

# PPS User Guide

Version v0.20.0-M2

# Table of Contents

1. PPS Workbench	2
1.1. Preferences	4
2. TMSc textual editor	5
2.1. Project Explorer view	5
3. TMSc graphical viewer	6
3.1. Timed message sequence chart view tab	6
3.1.1. Rendering strategies	7
3.1.2. Relations to (Message) Sequence Charts and Flame Graphs	7
3.2. Selection view tab	9
3.3. Project Explorer view	9
3.4. Outline view	9
3.5. PlantUML view	10
3.6. Data Analysis view	10
3.6.1. Time series viewer	10
3.6.2. Histogram viewer	10
4. Common Viewer Functionality	11
4.1. Panning and Zooming	11
4.2. Selection in Chart	13
4.3. Measure Time	13
4.4. Export chart to PNG file	14
4.5. Synchronize Timing	14
4.6. Dynamic Inclusion Filtering	16
5. Release Notes	18
5.1. PPS v0.20	18
5.2. PPS v0.0.3	18
5.3. PPS v0.0.2	18
5.4. PPS v0.0.1	18

This user guide describes how to use the Platform Performance Suite (PPS).



# Chapter 1. PPS Workbench

PPS is an Integrated Development Environment (IDE) based on [Eclipse](#). For common information about the Eclipse Workbench, please read the [Eclipse user guide](#). The figure below shows a [screenshot of the PPS IDE](#) with the most important editors and views for PPS. The remainder of this section explain these views.

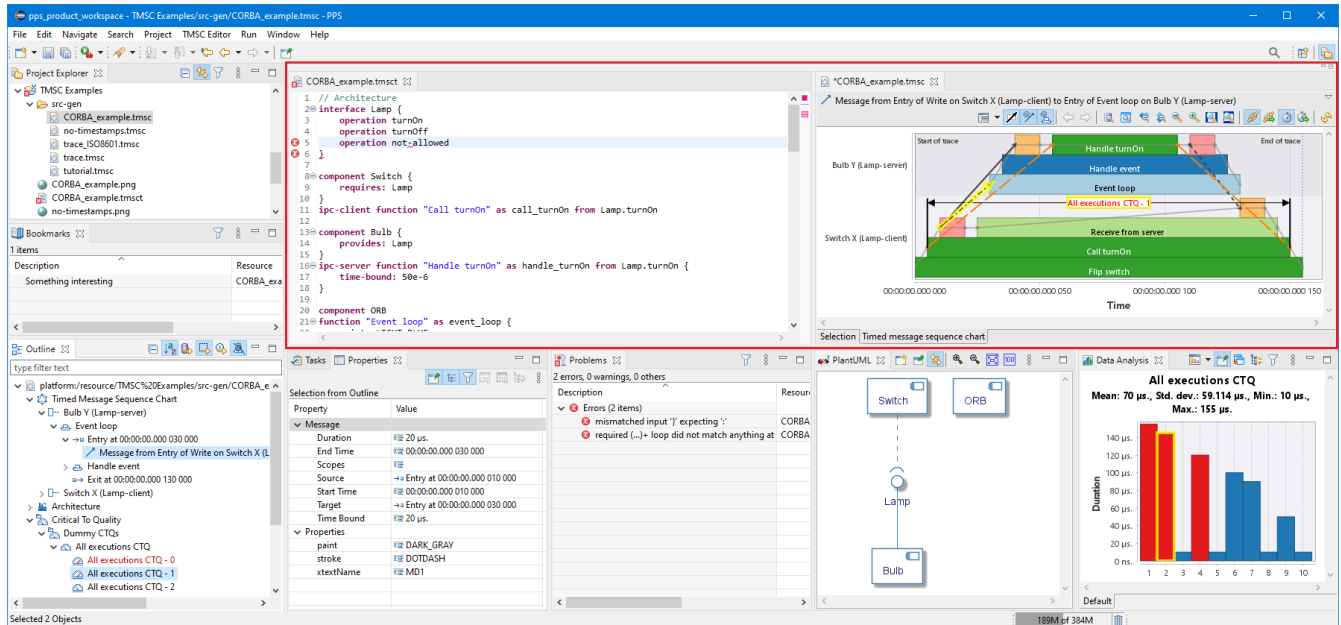


Figure 1. A screenshot of the PPS IDE

## Project Explorer view

Within Eclipse, the files to work with are part of a project. The available projects are shown in the [Project Explorer view](#). Each project is simply a directory on the hard disk, and can be expanded to show the files and directories it contains. By double clicking a file, it is opened in the [editor area](#) by the default editor that is associated with that file.

The files in the Project Explorer view support contextual actions by means of the [Context menu](#). The supported actions for a file are explained in more detail in the section of the editor that is associated with that file.

## Editor area

The editor area is marked as a red rectangle in a [screenshot of the PPS IDE](#). In the screenshot, the editor area contains two editors side-by-side. By default editors will be opened stacked, but editors can be moved by means of dragging the tab of an editor to another location.

The left editor shows the [TMSM textual editor](#) and the right editor shows the [TMSM graphical viewer](#), that both are explained in more detail in their corresponding sections.



Opening a textual editor and graphical viewer side-by-side is useful as the changes in the textual editor will be reflected in the graphical viewer upon save.

## Bookmarks view

Bookmarks are a simple way to navigate to resources that are used frequently. The [Bookmarks view](#) displays all bookmarks in the Workbench.

## Outline view

The [Outline view](#) displays an outline of a structured file that is currently open in the [editor area](#), and lists structural elements. The contents of the Outline view are editor specific and therefore explained in more detail in the section of its editor.

