

## DevOps Model RealTime 12.1 Installation Instructions

---

## Table of Contents

1 Introduction.....	3
2 Installing from the Eclipse workbench.....	4
2.1 Install.....	4
2.2 License Setup.....	9
2.3 Update.....	10
2.4 Uninstall.....	10
3 Installing from the Command Line.....	12
3.1 Install.....	12
3.2 Uninstall.....	13
3.3 Extract identifiers for install categories.....	13
4 Installing using the Eclipse Installer.....	14
4.1 Basic usage of the Eclipse Installer.....	14
5 Shared Installations and Pre-Configured License Setup.....	18
5.1 Read-Only Installation.....	18
6 Post-Installation Setup.....	19
6.1 Modifications in eclipse.ini file.....	19
6.2 Configuring GTK Theme.....	19
7 Installing without Internet Access.....	20

---

# 1 Introduction

This document guides you through the installation of DevOps Model RealTime. It covers three different installation scenarios:

- An interactively performed local installation in [section 2](#)
- A command line based installation suitable for scripting in [section 3](#)
- An installation based on the Eclipse Installer in [section 4](#)

To follow the instructions in this document and successfully install Model RealTime 12.1 you will need the following:

- An installation of **Eclipse IDE for C/C++ Developers** for Eclipse 2024-06. This can be downloaded from <https://www.eclipse.org/downloads/packages/release/2024-06/r>. If you use the Eclipse installer it will be downloaded automatically.
- Access to a repository containing the Model RealTime update site. This can either be a locally downloaded zip file or the address of a shared repository that has been preconfigured for your usage.

You can download the Model RealTime update site zip file from [My HCLSoftware](#).

- A valid license. This can either be provided by means of a [license server](#) that contains Model RealTime licenses, or it can be a license key or file (permanent or trial) which you have obtained from HCL.
- An installation of a Java development kit (JDK) 64 bit.

**NOTE:** Model RealTime 12.1 requires a Java 64-bit Java 21 Runtime Environment (JRE) for running Eclipse and Model RealTime. (see the System Requirements for the exact recommended versions).

By default Eclipse 2024-06 distribution includes Open JDK version 21, but if you want to change it you could specify another JDK to use by editing the file `eclipse.ini` (found in the Eclipse installation folder). Specify the path to your JDK installation using the `-vm` argument. For example:

```
-vm  
D:/jdk-21/bin
```

## 2 Installing from the Eclipse workbench

Here we describe how to install Model RealTime as an update site on top of Eclipse 2024.06 (4.32) using the Eclipse graphical user interface. The sections that follow will cover how to install it from the command line or using the Eclipse installer.

**NOTE:** Some of the screen shots below were generated on the Windows platform for an older version of Model RealTime, and will therefore differ slightly when installing the latest version of Model RealTime 12.1 and when installing on Linux.

### 2.1 Install

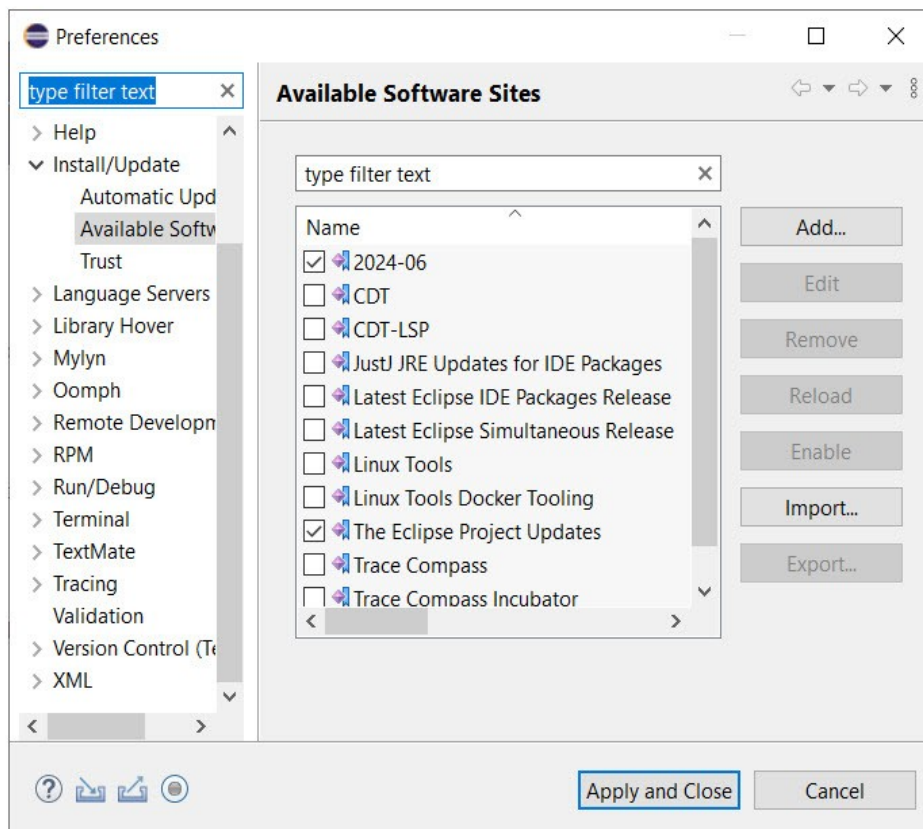
**NOTE:** You must have sufficient administration rights so that the OS will allow you to write the files to the installation directory of Eclipse for installing all features of Model RealTime.

