Ocelotl User Guide

Youenn Corre
Damien Dosimont

Ocelotl User Guide

Youenn Corre [*]
Damien Dosimont [†]

Project-Team MOAIS

Abstract: This guide describes the Ocelotl tool, as available in the github master repository `https://github.com/dosimont/ocelotl`.

[*] INRIA, youenn.corre@inria.fr
[†] INRIA, damien.dosimont@imag.fr

Table of contents

1   Introduction

Ocelotl is a tool to explore, analyze and visualize execution traces through aggregation techniques. It is part of the SoC-Trace project, and as such, it is integrated in the FrameSoC framework as an Eclipse plugin.

2   Launching Ocelotl
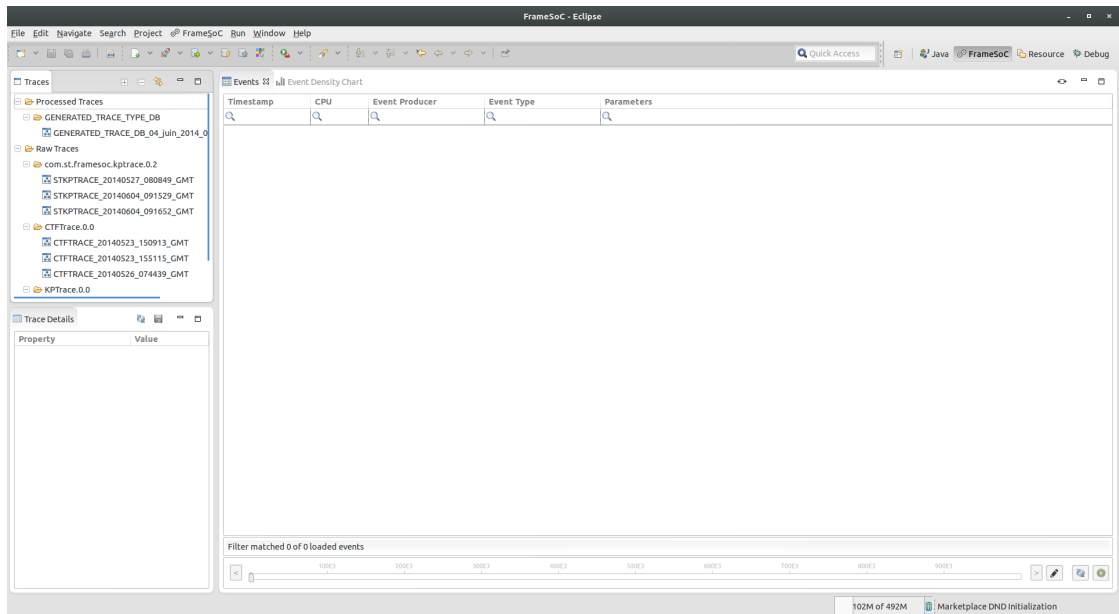
Figure 1: FrameSoC Eclipse Workbench at startup

In order to launch Ocelotl:

1. - Go to the FrameSoC menu, choose the *Trace Analysis* entry and click on *Launch Analysis Tool*.
   - Or click on the *Analyze Trace* icon.
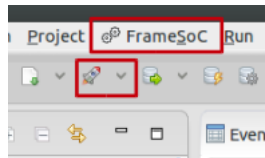
Figure 2: The FrameSoC menu and the *Analyze Trace* icon

2. Select the Ocelotl tool.

3. No argument is needed so just click *OK*.

The Ocelotl tool view should open, as illustrated in Fig. 4.

Figure 3: Ocelotl launching window

Figure 4: Ocelotl at startup

3 Ocelotl Features

In this section, we describe how to load a trace, launch the analysis and explore the results by modifying parameters.

Figure 5: Trace configuration interface

3.1   Loading a trace

3.1.1   Select a trace

In the top right corner of Ocelotl, the *Trace Overview* tab allows to select a trace to analyze (element 1 in Fig. 5). It shows the traces already exported with Framesoc.

