Generoso Pagano

INRIA REPORT

December 2014

Project-Team MESCAL



Framesoc User Guide [DRAFT]

Generoso Pagano *

Project-Team MESCAL

Inria Report — December 2014 — 22 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, generoso.pagano@inria.fr

RESEARCH CENTRE
GRENOBLE – RHÔNE-ALPES

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

Table of contents

1 Introduction

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

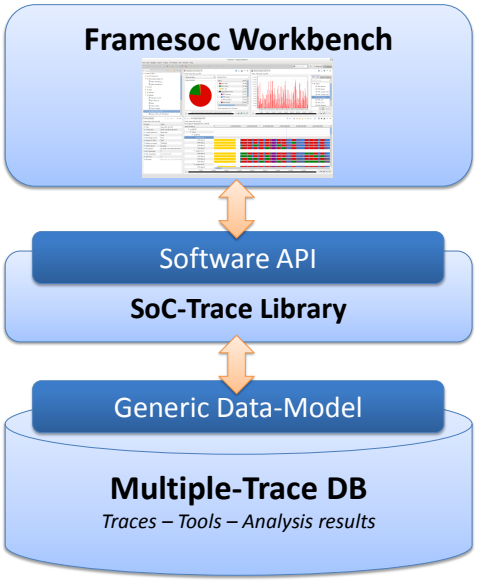


FIGURE 1 – Framesoc Architecture

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 thees 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 easyest 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

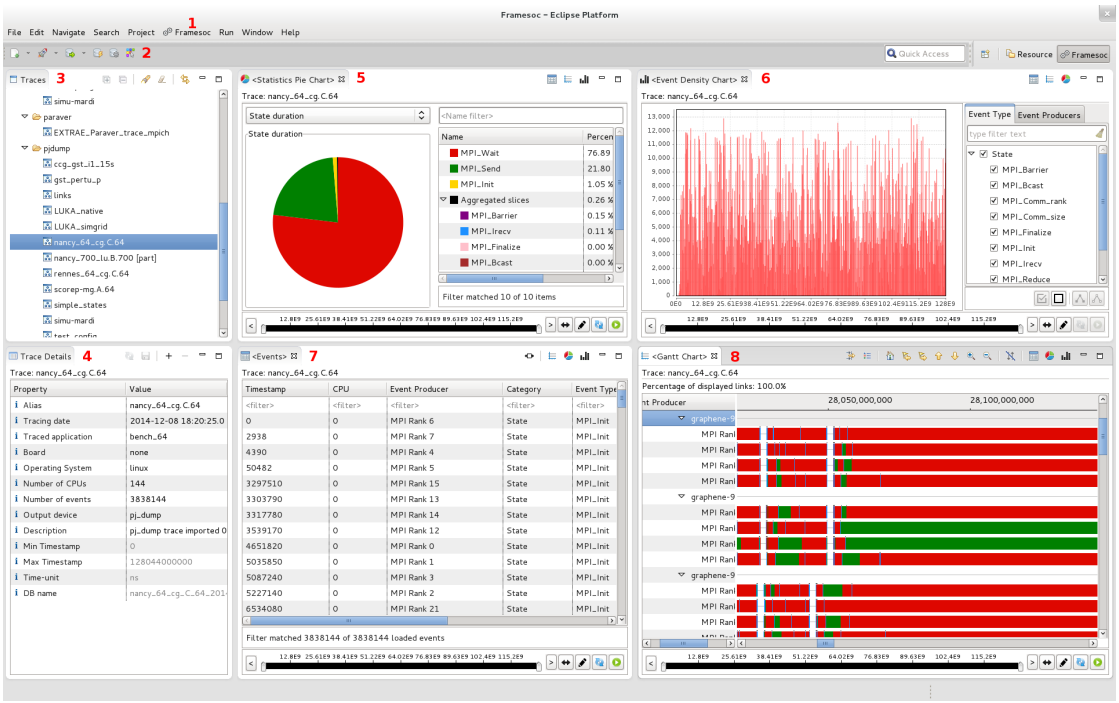


FIGURE 2 – Framesoc Eclipse Perspective

Framesoc provides an Eclipse perspective [1] 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

1. Within 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. 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 is surrounded by < and >. 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

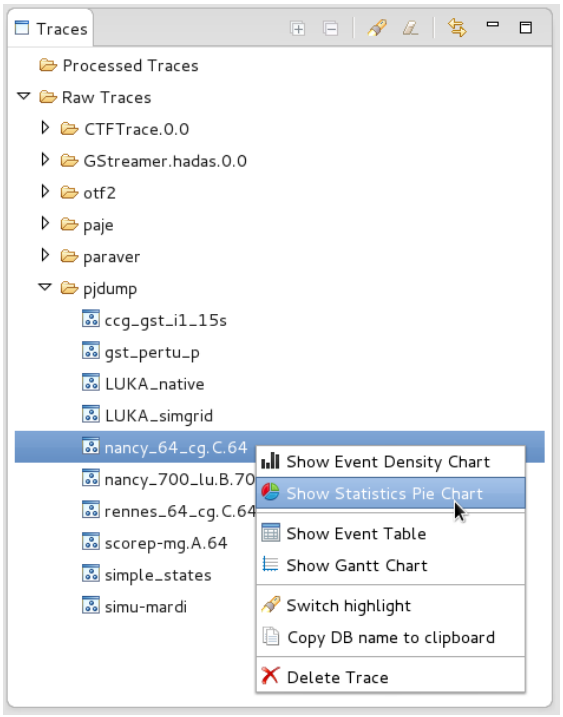


FIGURE 3 – Traces view

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

2. As 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.

Double-clicking on a trace opens the Event Density Chart for that trace (Subsection 5.3). 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.