## Properties view

The [Properties view](#) displays property names and values for a selected item such as a [Message](#) in [a screenshot of the PPS IDE](#). The contents of the Properties view are editor specific and therefore explained in more detail in the section of its editor.



By default the Properties view is stacked behind the [Tasks view](#).

## Problems view

As you work with resources in the workbench, various builders may automatically log problems, errors, or warnings in the [Problems view](#). For example, when you save a [textual TMSC](#) file that contains syntax errors, those will be logged in the Problems view. When you double-click the icon for a problem, error, or warning, the editor for the associated resource automatically opens to the relevant place.



By default the Problems view is stacked behind the [Tasks view](#).

## PlantUML view

The plantuml library implements a DSL and renderer for many UML diagrams (class, sequence, objects, states, activities, ...). See <http://plantuml.com> for more info about the DSL and renderer. The [PlantUML view](#) integrates plantuml's functionality into Eclipse and displays contextual UML diagrams for a selected item such as an [Architecture](#) in [a screenshot of the PPS IDE](#). The contents of the PlantUML view are editor specific and therefore explained in more detail in the section of its editor.

## Data Analysis view

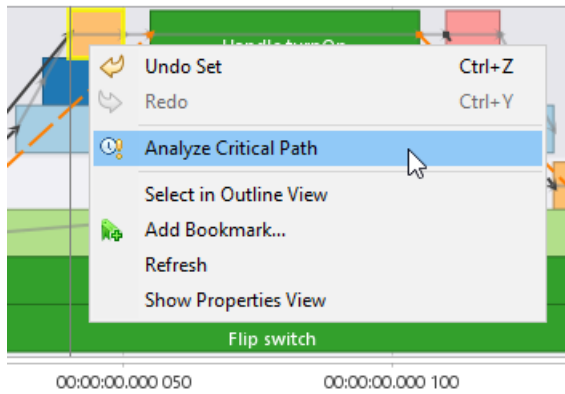
The Data Analysis view displays statistical information for a selected item such as a [Time series viewer](#) for a [metric instance](#) in [a screenshot of the PPS IDE](#). The contents of the Data Analysis view are editor specific and therefore explained in more detail in the section of its editor.



By default the Data Analysis view is stacked behind the [PlantUML view](#).

## Context menu

Many views in the PSS IDE support [selection](#) and with that, contextual actions based on the selection. The context menu can be visualized by right-clicking the mouse in any part of the view that has the selection. The context menu shows all available actions for both the view and its selection.

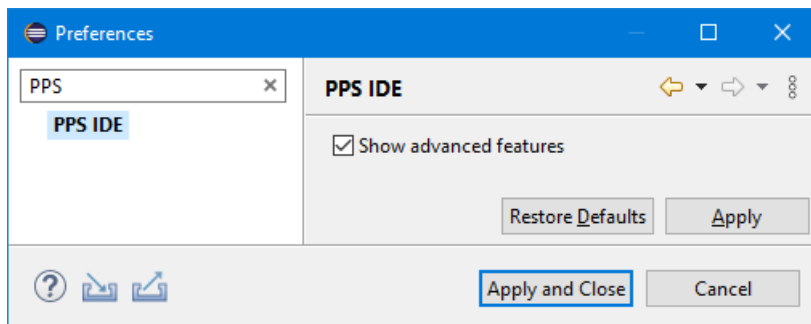


## 1.1. Preferences

Use the command [Preferences](#) dialog pages to set how you want Eclipse to operate. The menu path is **Window** › **Preferences** on Windows and Linux - on macOS the Preferences can be found in the **App** menu. The filter input at the left top can be used to easily navigate to the PPS settings.



The '*Show advanced features*' option enables experimental features in PPS. These features may be unstable, undocumented and might produce unexpected results. All editors should be closed and re-openend after changing this setting for the changes to take effect.