3.1.2   Microscopic Description Settings

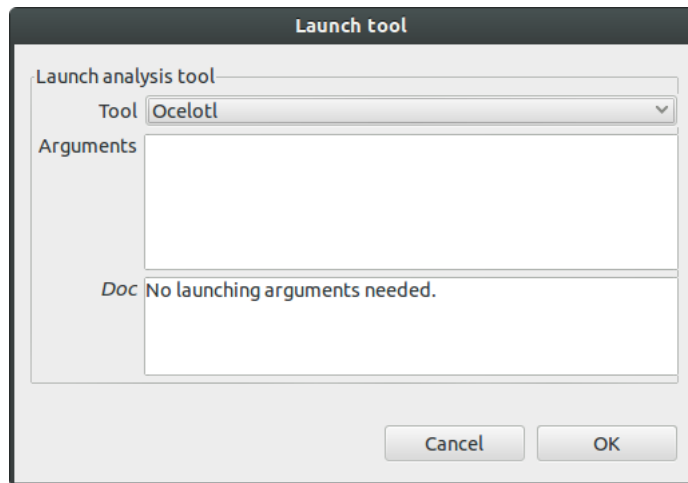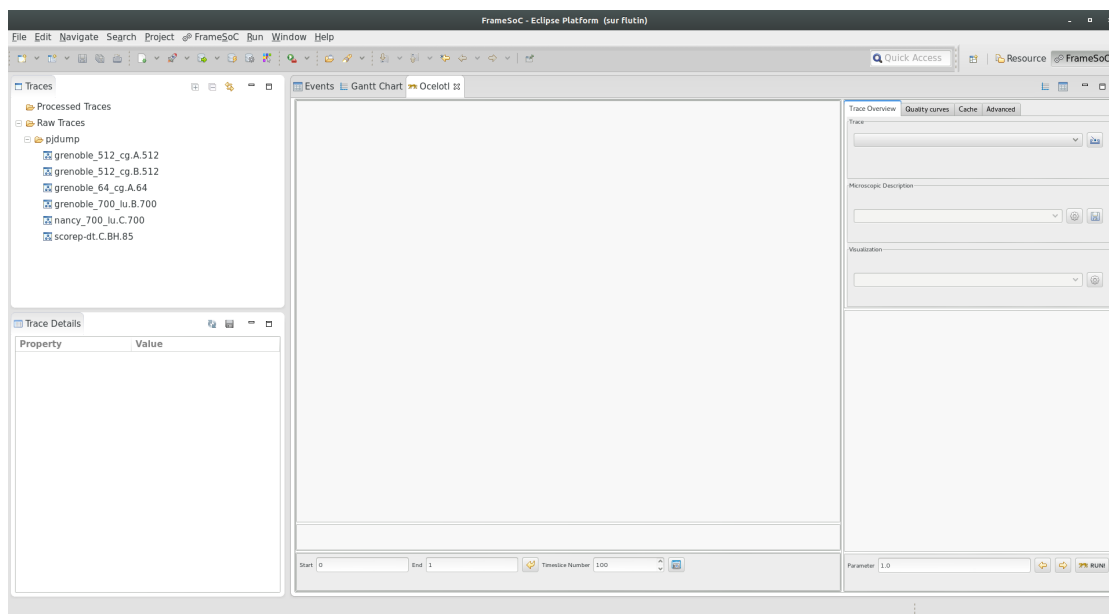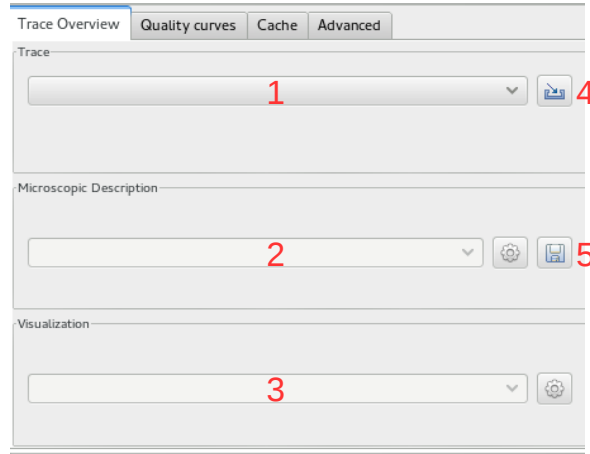Once a trace is selected, the *Microscopic Description* combo becomes available (element 2 in Fig. 5). It displays the operators compatible with the current trace. At the moment, the available operators are:

- The *Event Distribution*, that performs an analysis based on the event occurrences. This type of analysis is compatible with all the traces.

- The *State Distribution*, that performs an analysis based on the event duration. This analysis requires the trace to contains events of the type state.

- The *Variable Distribution*, that performs an analysis based on the variation of the value of the event over time. This analysis requires the trace to contains events of the type variable.

Moreover, all of these operators have a spatio-temporal equivalent (*Hierarchical Event Distribution*, *Hierarchical State Distribution* and *Hierarchical Variable Distribution*) which take into account the spatial dimension as well (i.e. the hierarchy of event producers).

Once the type of distribution selected, it is possible to configure the analysis by clicking on the *Settings* button, which opens a setting window (cf. Figure 6).
   By default, all the event types and all the event producers are selected. The available settings are:

- in the *Event Types* tab, the filtering of the event types involved in the analysis;

- in the *Event Producers* tab, the filtering of the event producers (the resources that produce the events) involved in the analysis. The producers are shown with their hierarchy, and
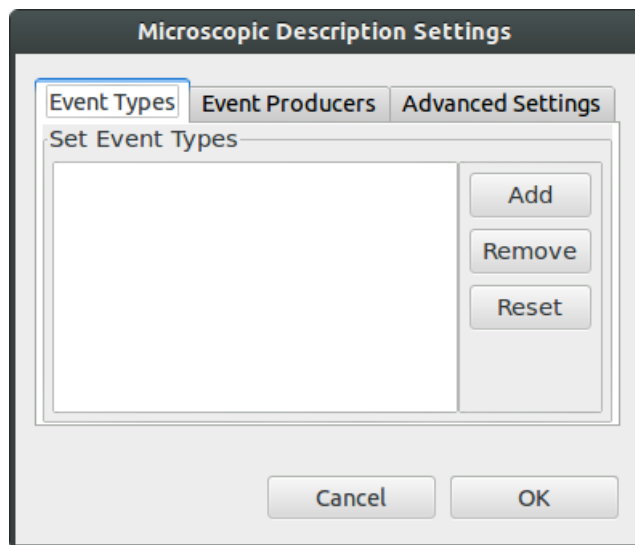
Figure 6: Microscopic Description Settings

depending on whether the operator required a valid hierarchy or not[1], the deselection of a parent node might lead to the deselection of its children nodes. The *Add Result* button enables to load a set of event producers saved as an analysis result by another tool;

3.1.3 Visualization Settings

Once a distribution is selected, the *Visualization* combo (element 3 in Fig. 5) provides a list of available views, in accordance with the compatibility of the previously selected time operator. The currently available choices are:

- *Parts*: for a visualization showing aggregates that correspond to an homogeneous behavior in the trace. The aggregates are separated by a blank space and all have a different color. In *Settings*, the available parameters are:

  - *Aggregate parts*: gathers time slices that belong to the same aggregate. If not active, each time slice is separated, and their aggregation is shown thanks to their color.

  - *Show part number*: attributes a number to each aggregate to help distinguishing them.

- *Proportions*: for a visualization showing the same aggregates as in *Parts*, but also the proportion of event occurrences (or the ratio of state duration, if state distribution was selected in the previous step) for each aggregate. In this visualization, each event/state is separated vertically in each aggregate and their proportion is shown by making the height of each event/state in an aggregate proportional to its occurrence/duration. The events/states that have a proportion too small to be visible, are aggregated and displayed on top of the aggregate as a dotted line. In the *Settings* window, you can customize the color of each event.