If you don't have rights to write the files to the installation directory of Eclipse follow the instructions in the section on [5.1.Read-Only Installation](#).

**NOTE:** Make sure the Eclipse installation path does not contain any space characters. There are components in Model RealTime which do not work correctly if they are located in a path that contains a space.

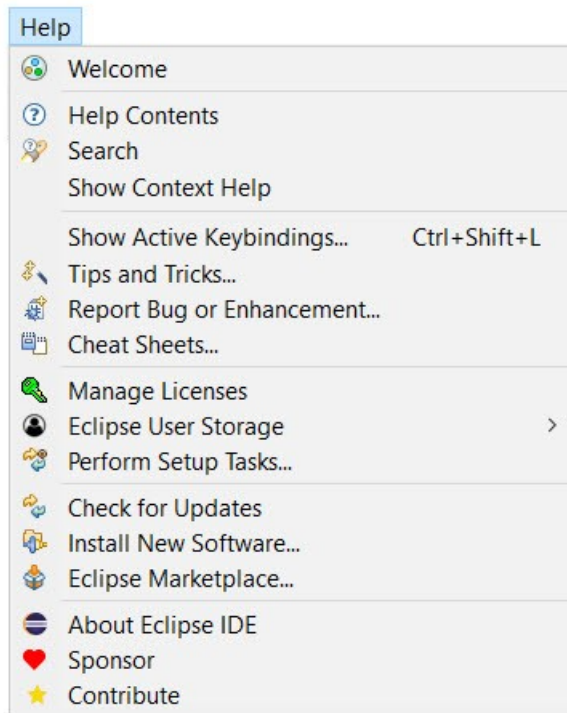
**NOTE:** Model RealTime can be installed on top of many Eclipse distributions, but we recommend to use "Eclipse IDE for C/C++ Developers". Installing it on other distributions may take a very long time since many dependent plugins need to be downloaded and installed.

1. Start Eclipse.
2. Open **Window - Preferences** and go to the **Install/Update - Available Software Sites** preference page. Make sure to uncheck all the software sites except the following: **2024-06, The Eclipse Project Updates**.

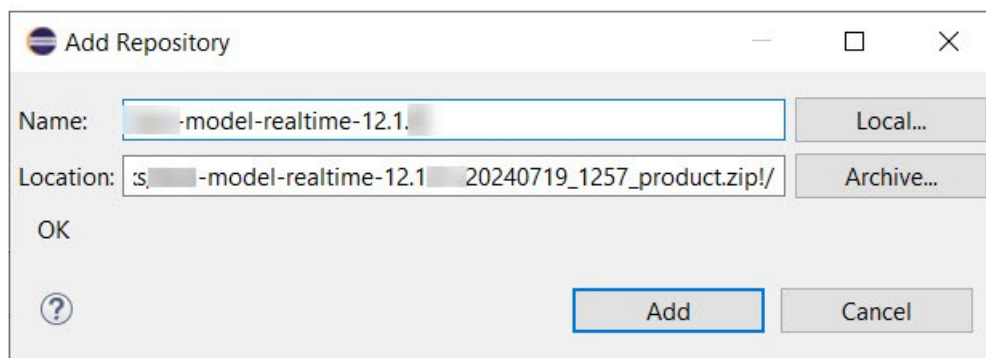


**NOTE:** It is important to uncheck items like **CDT**, **Latest Eclipse Simultaneous Release**, **Latest Eclipse IDE Packages Release** and others that can trigger Eclipse components upgrade, because such upgrades might cause errors during Model RealTime installation.