If more traces are 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.

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 DB [3] (5).

4.1.1 Trace Filter

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

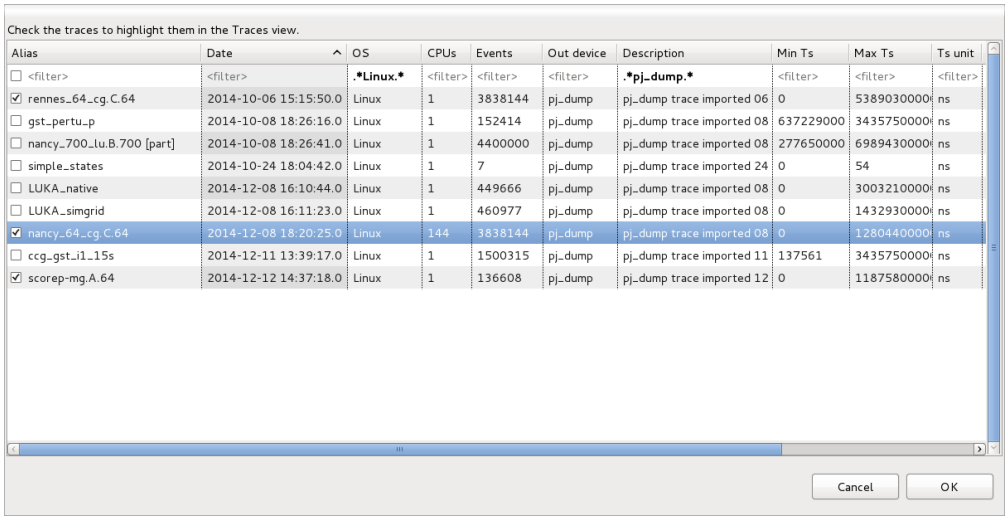


FIGURE 4 – Trace Filter Dialog

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

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

editable fields, allowing for filtering on the corresponding column content. Filtering can be done using regular expression. 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.

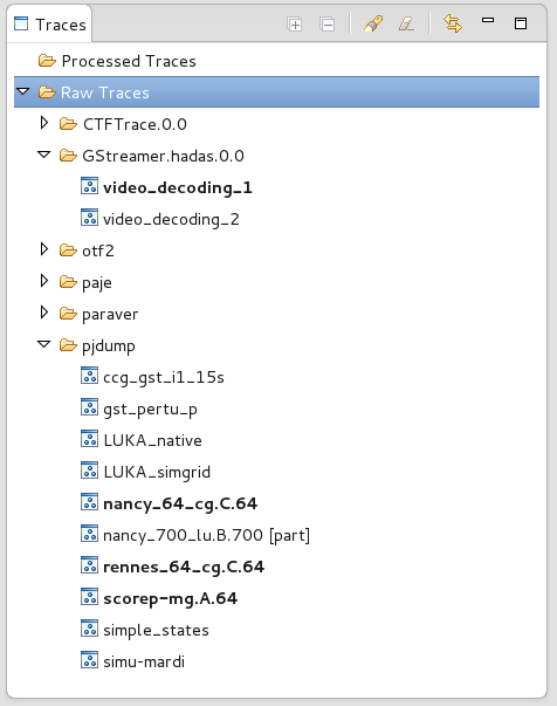


FIGURE 5 – Highlighted traces in Traces view

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 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).

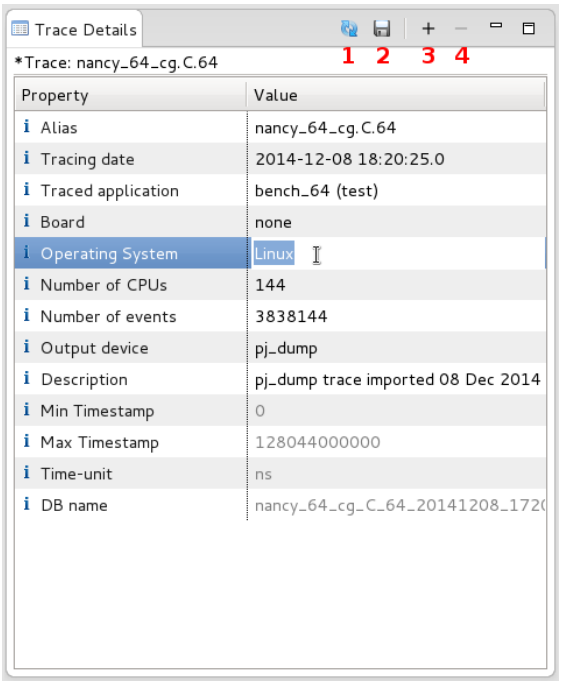


FIGURE 6 – Trace Details view

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, a dialog to add a new property is shown (Figure 7).