- *Matrix*: is the equivalent of the Parts view but with the addition of the spatial dimension.

---

[1]The spatio-temporal operators take into account the hierarchy, while the temporal ones do not.

- *Mode*: is the equivalent of the Proportions view but with the addition of the spatial dimension.

3.2   Quality Curves

Ocelotl displays curves representing the different aggregation levels, as shown in Fig. 7. The X axis is the aggregation ratio from 1 (fully aggregated) to 0 (maximal level of disaggregation). The green curve shows the level of displayed complexity and the red is the level of displayed information. The vertical blue line shows the current value of the parameter.
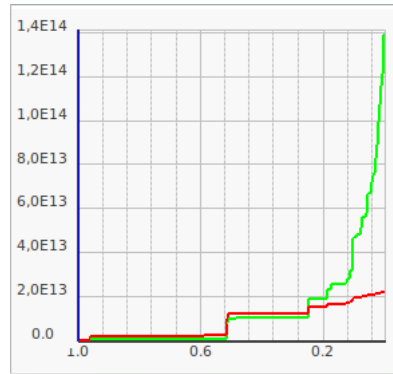

Figure 7: Quality curves in rising order.

3.3   General Settings

Next to the *Trace Overview* tab, three other tabs provide more configuration parameters:

3.3.1   Cache Settings

To prevent time-consuming loading of traces from the database, we use a cache system to accelerate access to the trace raw data. If the cache is enabled in the settings, then first time a trace is analyzed, a cache file is generated. A cache file is a plain text file that contains non-null values of the trace, and each cache is specific to an aggregation operator, and a specific time region.
Several settings are available to configure the cache:

- *Cache Enabled*: allow to enable or disable the cache.

- *Empty Cache*: delete all the cache files in the current cache directory.

- *Datacache directory*: specify the directories where the cache files are stored.

- *Maximum datacache size*: specify the maximum cache size in megabytes. When the maximum size is reached, the cache files with the oldest accessed dates are deleted until enough space is available. Specifying a value of -1 means that there is no maximal limit to the cache size.

- *Cache time slice*: the first time a trace is loaded, a cache file is generated — if the cache is enabled. This parameters specifies the number of time slices used for the generated cache. A higher number means a more accurate cache, but also a bigger cache file.

- *Cache policy*: specify the rebuilding policy used to rebuild the data from the cache file when a "clean" rebuilding is not possible[2]. The available choices are:

  ◦ *Fast*: the dirty time slices are rebuilt by multiplying the values proportionally to the division of the cache time slice between the two new time slice (e.g. for a time slice which is split with 30% of its time in one time slice and 70% in the other, its values will be multiplied by 0.3 and 0.7 respectively). While this is a fast method, it may not be accurate in all cases (especially in cases with high ratio of dirty time slices).

  ◦ *Accurate*: the dirty slices are rebuilt by performing a request to the database so that the events are rebuilt precisely. Depending on the event nature (punctual event, state, etc.), the rebuilding can be pretty slow.

  ◦ *Automated*: try to automatically choose the best strategy between *Fast* and *Precise*.

  ◦ *Ask me*: show a dialog to the user at each run, from which he can select a strategy.

3.3.2 Quality Curves settings

The tab *Quality Curves* enables to change the appearance of complexity and information curves provided by the aggregation algorithm:

- *Normalize Qualities*: normalizes the curves and scale them on the interval $[0 : 1]$, which improves their readabilty. **Note:** normalization should not be used with spatio-temporal operators since it provides incorrect results.

- You can choose between two representations of the curves: rising curves (complexity gain, information gain) or decreasing curves (complexity reduction, information loss).

- The *Threshold* parameter determines the quality curves precision. The more its value decreases, the more the precision increases and the more aggregation steps are found. However, setting a too small value may lead to bad performances, so we advice to keep the default value.