# Chapter 2. TMSC textual editor

## 2.1. Project Explorer view

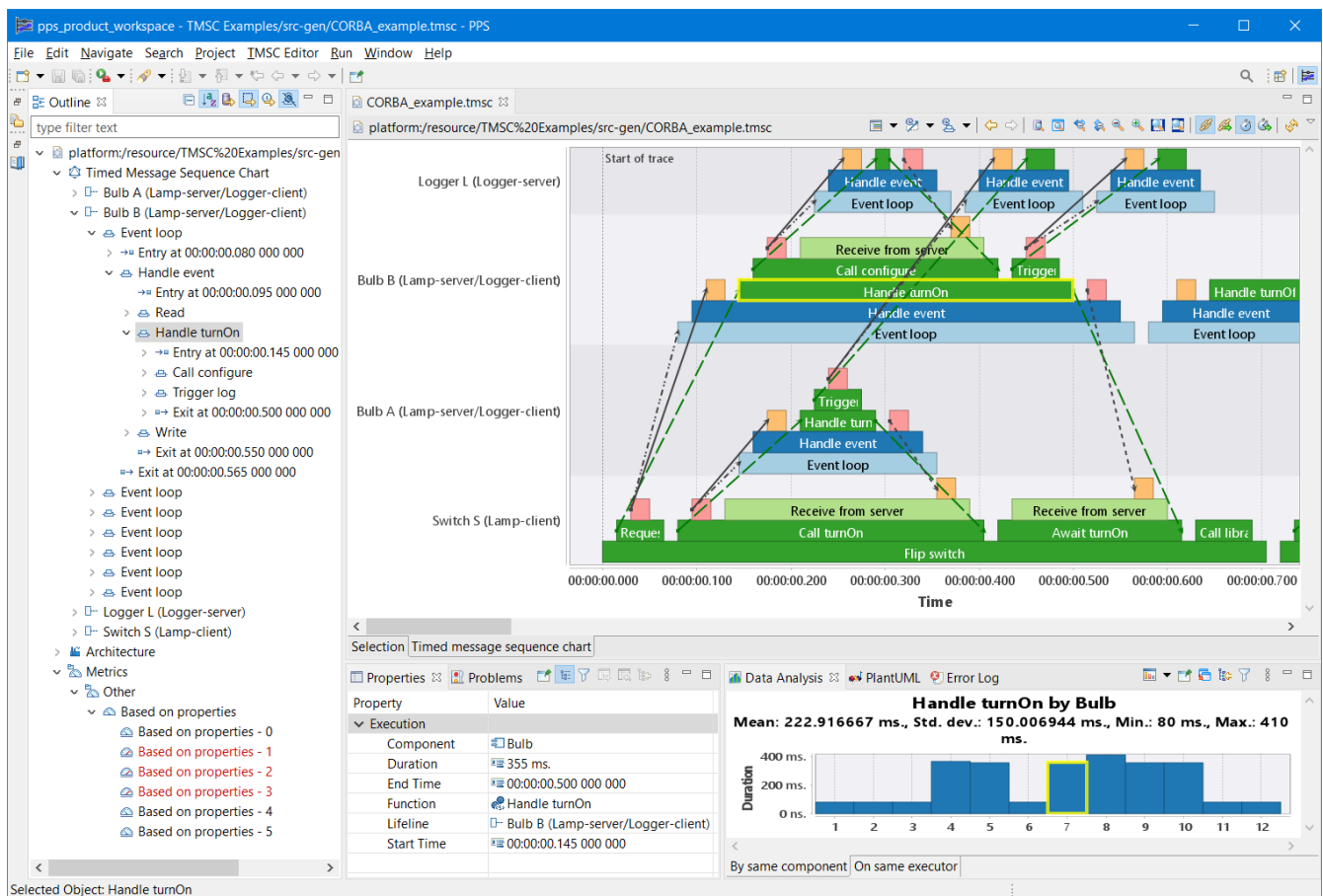
The `tmsc.t` file extension is associated with the TMSC textual editor.

# Chapter 3. TMSC graphical viewer

## 3.1. Timed message sequence chart view tab

The runtime behavior and timing of software applications can be visualized using Timed Message Sequence Charts (TMSC). In a TMSC, the life-line of an executor is shown on as a horizontal swimming lane, and (real) time is shown along the horizontal axis. The executor labels are shown on the vertical axis. The function calls that are executed by an executor are shown as bars within the swimming lane. The bars are labeled with their execution name. Nested function call executions are drawn as stacked bars, resulting in a call stack. Arrows between the bars denote the data and/or control flow dependencies. The arrows indicate where executors are communicating with each other, and where executor local events are raised and handled. Section [Relations to \(Message\) Sequence Charts and Flame Graphs](#) compares TMSCs to similar representations.

Due to [dynamic inclusion filtering](#) some executors might not be shown; however, the visible executors might communicate with this hidden executors. Communication text balloons are shown in this case, depending on the [selected dependencies visibility](#). The arrow from/to the balloon indicates the data flow of the communication and the text in the balloon describes the other side.



It is possible to select executions or dependencies by clicking on them, also see section [Selection in Chart](#). Selected elements are shown with a thicker yellow border. Once selected, the details of the execution can be seen in the [Properties](#) view and in the [Data Analysis](#) view.



By default, the *Properties* view displays normal properties. To view expert













properties click the  [ **Show Advanced Properties** ] button to reveal them.

The *Timed message sequence chart* view tab supports some common functions to support analysis of its data, like:

- Dynamic filtering based on the *Outline* view selection, see Section [dynamic inclusion filtering](#).
- Select executions and dependencies by clicking on them, see section [Selection in Chart](#). Once selected, details can be seen in the *Properties* view and in the [Data Analysis view](#).
- Measuring the time between executions, see Section [Measure Time](#).
- Synchronize the current visible time window with all other supported editors, see Section [Synchronize Timing](#).
- Place a [bookmark](#) at a specific area within a file.
- Saving a snapshot of the chart, see Section [Export chart to PNG file](#).
- Panning and zooming, see Section [Panning and Zooming](#).

On top of these common functions, the *TMSC* tab has the following additional action bar buttons.

Icon	Action	Description
	<b>[ Select Rendering Strategy ]</b>	Changes the appearance of the TMSC depending on the analysis that is applied to the TMSC. Please read section <a href="#">Rendering strategies</a> for more information. Please note that the icon depends on the current selection and therefore might be different than illustrated here.
	<b>[ Select Dependencies Visibility ]</b>	 Show all dependencies.  Hide communication text balloons on dependencies.  Also hide dependencies that cross the current visible area of the chart. In other words, dependencies that do not start or end within the visible area.  Hide all dependencies.
	<b>[ Select Executions Visibility ]</b>	 Show all executions.  Hide communication text balloons on executions.  Hide all executions.