3. Choose the **Install New Software...** command from the **Help** menu.

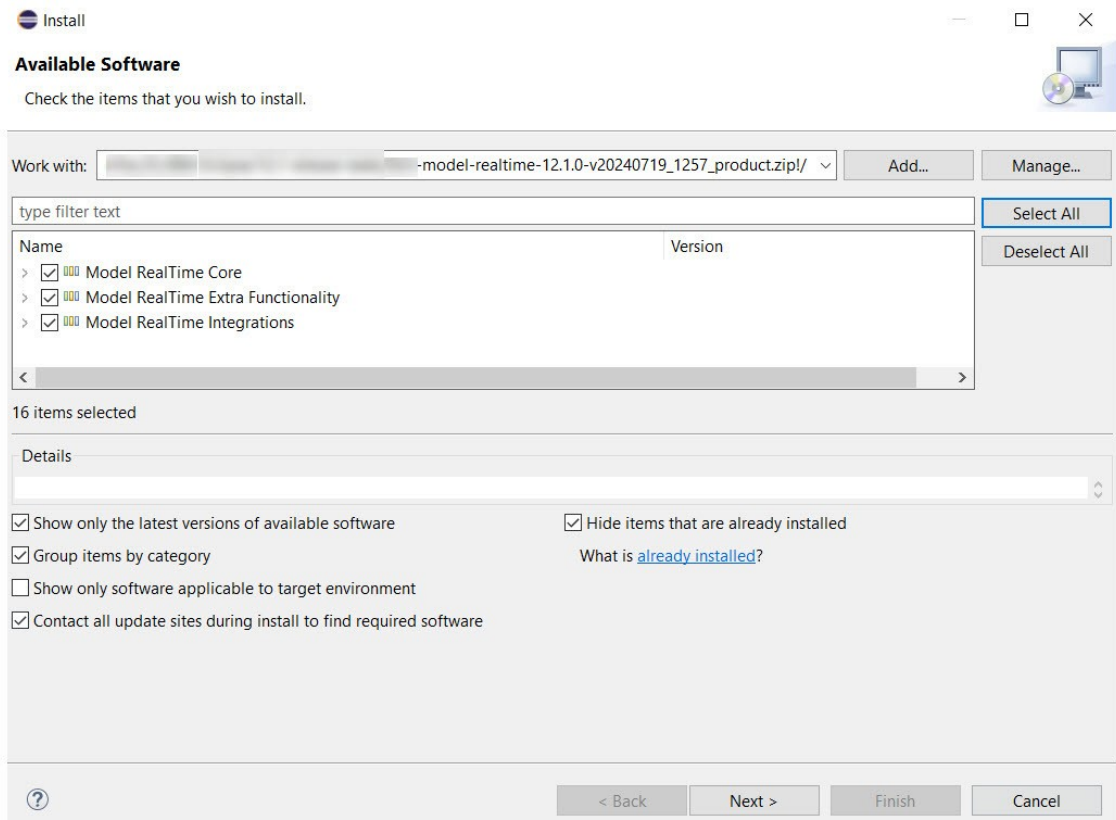


4. Press the **Add...** button and enter the location of the Model RealTime update site. If the update site is located on a server, you can directly type its URL in the **Location** field. If instead you have it as a ZIP file in the file system, press the **Archive...** button and select it. If the ZIP file already was unpacked to a folder, you should instead use the **Local...** button and select the folder. You can give a meaningful name to the repository in case you want to install from it again in the future. For example:



Note that the Model RealTime ZIP file that you should install from ends with "product". There is also another ZIP file in what you download that contains an offline version of the documentation.

5. Select the added repository in the **Work with** drop down list. A list of Model RealTime installation categories will be shown in the list below. It will look something like this (which categories that are available depends on the exact version of Model RealTime):



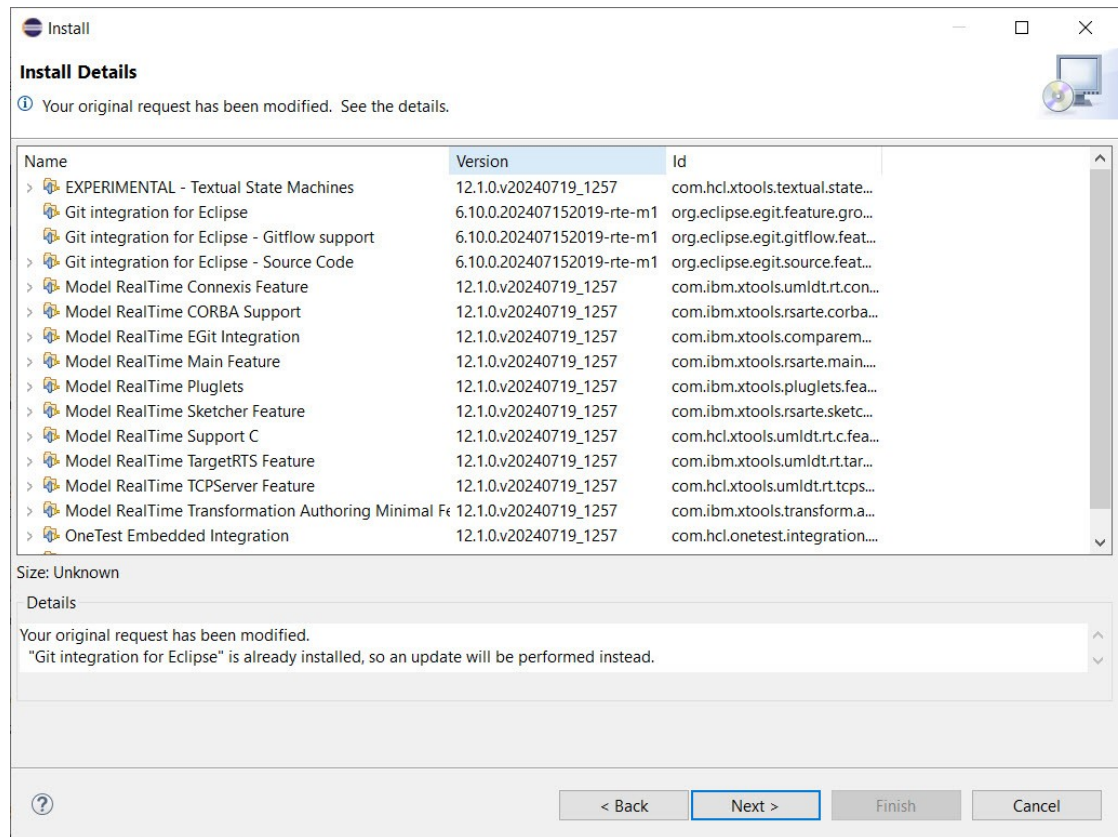
Under each installation category you can see the features it contains. You can select individual features to install, but often you will select only the top-level categories to install all features they contain.

The core Model RealTime functionality is contained in the **Model RealTime Core** category. All other categories contain optional additional functionality which you may or may not choose to install depending on your needs.