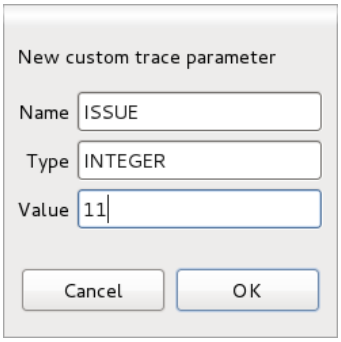


FIGURE 7 – Dialog to add a new custom property as trace metadata

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.

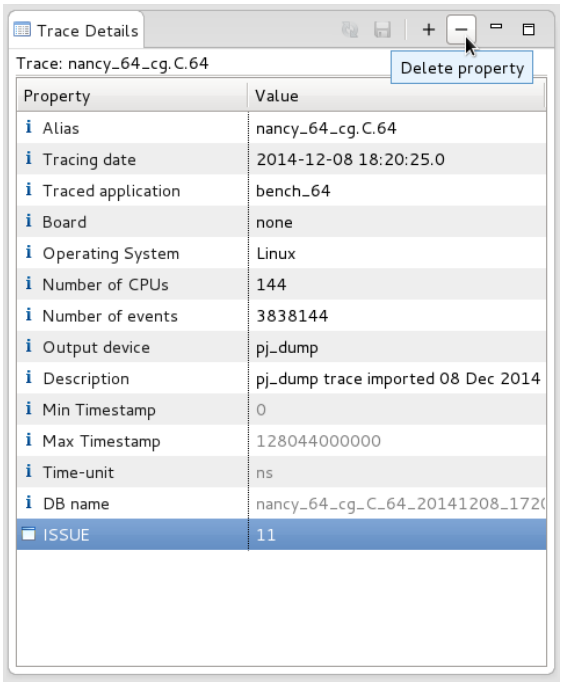


FIGURE 8 – Trace Details view after adding a custom property (ISSUE)

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/dezooming 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.)

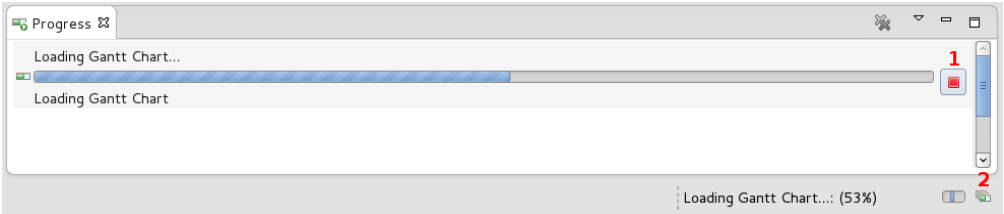


FIGURE 9 – Progress view enabling to cancel a loading job

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).

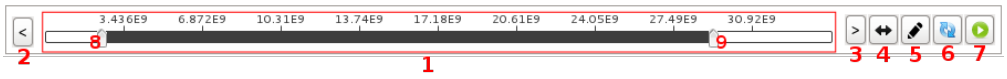


FIGURE 10 – Time management bar present in all Framesoc analysis view

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 two knobs of the double range slider (8 and 9) identify the portion of the trace (colored in black) actually loaded in the table. 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 has to change the width of the black bar using the knobs 8 and 9. 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.
- *Time window navigation*: this possibility involves using the two arrow buttons 2 and 3, to select a time window that has the same size of 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 all the trace duration.

View communication Using the Framesoc Bus (described in the RT-447 [4]), Framesoc views can communicate to 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 6.3) 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 12 shows all the possible *Show in...* buttons.

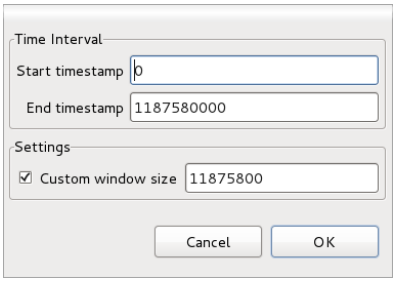


FIGURE 11 – Time window dialog

FIGURE 12 – *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, the a dialog is shown to the user in order to select one of the possible statistics operators (see Subsection 5.2). Button 3 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.