### 3.1.1. Rendering strategies

### 3.1.2. Relations to (Message) Sequence Charts and Flame Graphs

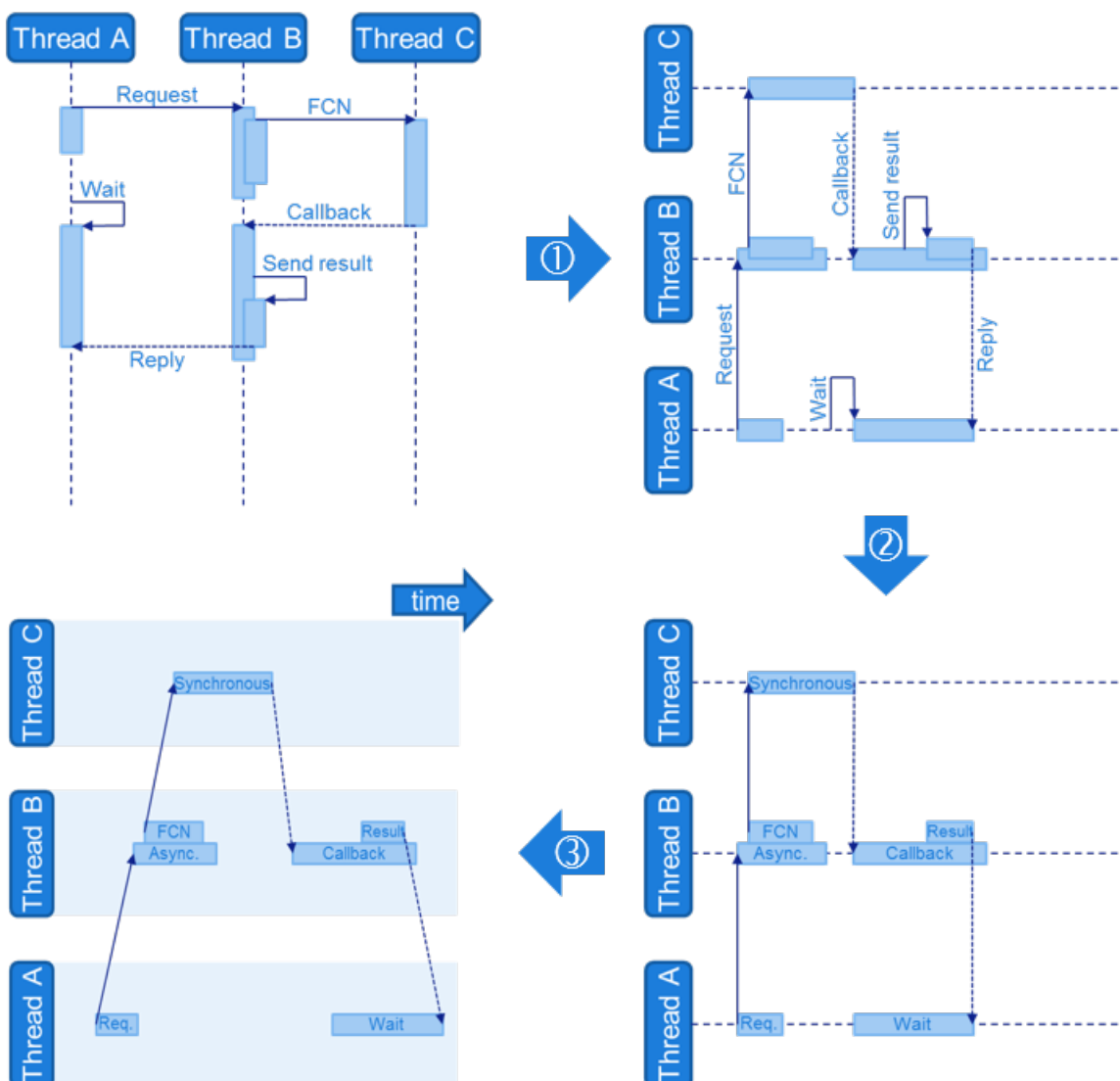
To give some more intuition about Timed Message Sequence Charts, they are compared to similar representations.

Timed Message Sequence Charts are similar to UML Sequence Diagrams and SDL Message Sequence

Charts. The main differences are:

- The horizontal axis and vertical axis are swapped such that life-lines are positioned horizontally, and the vertical axis shows the life-line labels.
- The life-lines represent autonomous execution entries, similar as in SDL.
- Life-lines are shown as swim-lanes.
- Arrows represent asynchronous messages, i.e. the sending executor continues execution directly.
- Overlapping executions are drawn as stacked boxes, instead of overlapping boxes.
- Labels are shown inside the executions, instead of on the arrows.
- The horizontal axis shows the (real) time, such that the length of an execution represents the actual duration of that execution, and the distance between the start and the end of an arrow represent the latency of message passing.
- Fragments, such as loops, alternatives and references, are not allowed in Timed Message Sequence Charts.

These differences are illustrated with the picture below.



Timed Message Sequence Charts are also similar to Call Charts or Flame Charts. It can be argued that Timed Message Sequence Charts have a Flame Chart for each executor. These Flame Chart are connected by arrows to show inter-executor relations. Note that, Flame Charts should not be confused with Flame Graphs, Flame Graphs put the alphabet on the horizontal axis instead of the time, [see Brendan Gregg's pages](#).



## 3.2. Selection view tab

The *Selection* view tab is similar to the *Outline* view. It shows the model in a tree view. Clicking an element in the tree selects the element and shows the details of that element in the *Properties* view.

This view supports dynamic filtering based on the *Outline* view selection (see section [dynamic inclusion filtering](#)) to filter elements of interest.

## 3.3. Project Explorer view

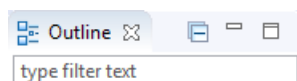
The `tmsc` and `tmscz` file extensions are associated with the TMSC graphical viewer. These files support the following contextual actions:

Icon	Action	Description
	[ Analyze metrics... ]	TODO
	[ Time-shift... ]	TODO


## 3.4. Outline view




The *Outline* view supports dynamic filtering of its elements, by means of entering a [search pattern](#) in its filter text box.