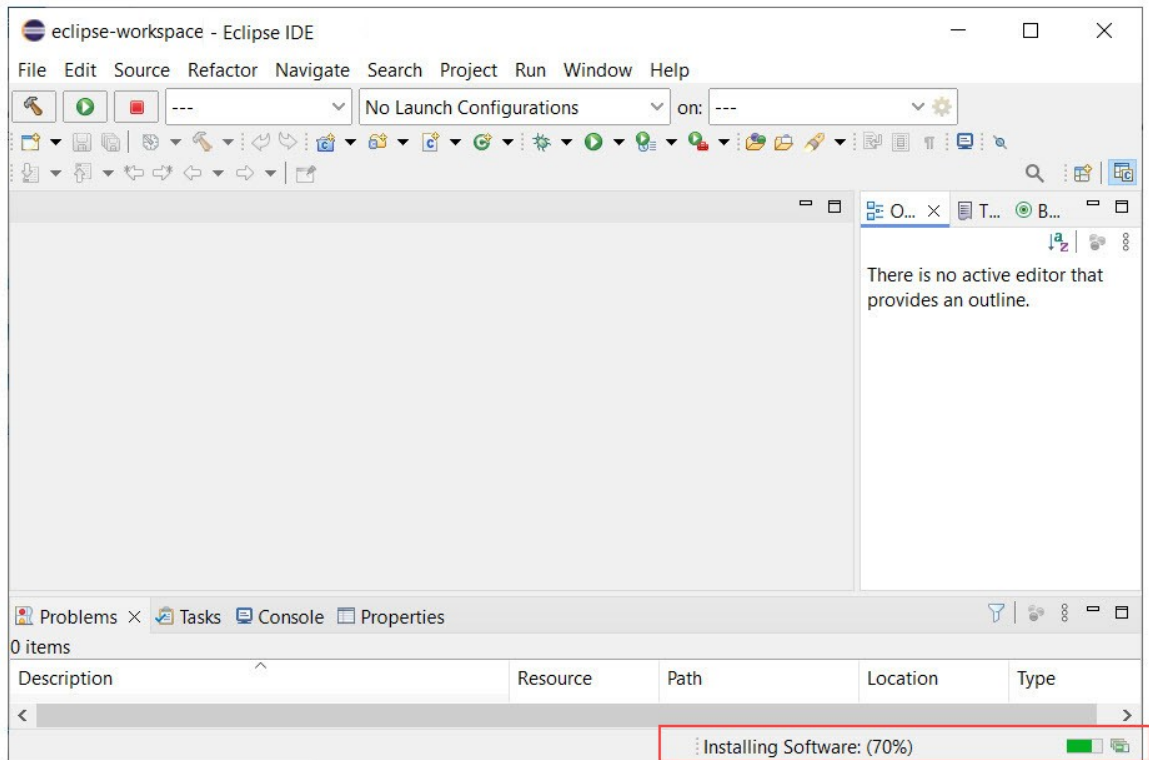
6. Select **Model RealTime Core** and any of the other optional categories you want to install. Press **Next**.

- **NOTE:** If you get errors at this step double check that you run with sufficient privileges to be able to make updates to the system and files. (On Windows you may need to start Eclipse using the “Run As Administrator” command and on Linux with root rights). Also make sure that you use a Java 21 JVM.
- **NOTE:** Depending on what Eclipse distribution you use, some Eclipse plugins which Model RealTime requires may be missing. Such plugins must be downloaded during the installation. To allow this to happen automatically you must select the checkbox **Contact all update sites during install to find required software**. As explained above, Eclipse 2024-06 (<https://www.eclipse.org/downloads/packages/release/2024-06/r>) must be in the list of available software sites (*Preferences – Install/Update – Available Software Sites*). If your machine does not have internet access, please follow the instructions given in the chapter [Installing without Internet Access](#).
- **NOTE:** If you install the Model RealTime EGit Integration (under the category Model RealTime Integrations) you also must install EGit (Git Integration for Eclipse). Model RealTime has its own version of EGit which is extended compared to the one that is part of Eclipse itself. At a minimum you must install the component called "Git integration for Eclipse".
- **NOTE:** If you choose to install the Model RealTime EGit Integration into an Eclipse installation that already contains a version of EGit that is different from the version included in the Model RealTime EGit Integration, you will be informed with a page similar to the one shown

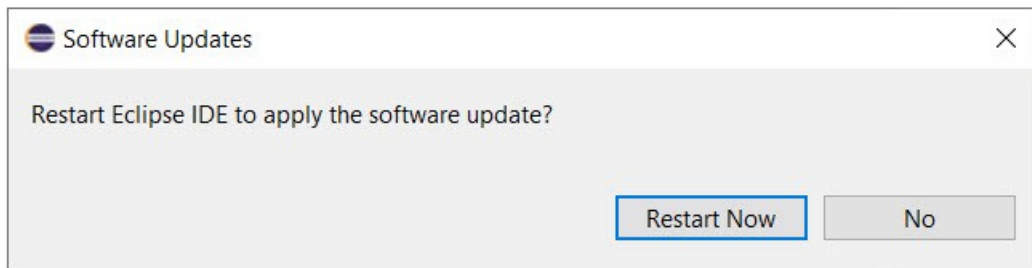
below that an update of EGit feature will be performed. This will automatically uninstall the existing EGit feature, before installing the Model RealTime EGit Integration.



7. If you already had an older version of Model RealTime 12.1 installed, you will get an **Install Details** page with a message that “Your original request has been modified”. This is just information that Eclipse detected an old version of Model RealTime which it will update. See chapter 2.3 for more information. Just press **Next** to proceed.
8. A final **Review Licenses** page will now be shown. If you accept the license agreements, click the appropriate button and then **Finish**.
9. The installation will now start. This can take several minutes so please be patient.



10. After the installation is complete, you will be asked to restart Eclipse. Answer **Yes**. You are not encouraged to try to use Model RealTime without first restarting Eclipse.