5.2 Statistics Pie Chart

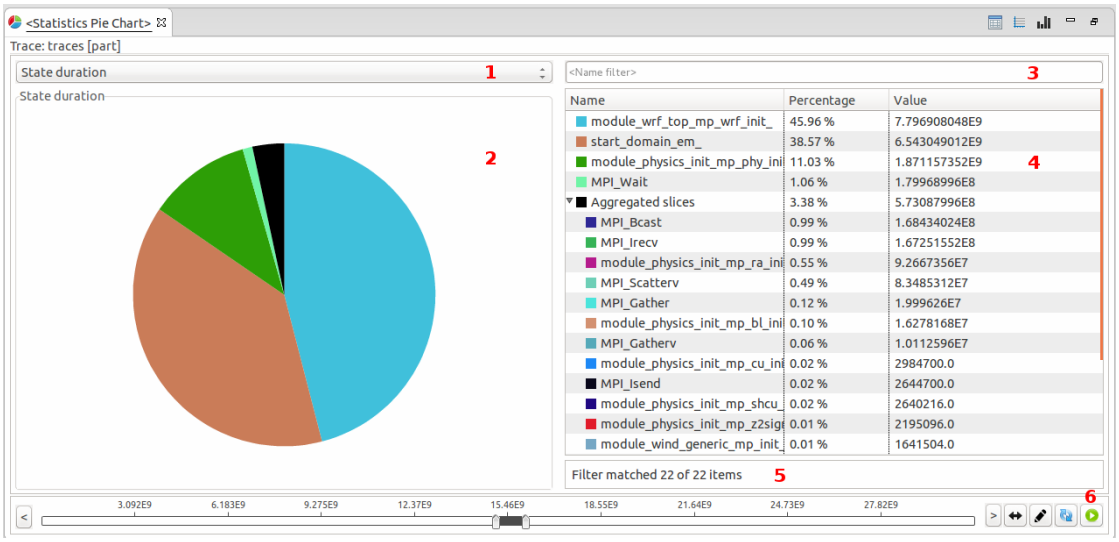


FIGURE 13 – Statistics Pie Chart view

The *Statistics Pie Chart* view (Figure 13) 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. 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.
- 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 13 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.

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.

5.3 Event Density Chart

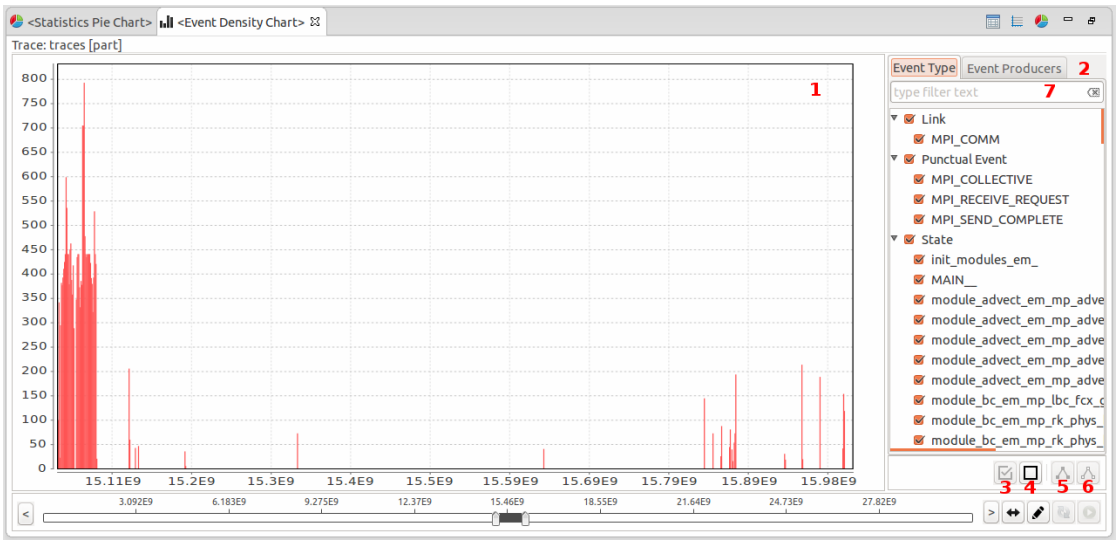


FIGURE 14 – Event Density Chart view

The *Event Density Chart* view (Figure 14) is a Framesoc analysis view displaying the event density over time in the form of a histogram.

The histogram is on the left of the view (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 and dezoom portions of the chart. In particular, to zoom a portion of the chart, the user has to click to the start of the portion, drag the mouse going to the right up to the end of the portion, then release the click. To completely dezoom, the user has to do the same as above, but dragging the mouse to the left this time. Hovering the mouse on a bin, some information about the bin is displayed (the central timestamp and the number of events).

On the right, there is a tab browser (2), containing two tabs: *Event Types* and *Event Producers*. The *Event Types* tab allows the user to select which event types will be used to generate the histogram. The *Event Producers* tab does the same for event producers. Both tabs present their content in a tree, reflecting the hierarchy of the displayed entities. The tree elements can be filtered using the text box indicated as 6. Each element of the tree can be checked/unchecked directly, by using the checkbox beside it. To check/uncheck more elements contextually, the user can use the buttons indicated as 3 (check all), 4 (uncheck all), 5 (check subtree), 6 (uncheck subtree). These buttons are enabled/disabled according to the element currently selected in the tree.

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.