The *Outline* view shows the model in a tree view. Clicking an element in the tree selects the element and shows the details of that element in the *Properties* view.

By selecting elements in the tree both the resource and time context of an editor view can be reduced, see also Section [dynamic inclusion filtering](#). With reduction of the resource context is meant that the editor area will show only content which is related to the selected *Outline* elements. For example, when a life-line is selected, the [TMSC view tab](#) view will only show this life-line and the life-lines that it directly communicates with. This feature is active if the  **[ Push Context ]** toggle bottom is selected.

When the  **[ Push Time Range ]** toggle button is selected, the editor area will automatically show the time range which includes the time context for the selected *Outline* elements.

The model elements in the content outline tree support the following contextual actions:

Elements	Icon	Action	Description
TMSC		[ Analyze metrics... ]	TODO
TMSC Scoped-TMSC		[ Analyze slack ]	TODO
Scoped-TMSC		[ Save as TMSC ]	TODO
Metric Metric-Instance		[ Add metric annotations to TMSC ]	TODO
Measurement Metric-Instance		[ Analyze critical path ]	TODO
Metric-Instance ( <i>exceeding budget</i> )		[ Analyze root cause ]	TODO

## 3.5. PlantUML view

## 3.6. Data Analysis view

### 3.6.1. Time series viewer

### 3.6.2. Histogram viewer

# Chapter 4. Common Viewer Functionality

This chapter describes functionality that is commonly available for viewers in PPS.

## 4.1. Panning and Zooming

The chart viewers supports zooming and panning, as follows:

### *Zooming, using the mouse*

- To zoom in, click in the chart area and drag towards the right and downwards. A rectangle shows the selected area. Once the mouse is released, the chart is zoomed in, only showing the selected area.
  - Hold the **Ctrl** key while clicking and dragging to zoom horizontally only.
  - Hold the **Ctrl** + **Shift** keys while clicking and dragging to zoom vertically only.
- To zoom out, click in the chart area and drag to the left. Release the mouse to zoom all the way out, to show the entire chart.
  - Hold the **Ctrl** key while clicking and dragging to zoom out horizontally only.
  - Hold the **Ctrl** + **Shift** keys while clicking and dragging to zoom out vertically only.

### *Zooming, using the mouse wheel*

Scroll up to zoom in and scroll down to zoom out. The location of the mouse will be used as focus area, meaning that the chart will be zoomed in or out around that point.

- Hold the **Ctrl** key while scrolling with mouse wheel to zoom horizontally.
- Hold the **Ctrl** + **Shift** keys while scrolling with mouse wheel to zoom vertically.

### *Zooming, using the keyboard*

First click somewhere in the chart to activate keyboard focus. While holding the **Ctrl** key, the arrow keys can be used to zoom the chart in or out. Both the normal arrow keys as well as the arrow keys on the numpad can be used.

- Use the **→** or **←** key to zoom in or out horizontally.
- Use the **↓** or **↑** key to zoom in or out vertically.

In addition to the arrow keys, also 'browser style' zooming is supported.

While holding the **Ctrl** key:

- Use the **+** key to zoom in both horizontally as vertically.
- Use the **-** key to zoom out both horizontally as vertically.
- Use the **0** key to zoom all the way out, showing the entire chart.

### *Panning, using the scroll bars*

- Use the vertical scroll bar or mouse wheel to scroll vertically.
- Use the horizontal scroll bar to scroll horizontally.

### *Panning, using the mouse*












Hold the **Alt** key while clicking and dragging the chart in the direction to pan.





### *Panning, using the keyboard*

First click somewhere in the chart to activate keyboard focus. The arrow keys can be used to pan the chart in the desired direction. Both the normal arrow keys as well as the arrow keys on the numpad can be used.

Right click on the chart to show a menu with even more options.

Furthermore, the action bar for time based views show additional buttons to zoom.


Icon	Action	Description
	<b>[ Zoom to fit all the data in the data set ]</b>	<div>Zoom out to show all the data in the current view.</div> <div> This action can take much time as PPS typically works with large data sets.</div>
	<b>[ Zoom to fit all content ]</b>	<div>Zoom out to show as much content as possible (considering performance) in the current view.</div>
	<b>[ Zoom to fit all content horizontally ]</b>	<div>Zoom out to horizontally show as much content as possible (considering performance) in the current view.</div>
	<b>[ Zoom to fit all content vertically ]</b>	<div>Zoom out to vertically show as much content as possible (considering performance) in the current view.</div>
	<b>[ Zoom out ]</b>	<div>Zoom out a bit.</div>
	<b>[ Zoom in ]</b>	<div>Zoom in a bit.</div>
	<b>[ Zoom to current selection ]</b>	<div>This will zoom the view to show the active selection only.</div> <div> Only available if <a href="#">Selection in Chart</a> is available for this viewer.</div>
	<b>[ Zoom to fit the time of the current selection ]</b>	<div>This will zoom the view to show the time range for the active selection.</div> <div> Only available if <a href="#">Selection in Chart</a> is available for this viewer.</div>
	<b>[ Zoom to fit the height of the current selection ]</b>	<div>This will zoom the view to show the height for the active selection.</div> <div> Only available if <a href="#">Selection in Chart</a> is available for this viewer.</div>