- **NOTE:** The first start-up of Model RealTime can take a long time, especially if you installed it on top of another Eclipse distribution than “Eclipse IDE for C/C++ Developers”. The reason is that all downloaded plugins must be reconciled with already existing Eclipse plugins, and this can take a significant amount of time. Subsequent start-up times will be much faster.

11. When Eclipse has started again you can confirm that Model RealTime was successfully installed. Open the **About** dialog and locate the Model RealTime icon:





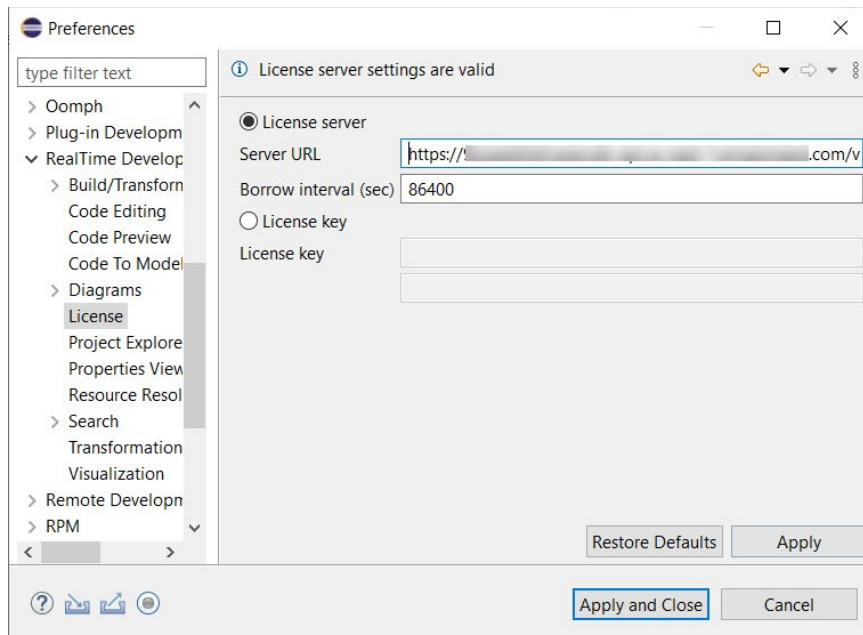
12. Click the button to view the installed version of Model RealTime and release.

You can also find the installed Model RealTime features if you press the **Installation Details** button in the About dialog, and then click the **Installed Software** tab.

## 2.2 License Setup

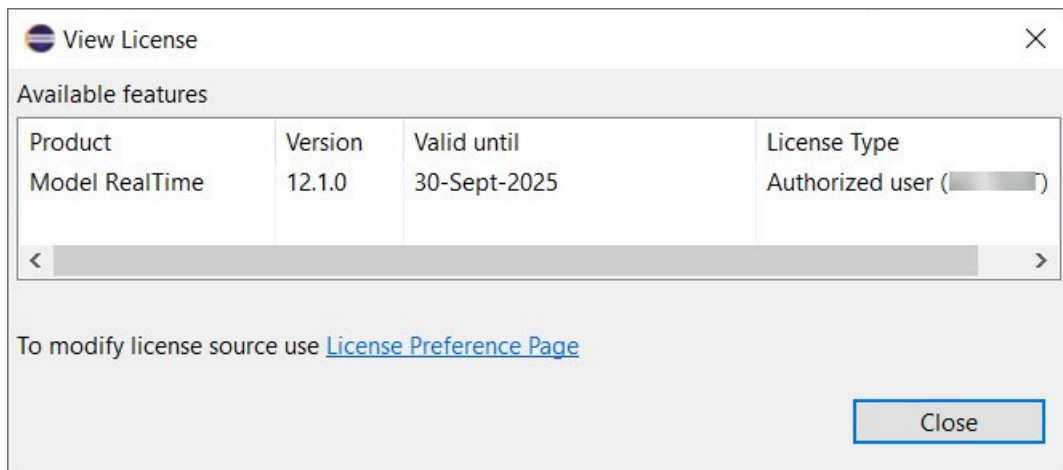
Model RealTime can start without having a license, but you will not be able to do much before you have applied a valid license.

Use the preference page **RealTime Development – License** to apply a license.



There are two ways to specify the license:

- **License server**  
This is the server URL to be used as the license server once you purchase the product. This could either be a URL of a locally set-up license server, or a cloud license server. Local license server URLs usually look like `https://myserver.com:<PORT>/v1/licensepools/XXXXXX/leases/`. An example of cloud license server URL is `https://cloudserver.com/v1/licensepools/XXXXXX/leases`.  
You can specify a borrow time for the license. It determines how long the license is kept by Model RealTime. By default, the borrow time is set to 24 hours (86400 seconds).
- **License key**  
If the license server option is unavailable or not functioning, as a backup plan, you can use a valid license key to activate the product and enable an authorized license.  
To view the license, you can open the **View License** dialog by means of the command **Help – View License...**



**NOTE:** To manage your Model RealTime licenses (and licenses for other HCL products) go to [my.hcltechsw.com](https://my.hcltechsw.com).

---

## 2.3 Update

If your Eclipse already contains a version of Model RealTime, you can update it to a later version. The steps you follow then are the same as described above. The only difference is that you will get a message telling you that some or all of the selected Model RealTime features are already installed, and that an update will be performed.

It is normally not necessary to uninstall the old version of Model RealTime before installing the new version.