5.4 Event Table View

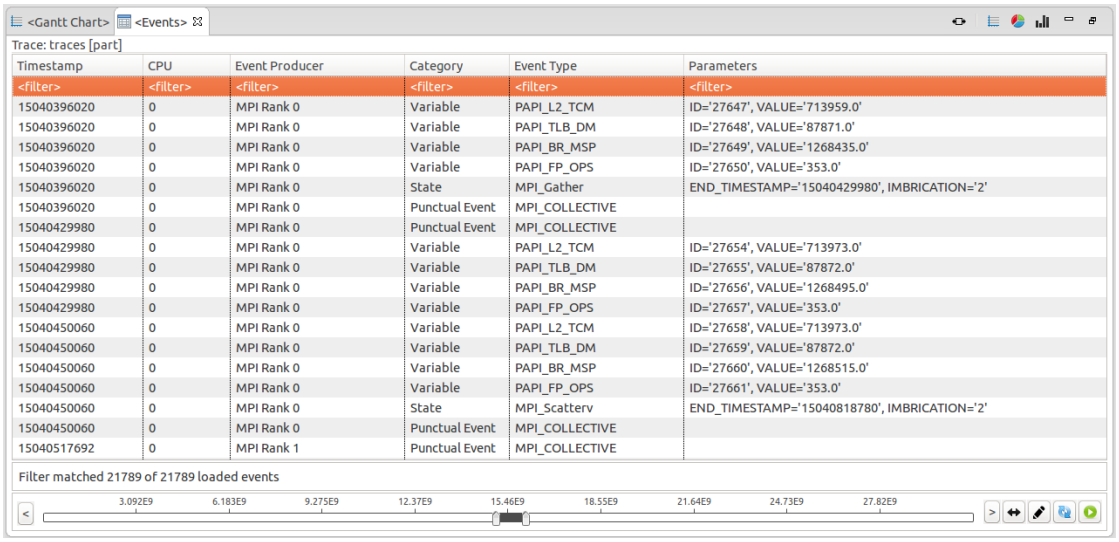


FIGURE 15 – Events view

The *Events* view (Figure 15) 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.
- Event Type: the event type name.
- Parameters: the list of the event custom parameters, with the format NAME='VALUE'.

Each column header, when clicked, triggers the sorting of the rows according to the corresponding column. The first row of the table contains an editable filter for each column. These filters accept regular expressions. For example, in Figure 15 only the events produced by a given producer are filtered. The number of events matched by the filter, over the totality of loaded events, is shown in the status bar under the table viewer.

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

The view toolbar contains an *adjust* button, which triggers column width resizing in order to fit the actual content.

5.5 Gantt Chart View

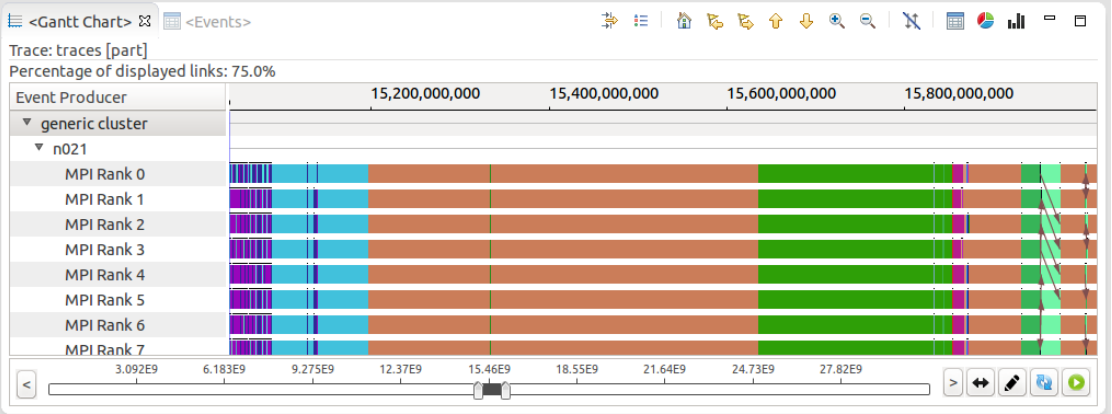


FIGURE 16 – Gantt Chart view

This view (Figure 16 shows a Gantt chart representation of trace events. This kind of representation is pretty common in trace analysis frameworks (e.g., Lttng 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 viewer [4], which is composed of two parts. On the left there is the hierarchy of event producers, grouped by CPU. 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 as colored rectangles). If there are links in the trace (e.g., communications), they are represented as oriented arrows. Note that the colors used to fill the rectangles depend on the event type. For a given event type, the same color is used in this Gantt view and in the Pie Chart view (Subsection 5.2). Note that hovering the mouse on a given state, triggers the displaying of a tooltip showing the event type name (Figure **??**).

The view toolbar contains the following buttons:

- Six buttons enable six different modes, where a mode defines the behavior of a mouse click on the time chart:
  - Select mode: it is possible to select with the mouse a time region in the time chart.
  - Drag mode: it is possible to drag the time chart with the mouse.