3.3.3 Advanced Settings

In the *Advanced Settings* tab, you can configure some advanced settings[3]:

- *Event Number Retrieved by Threads*: set how much events are loaded by threads for each iteration during the trace reading.

- *Event Producers per Query*: enables to divide the query into several queries with a fixed number of event producers. This functionality is useful if the disk space is low, as the temp directory is used to store the query result during the trace reading. The default value is 0 which corresponds to fetching all producers in one query.

- *Working Threads*: set the number of active threads during the trace reading.

---

[2]A "clean" rebuild means that no time slice of the cache is split inside two time slices of the currently built microscopic description (such time slices are called "dirty" time slices). Typically it is clean when the number of time slices of the cache is dividable by the current number of slices (for example, if the cache was built with 10 000 time slices and the current rebuilt uses only 100 time slices, then one time slice of the new model will be built each from 100 time slices of the cache. But if the current rebuild is using 150 time slices, then some time slices of the cache would be split between two of the new time slices. The rebuild strategies are thus used for those "dirty" cases.

[3]**Note:** Modifying these values might decrease the trace reading performances.

3.4   Time settings

In order to tune the time settings, several parameters can be modified in the bar under the graph display area (Figure 8).
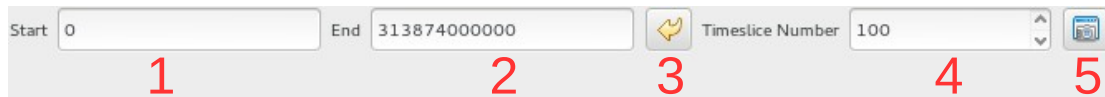

Figure 8: Time Settings

3.4.1   Time Stamps

You can change the trace time bounds thanks to the *Start* and *End* fields (1 and 2 resp. in Figure 8) at the bottom of the window. By default, after loading a trace, the values are the starting and ending timestamps of the trace. If an area of the graph is selected, or if a zooming operation is performed, the displayed values are the starting and ending timestamps of the selected area. The *Reset* (3) button enables to get the original values back.

3.4.2   Time Slices

The *Timeslice Number* field (4) corresponds to the number of time slices that will be used to compute the aggregation. It is advised to tune this parameter in order to find a granularity that is convenient to the analysis. As the aggregation algorithm complexity is dependent on this parameter, it is advised to increment it progressively to keep good performances.

3.5   Analysis settings

To analyze the trace, several parameters allows to explore different levels of aggregation and view. The aggregation settings are located under the *Quality curves* display.

Figure 9: Aggregation Settings

3.5.1   Run an Analysis

After having set the configuration, click on RUN! (3 in Fig. 9) (or press ENTER on the keyboard). The aggregation is then computed. This operation can take a while. When it is finished, the visualization and the curves are shown, as illustrated in Fig. 10.

3.5.2   Change the Aggregation strength

In order to find a relevant aggregation, you can tune the *Parameter* setting. This parameter allows to explore different aggregation levels, by navigating between different balances of the information loss and the gain in complexity. There are three ways to do this:

1. Click on the curves. The corresponding parameter value is retrieved, and used to show the associated aggregation.

2. Use the < and > buttons (2) (or use left and right arrows on the keyboard). It increments or decrements the parameter.

3. Enter a value manually, in the *Parameter* field (1). The value must be between 1 and 0.

A value of 1 means that the aggregation strength is maximal: all the trace is aggregated. On the opposite, a value of 0 means that each time slice is disaggregated. An intermediate value provides a compromise between both extrema.

3.5.3   Zoom

Once you obtain an interesting aggregation level, you may want to focus on a particular time area. It is possible to zoom in on the trace by selecting an area by clicking and dragging with the left button of the mouse. Then, click on RUN! or press ENTER to display the selected area. In order to zoom out, you must click on *Reset* and then on RUN!, to go back to the global view of the trace.