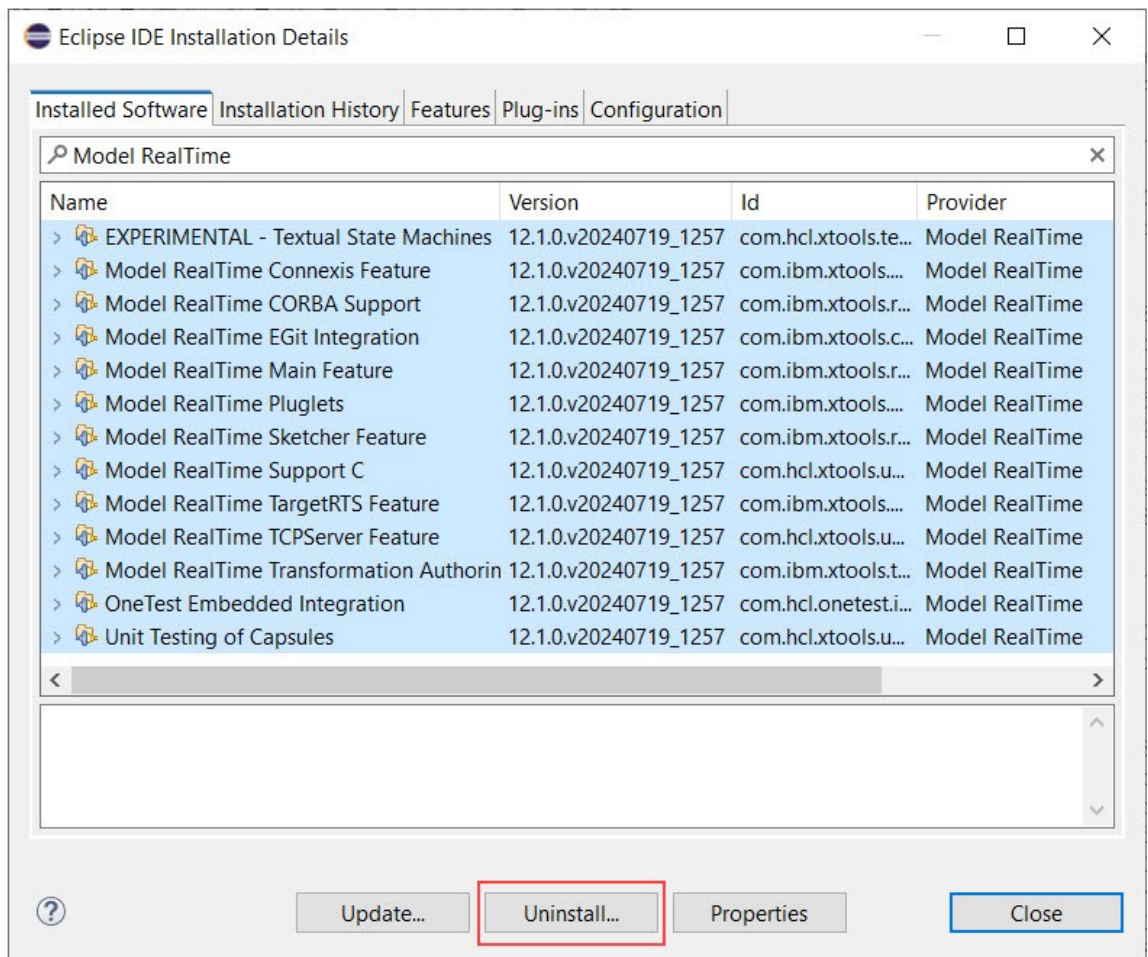
If you are installing from a repository that is being updated with new releases of Model RealTime you can also update your local installation using the Eclipse command **Check for Updates** in the **Help** menu.

---

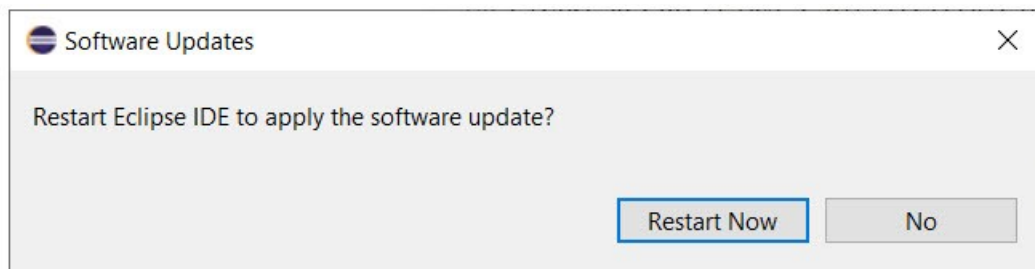
## 2.4 Uninstall

You can uninstall Model RealTime from the **About** dialog. Press the **Installation Details** button and then go to the **Installed Software** tab.

1. Select all lines starting with "Model RealTime".
  - **NOTE:** In case Model RealTime was installed from the command line (as described in the next chapter), you may see products grouped by several Model RealTime categories. In this case just select all top Model RealTime categories.
2. Press the **Uninstall...** button and then **Finish**.



3. After Model RealTime has been uninstalled, you will be asked to restart Eclipse. Accept it by pressing **Yes**.





```
<p2_director> -repository <model_rt_update_site> -installIU "v20240725_0727.model-  
realtime.core, v20240725_0727.model-realtime.extra, v20240725_0727.model-  
realtime.integrations, org.eclipse.egit.feature.group" -r  
http://download.eclipse.org/releases/2024-06
```

The category identifiers are given after the `-installIU` argument. See chapter 3.3 for how to extract the identifiers for available categories from the update site.