4. TODO cite trace compass

  - Horizontal Zoom In
  - Horizontal Zoom Out
  - Vertical Zoom In
  - Vertical Zoom Out
- The *adjust* button triggers the resize of content to fit the view size.
- The *clock* button enables/disables the time compression feature: if enabled, this feature hides large time regions without events.
- The *show all* button triggers the loading of the whole trace in the Gantt.
- The *table* button triggers the visualization of the current loaded time window in the Event Table view.

Under the Gantt viewer, there is a status bar displaying the trace time unit, the timestamp currently under the mouse cursor and the selected interval duration, if any.

Under the status bar, at the bottom of the view, there is a time management bar, which has exactly the same behavior as the one in the Event Table view (see Subsection 5.4 for the details). As it happens for the table, a new time window is loaded in the Gantt viewer only on demand, when the *play* button is pressed. Note that this time, given that it is possible to select a time region also on the Gantt viewer (if in selection mode), the selections in the viewer and in the double range slider are always synchronized, thus helping trace navigation.

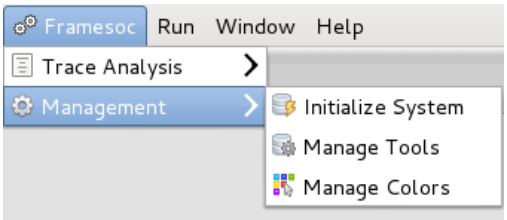
6 Framesoc Menu and Toolbar

When the Framesoc perspective is activated, the Framesoc menu (Figure 17) and the corresponding toolbar (Figure 18) are visible.

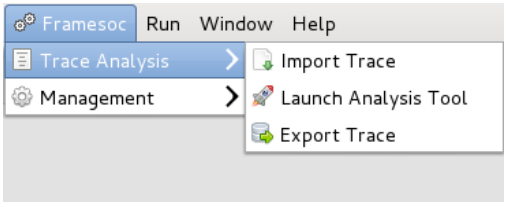
In the menu, the different functionalities are grouped in two categories: management and trace analysis. The management menu (Figure 17a) allows the user to access system configuration, tool management and color management. The trace analysis menu (Figure 17b) allows the user to launch importers, analysis 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.



(a) Management Menu



(b) Trace Analysis Menu

FIGURE 17 – Framesoc Menu

FIGURE 18 – Framesoc Toolbar

6.1 System Initialization

At system initialization, the user accesses a configuration wizard, whose first page is shown in Figure 19. This page allows the user to select the DBMS [5] 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.

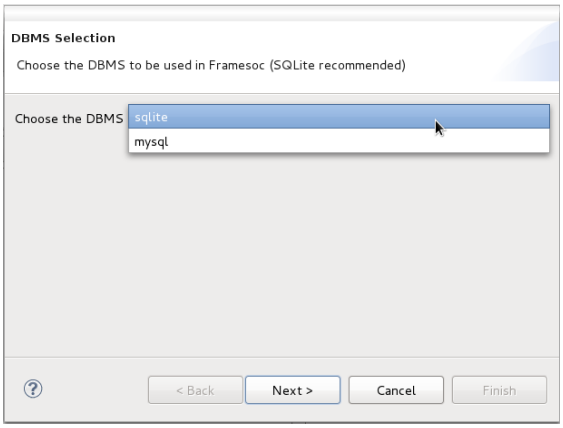


FIGURE 19 – System Initialization: DBMS selection dialog

Once the user has chosen the DBMS and pressed *Next*, he comes to the DBMS configuration page (Figure 20), which is different for each DBMS. If SQLite is selected (Figure 20a), the user has simply to enter the directory where he wants the database files to be kept. Otherwise, if MySQL is chosen (Figure 20b), the user has to specify some connection parameters (user-name, password, URL).

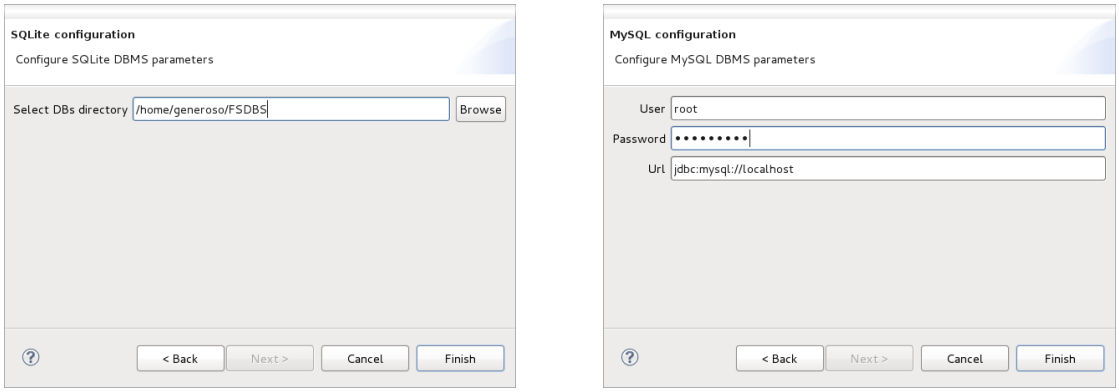
In both cases, after pressing *Finish*, the Framesoc storage subsystem is correctly configured. If a System DB already exists, it is reused, otherwise a new one is created. This configuration is saved in the Framesoc configuration file (Appendix A).