Icon	Action	Description
	<b>[ Back ]</b>	<p>This will navigate back to the previous visible area in the navigation history.</p> <div>  <p>The navigation history will be automatically cleared when new input is set to the chart.</p> </div>
	<b>[ Forward ]</b>	<p>This will navigate forward to the next visible area in the navigation history.</p> <div>  <p>The navigation history will be automatically cleared when new input is set to the chart.</p> </div>

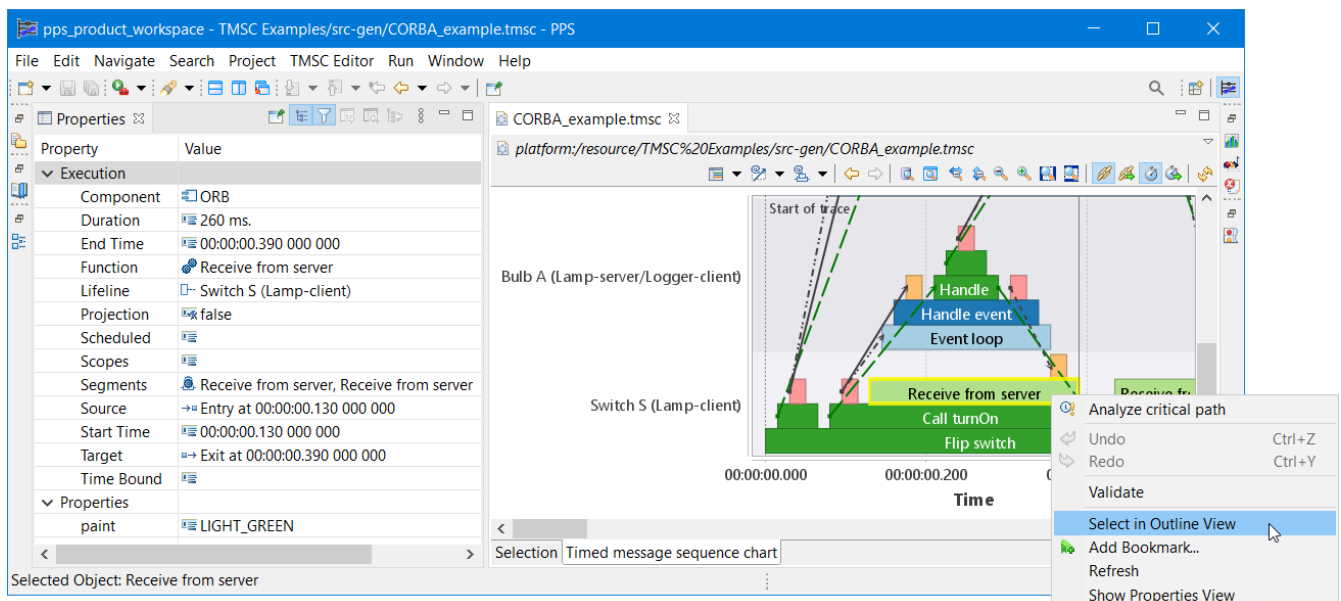
## 4.2. Selection in Chart

It is possible to select some of the items in a chart by clicking on them. If an item supports selection and is selected, it is shown with a thicker yellow border. Once selected, the details of the item can be seen in the *Properties* view.



By default, the *Properties* view displays normal properties. To view expert properties click the  **[ Show Advanced Properties ]** button to reveal them.



A context menu is available to select the current chart selection in the *Outline* view. This might be helpful for views that support [dynamic inclusion filtering](#).



The screenshot shows the TMS Editor interface. On the left, the **Properties** view is open, displaying execution properties for the selected object 'Receive from server'. The main area shows a sequence diagram with lifelines 'Bulb A (Lamp-server/Logger-client)' and 'Switch S (Lamp-client)'. A context menu is open over the 'Receive from server' message, with the option 'Select in Outline View' highlighted. The menu also includes 'Analyze critical path', 'Undo', 'Redo', 'Validate', 'Add Bookmark...', 'Refresh', and 'Show Properties View'.