3.5.4   Switch to another view

It is possible to display the trace in a Gantt chart or in an event table by clicking on the corresponding button in the top right corner. The newly opened view will show the events of the trace that are within the time bounds shown in the *Start* and *End* fields. You can focus on an area by selecting it with the mouse left button drag and then switching directly to another view, without loading the aggregation with the RUN! button.

3.6   Graph Display

Figure 10 shows an aggregated trace, with the proportion view. At the bottom of the graph, a time axis shows the time stamps of the currently displayed events. Events of different types have different colors. By letting the mouse over a part of the graph for a short time, a tooltip note shows the type of events represented by that part of the graph. For the event view, this note also displays the number of occurrences of the events divided by the duration of the aggregate[4]. For the state view, it is the state duration divided by the aggregate duration.

3.7   Load and save a cache file

3.7.1   Load

An alternative way to load a trace, instead of using the combo box, is to load it from its cache file, by using the button to the right of the combo box (element 4 in Fig. 5). It opens a new window, allowing to select a cache file. If the selected cache file is valid and corresponds to a trace currently in the Framesoc database, then the trace is automatically loaded with the settings saved in the cache file.

---

[4]The time unit of the trace is displayed in the *Trace Details* tab on the left bottom of the screen.
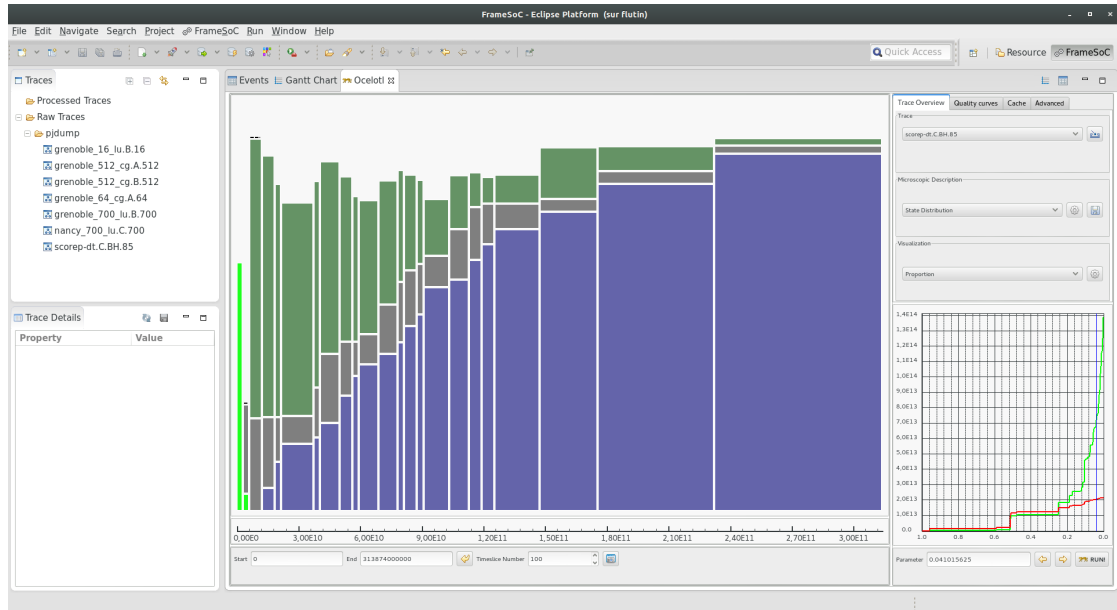
Figure 10: A graph showing an aggregated trace

3.7.2 Save

Once a run has been performed on a trace, a cache file is generated, in order to speed up the next analysis of the trace. It is then possible to save this cache to a specific location, in order to get a fast access later. This is done by pressing the button to the far right of the *Microscopic Description* combo box (element 5 in Fig. 5).

3.8 Take a snapshot

In order to save the currently displayed diagram and curve, it is possible to save them as images along with the current parameters of the trace in text file. This is done by pressing the button at the far right of the bottom bar under the display (element 5 in Fig. 8).

3.9 Known Issues

You may need to resize manually the different views, according to your resolution.