**NOTE:** The feature `org.eclipse.egit.feature.group` mentioned above refers to EGit. Normally a version of EGit is already present in Eclipse before you install Model RealTime, but if you choose to install the Model RealTime EGit Integration you should first uninstall the standard version of EGit using a command similar to:

```
<p2_director> -uninstallIU org.eclipse.egit.feature.group
```

---

## 3.2 Uninstall

**NOTE:** In Model RealTime version 12.1 **category identifiers begin with version qualifier**. An example of a command uninstalling all features for version `v20240725_0727` is:

```
<p2_director> -repository <model_rt_update_site> -uninstallIU "v20240725_0727.model-  
realtime.core, v20240725_0727.model-realtime.extra, v20240725_0727.model-realtime.in-  
tegrations"
```

where the category identifiers are given after the `-uninstallIU` argument. See chapter 3.3 for how to extract the identifiers for available categories from the update site.

---

## 3.3 Extract identifiers for install categories

The command for extracting the identifiers of available categories in the Model RealTime update site is:

### Windows:

```
<p2_director> -repository <model_rt_update_site> -list | findstr "model-realtime." |  
findstr /V "com.ibm.xttools." | findstr /V "com.hcl."
```

For example:

```
eclipse.exe -nosplash -application org.eclipse.equinox.p2.director -repository  
file:///E:/model-realtime-12.1-v20240719_1257_product -list | findstr "model-real-  
time." | findstr /V "com.ibm.xttools." | findstr /V "com.hcl."
```

### Linux:

```
<p2_director> -repository <model_rt_update_site> -list | grep "model-realtime." |  
grep -v "com.ibm.xttools." | grep -v "com.hcl."
```

The interesting lines in the result will look like the following (the qualifiers may be different):

```
v20240719_1257.model-realtime.core=1.0.0.885-cJJB2DBcMAMYAqH16GMeq  
v20240719_1257.model-realtime.extra=1.0.0._07E-cJIwpGED-z0V-TWCmky1-lC  
v20240719_1257.model-realtime.integrations=1.0.0.8J8--cJJub1A7QUoYyqr2wTFboOz-9iG
```

The category identifier is the name before the equal sign.

The update site also contains an Model RealTime specific version of EGit which has the same identifier as the standard version of EGit (`org.eclipse.egit.feature.group`).

---

## 4 Installing using the Eclipse Installer

The Eclipse Installer is a tool suitable for automating Eclipse-based installations. You can find more information about the Eclipse Installer at [https://wiki.eclipse.org/Eclipse\\_Installer](https://wiki.eclipse.org/Eclipse_Installer). There you also find instructions on how to download and install the Eclipse Installer.

**NOTE:** You must use version 1.4.0 or later of the Eclipse Installer. Installation errors will occur if you attempt to install Model RealTime with an older version of the installer.

---

### 4.1 Basic usage of the Eclipse Installer

Using the Eclipse Installer to install Model RealTime requires a setup file that specifies what should be installed. The minimal setup file for an Model RealTime 12.1 installation that installs all Model RealTime features, is the following:

```
<?xml version="1.0" encoding="UTF-8"?>
<setup:Product
  xmi:version="2.0"
  xmlns:xmi="http://www.omg.org/XMI"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:setup="http://www.eclipse.org/oomph/setup/1.0"
  xmlns:setup.p2="http://www.eclipse.org/oomph/setup/p2/1.0"
  name="Model RealTime"
  label="Model RealTime 12.1">
<version name="2024_06"
  label="2024_06"
  requiredJavaVersion="21.0.3">
<setupTask
  xsi:type="setup.p2:P2Task"
  label="Model RealTime including Eclipse 2024-06">
<requirement
  name="epp.package.cpp"
  versionRange="[4.32.0,4.33.0)"/>
<requirement
  name="org.eclipse.platform.feature.group"
  versionRange="[4.32.0,4.33.0)"/>
<requirement
  name="org.eclipse.rcp.feature.group"
  versionRange="[4.32.0,4.33.0)"/>
<requirement
  name="org.eclipse.sdk"
  versionRange="[4.32.0,4.33.0)"/>
<requirement name="org.eclipse.jgit.feature.group"/>
<requirement name="org.eclipse.egit.feature.group"/>
<requirement name="org.eclipse.jgit.pgm.feature.group"/>
<requirement name="_VERSION_.model-realtime.core"/>
<requirement name="_VERSION_.model-realtime.extra"/>
<requirement name="_VERSION_.model-realtime.integrations"/>
<repository
  url="https://download.eclipse.org/releases/2024-06"/>
<repository
  url="url_to_modelrealtime_repository"/>
</setupTask>
</version>
<description>Model RealTime Installation Script</description>
</setup:Product>
```

You can copy the text above and do the following replacements.