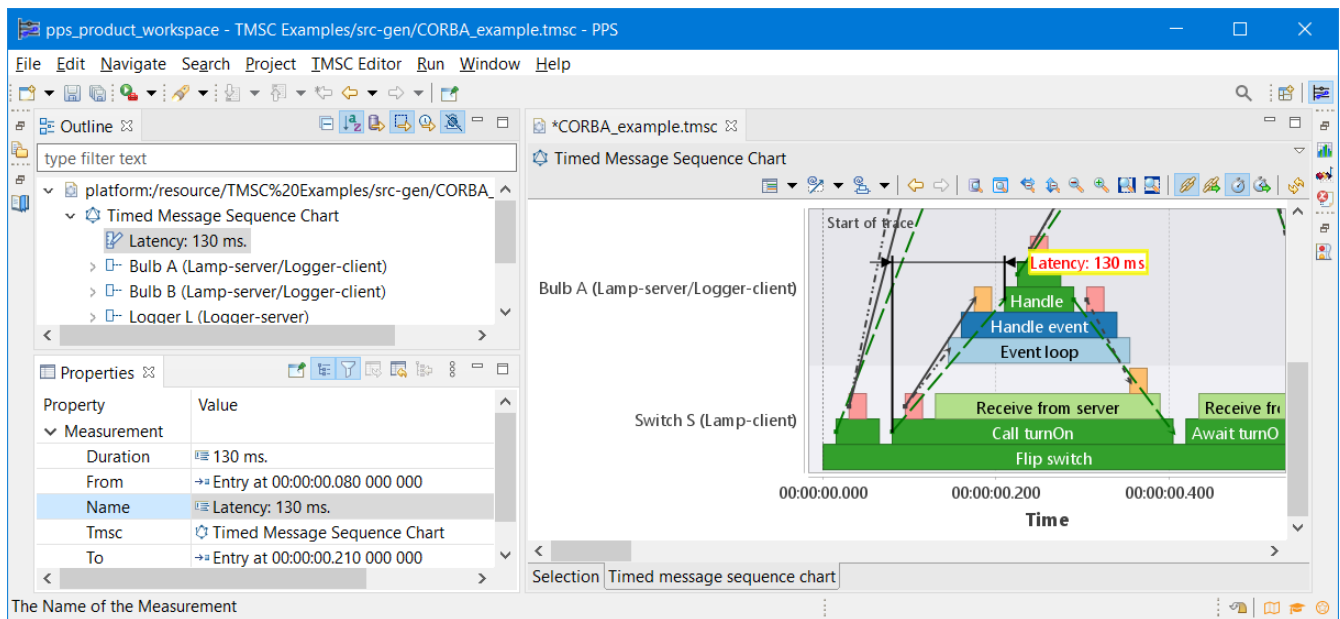
## 4.3. Measure Time

It is possible to measure the time between for instance the start of one task, and the termination of another task in a chart. While hovering over a task, a gray vertical support line is shown. While


hovering over the first half of the task, it is shown at the creation of the task. While hovering over the second half of the task, it is shown at the termination of the task. Hold down the **Alt** key while such a support line is shown, and left click it, to keep the support line fixed at that moment in time. Then hover to move the support line to another end point of a task and again left click while holding the **Alt** key. An annotation is then shown that visually and textually shows the time between the two support lines. The measurement is also added to the model and as shown  in the *Outline* view and the dirty (\*) state of the editor. The user can choose to save  the measurement or to undo the operation by using the **Edit > Undo** menu.



To change the text of the measurement annotation, please [select](#) the annotation and edit the *Name* in the *Properties View*.



## 4.4. Export chart to PNG file

It is possible to save a snapshot of a chart by clicking the **[Export As Image]**  from the view menu (small triangle button ▼ at the right top of the view). The action will open a save dialog which allows you to select the location and the name of the exported PNG file.



The PNG is saved with the same resolution and size as currently displayed.

## 4.5. Synchronize Timing

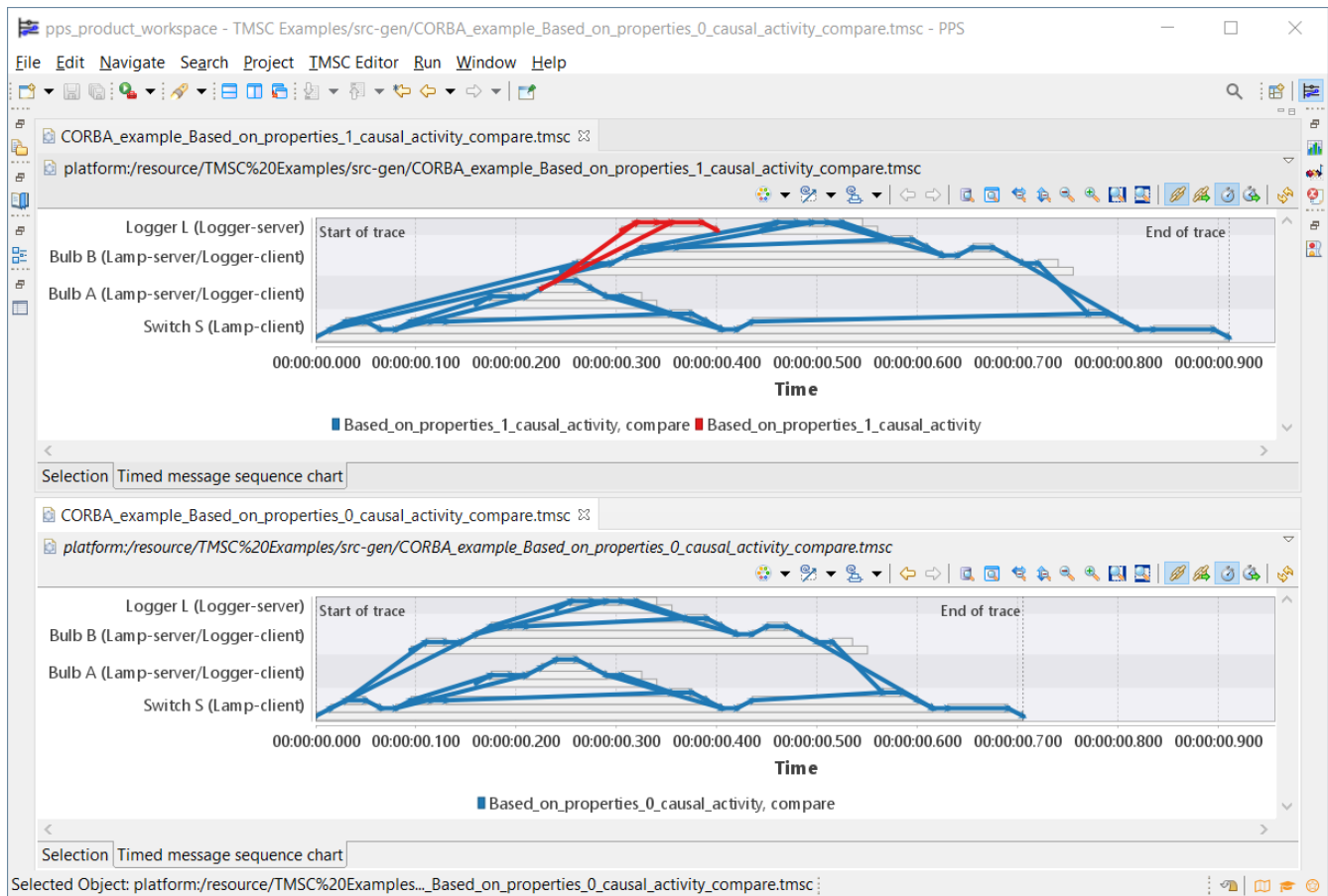
The time range in open viewers can be synchronized, so that multiple aspects on a time range as shown in the different viewers, can be analyzed simultaneously. When one editor is zoomed or scrolled, the other editors remain synchronized. This is especially helpful when editors are positioned above or 'on top of' each other. As in that case, the margins to the left and right of the time range will be synchronized as well, such that the time ranges have the same width in the viewers. The data for corresponding points in time are then vertically aligned, making it easy to correlate the data from the different viewers.