Note that after the initialization, 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 runtime [6], 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.

Note also that at each Framesoc startup, the system automatically checks that the storage configuration is good. If it is not the case, the system initialization wizard is automatically launched.

5. Data Base Management System
6. As 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.
7. The user is asked for confirmation if the `ask_for_tool_removal` variable is set to `true` in Framesoc configuration file (see Appendix A)



(a) SQLite configuration dialog

(b) MySQL configuration dialog

FIGURE 20 – System Initialization: DBMS configuration

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

6.2 Tool Management

The tool management dialog (Figure 21) simply displays the list of tools registered to the system, enabling the installation, modification and removal of external *black-box* tools (tools that are not eclipse plugins).

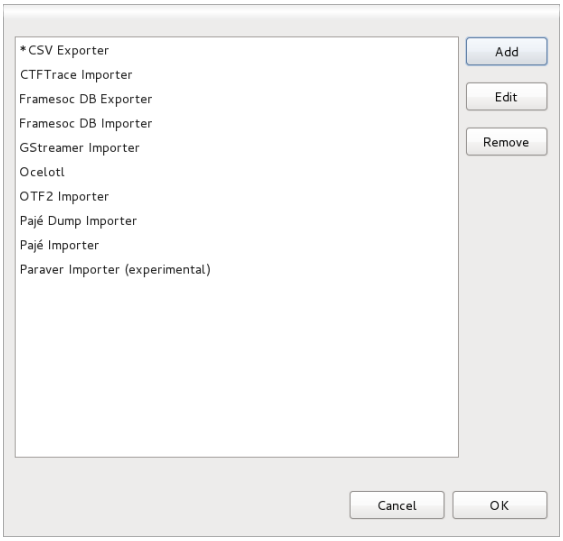


FIGURE 21 – Tool manager

Adding or modifying a *black-box* tool requires the user to edit the different fields of the dialog displayed in Figure 22. In particular, the user has to pick a unique tool name, specify the launch command, select the tool type (import, analysis, export), write the launch documentation.

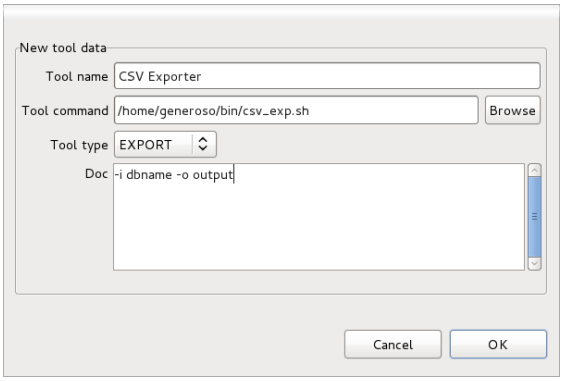


FIGURE 22 – Black-box tool dialog

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).

6.3 Color Management

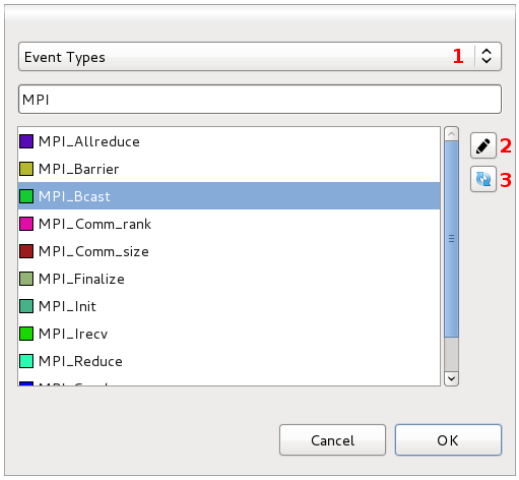
The color management dialog (Figure 23a) allows the user to modify the colors associated to event producers and event types in a centralized way for the whole workbench. The combo box at the top of the dialog (1) enables the selection of the entity (event producer or event type). The list below enumerates 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 23b). Pressing *OK* saves all changes. Pressing the *reset* button (3), reverts unsaved changes. 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.

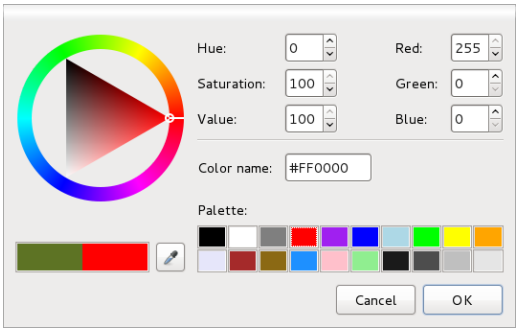
6.4 Trace Import

The trace import dialog (Figure 24) allows the user to import a new trace into the system using one of the registered importer tools. The user selects the importer from the combo box at the top, then he specifies the trace files (if more than one file is needed, all the files should be selected in the browser dialog opened when the *Browse* button is pressed). If the importer requires additional parameters (the *Doc* field normally provides this information), the user specifies these parameters too. At the botton of the dialog, there is an *Error Message* zone, intended to inform the user about any error in the input.