Time range synchronization only works if the ranges of two viewers overlap. If the






current visible range of a viewer is outside the range of another editor, time range synchronization behavior is unpredictable.




Only PPS viewers that have a time range (typically the x-axis) support time range and margin width (the width taken by the margin labels, margin ticks, etc) synchronization. If a viewer supports time range and margin width synchronization, some buttons will be present in the toolbar of the viewer.

Icon	Action	Description
	<b>[ Synchronize Time Range ]</b>	If pushed, the current time range of the editor/view is explicitly synchronized to all other supported editors/views that have time range synchronization enabled.
	<b>[ Enable Time Range Synchronization ]</b>	If checked, the editor/view will participate in time range synchronization. If not checked, changes to the time range of the editor/view will not be automatically synchronized to any other editors/views, and the editor/view will not be synchronized to time range changes of other editors/views.
	<b>[ Synchronize Margin Width ]</b>	If pushed, the current margin width of the editor/view is explicitly synchronized to all other supported editors/views that have margin width synchronization enabled.

Icon	Action	Description
	<b>[ Enable Margin Width Synchronization ]</b>	If checked, the editor/view will participate in margin width synchronization. If not checked, changes to the margin width of the editor/view will not be automatically synchronized to any other editors/views, and the editor/view will not be synchronized to margin width changes of other editors/views.

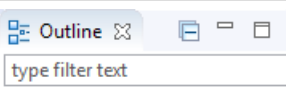

Currently, PPS viewers do not support manually changing the margin width. As such, the margin width is only synchronized by explicit request, using the **[ Synchronize Margin Width ]** toolbar button () . To reset the margin width of an editor/view, click the **[ Refresh ]** toolbar button () .



Disabling time range and margin width synchronization can be useful when dealing with large traces. Zooming in to a specific part of a trace while having time range synchronization enabled may lead to performance issues. To mitigate this, disable time range synchronization on a view, zoom in to the time range of interest within that view, re-enable time range synchronization for the view, and press the **[ Synchronize Time Range ]** toolbar button () to explicitly synchronize the new time range to the other views. That way, the other views are only updated once.

Time range and margin width synchronization are performed per workbench window. If you open a second workbench window, the editors/views within each workbench window are synchronized. Editors/views across different workbench windows are not synchronized. This is helpful when looking at different traces in separate workbench windows.

## 4.6. Dynamic Inclusion Filtering

Typically PPS works with large data sets. This complicates focussing on relevant data and the following functions therefore have been added to the *Outline* view.

Icon	Action	Description
	<b>[ type filter text ]</b>	Dynamic filtering of its elements, by means of entering a search pattern in its filter text box.  The pattern may contain '*' for 0 and many characters and '?' for exactly one character. Literal '*' and '?' characters must be escaped in the pattern e.g., '\*' means literal '*', etc. Escaping any other character (including the escape character itself), just results in that character in the pattern. e.g., '\a' means 'a' and '\\' means '\'
	<b>[ Push Context ]</b>	If checked and one or more elements are selected, the selected elements are set as input for the active editor. The editor will treat this input as its context and show the most relevant information accordingly.

Icon	Action	Description
	<b>[ Push Selection ]</b>	If checked and one or more elements are selected, the selected elements are also highlighted in the active editor.
	<b>[ Push Time Range ]</b>	If checked and one or more elements are selected, the time range to show these elements is calculated and is set to the active editor.

# Chapter 5. Release Notes

The release notes of the Platform Performance Suite (PPS) are listed below, in reverse chronological order.

## 5.1. PPS v0.20

This release prepares PPS to be open sourced.

The highlights of this release are:

- TMSc meta-model changes, see developer guide for more information:
  - Added **Host** concept to the deployed architecture life-cycle phase.
  - **Executions** are not **Dependencies** anymore.
  - Added **ITimeRange** interface to support time-based selections.
- This release also includes many bug fixes and improvements.

## 5.2. PPS v0.0.3

The highlights of this release are:

- Upgrade from Eclipse 2020-06 to Eclipse 2021-12, see [New and Noteworthy](#) for more information.
- Reuse common libraries from [Eclipse TRACE4CPS™](#) and [Eclipse LSAT™](#)
- PPS RCP now packages a JavaSE-11 JRE
- This release also includes many bug fixes and improvements.

## 5.3. PPS v0.0.2

This release of PPS adds the feature of defining, resolving and analyzing metric instances in a TMSc. When a metric instance exceeds its budget, a detailed root cause analysis can be performed to diagnose which task(s) caused this.

## 5.4. PPS v0.0.1

This first release of PPS includes a [TMSc graphical viewer](#) that is supported by a [textual syntax](#).