Finally, after 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.



(a) Color management dialog



(b) Color editing dialog

FIGURE 23 – Framesoc Color Management

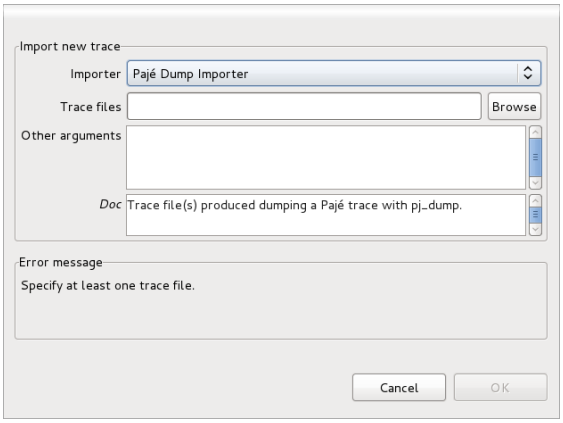


FIGURE 24 – Import trace dialog

6.5 Launch Analysis Tool

The launch analysis tool dialog (Figure 25) allows the user to launch one of the analysis tools registered to the system. The user selects the analysis tool from the combo box at the top. If the tool requires additional parameters (the *Doc* field normally provides this information), the user specifies these parameters too. At the botton of the dialog, there is an *Error Message* zone, intended to inform the user about any error in the input. Finally, after pressing *OK*, the tool is launched with the provided input.

6.6 Trace Export

The trace export dialog (Figure 26) works exactly as the launch analysis tool dialog, with the difference that this time the user can launch one of the exporter tools registered to the system.

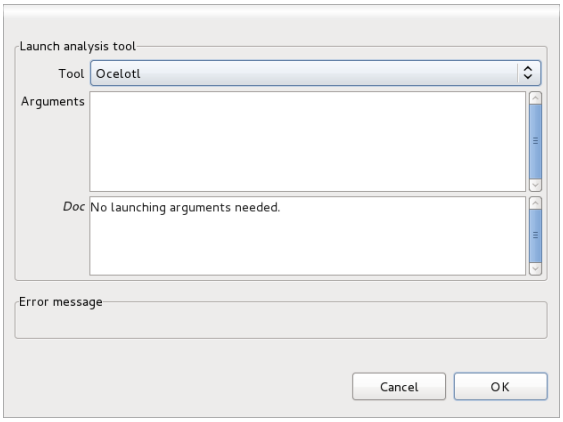


FIGURE 25 – Launch analysis dialog

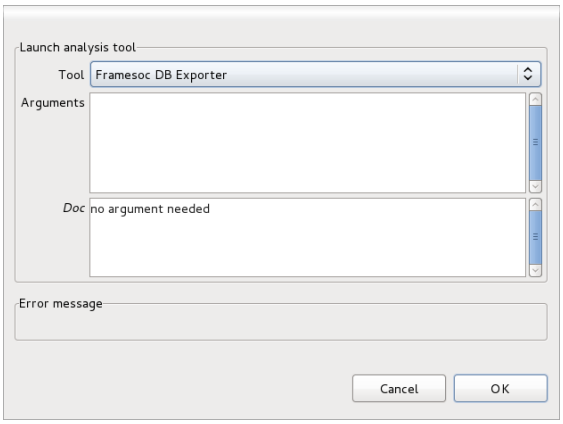


FIGURE 26 – Export trace dialog

A Framesoc Configuration File

The Framesoc configuration file (`.soctrace.conf`) is located in the user home directory. It is automatically generated and mostly configurable using the GUI.

The file 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 user name.

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

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

References

[1] SoC-TRACE project. `http://tinyurl.com/minalogic-soc-trace`.

[2] Generoso Pagano and Vania Marangonzova-Martin. SoC-Trace Infrastructure. Rapport Technique RT-0427, INRIA, November 2012.

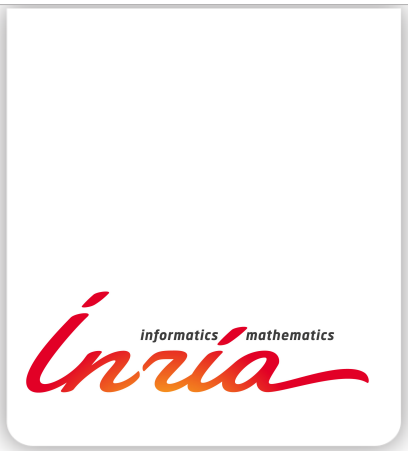
[3] Vania Marangozova-Martin and Generoso Pagano. Gestion de traces d'exécution pour le systèmes embarqués : contenu et stockage. Research Report RR-LIG-046, LIG, Grenoble, France, 2013.

[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