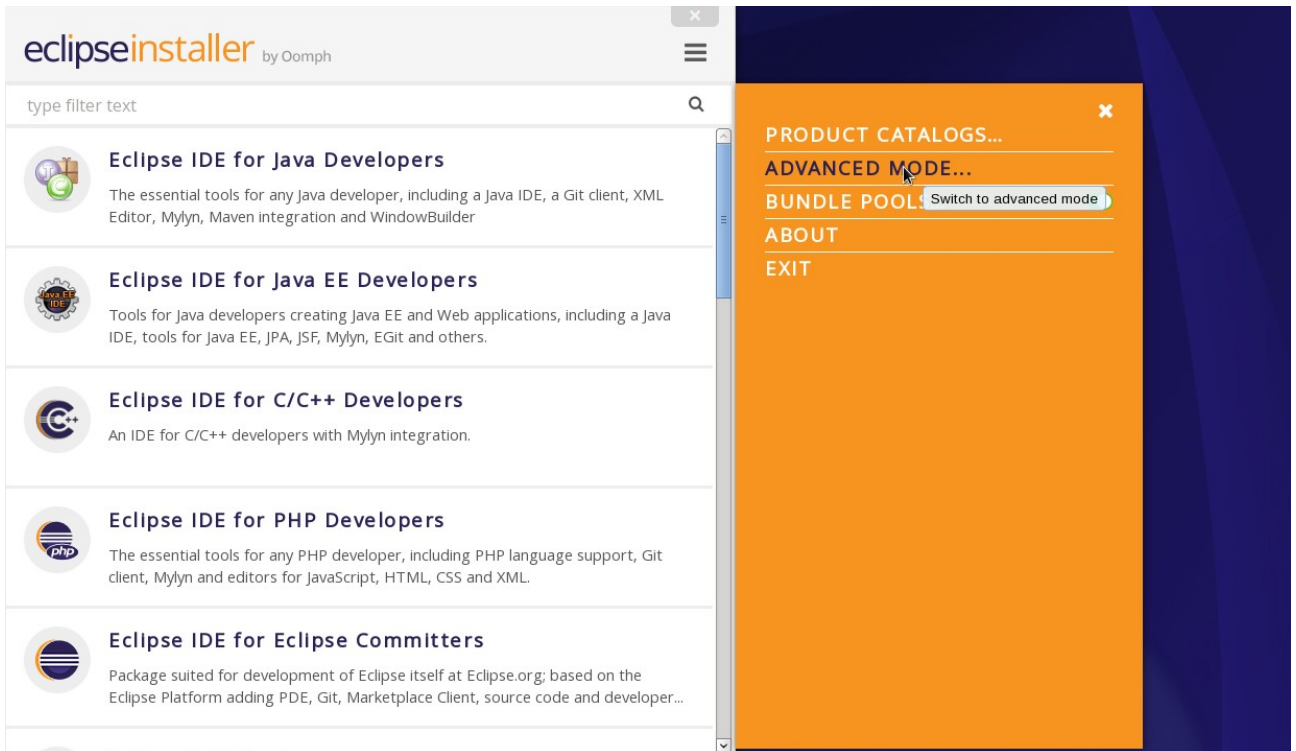
1. Replace ***url\_to\_modelrealtime\_repository*** with your local Model RealTime 12.1 repository location and use a file with this contents as a starting point for your Eclipse installer setup script. Note that copying from PDF format may change the hyphens in the text (“-”) and introduce “soft hyphens”. So check that the “-” characters still remain after copying.

Here is an example of what the repository URL for a local Model RealTime 12.1 repository (unzipped into a folder) can look like in the setup script:

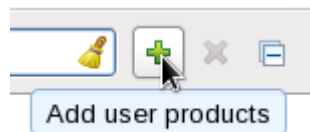
```
<repository url="file:///C:/hcl-model-realtime-12.1-v20240725_0727_product"/>
```

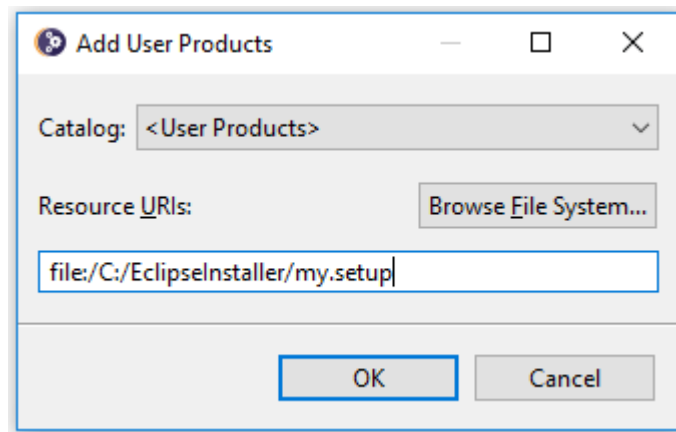
2. Replace ***VERSION*** with the actual version qualifier of the product you will be installing, for example v20240725\_0727. See Section [3.3 on how to extract the identifiers from the zip file](#).

To install Model RealTime, start the Eclipse installer and switch to Advanced Mode.



Use the command **Add user products** to choose your setup file.





You can now install Model RealTime from the main Eclipse Installer page by selecting **Model RealTime 12.1** and pressing **Next**.

In the subsequent dialogs the default values work fine, even though you may want to modify the installation folder name. If you add additional Eclipse projects to your installation, remember to confirm that the target platform is set to 2024-06.

**NOTE:** Make sure the Eclipse installation path does not contain any space characters. There are components in Model RealTime which do not work correctly if they are located in a path that contains a space.

After the execution gets completed successfully, you can see the following dialog:



**Progress**

Task execution has successfully completed. Press Back to choose different settings or Finish to exit.



> P2 Director (Model RealTime including Eclipse 2024-06 + JRE 21.0.1)

- Eclipse Ini --launcher.appendVmargs
- Eclipse Ini -Declipse.p2.max.threads=10
- Eclipse Ini -Doomph.update.url=https://download.eclipse.org/oomph/updates/milestone/latest
- Eclipse Ini -Doomph.redirection.index.redirection=index;->https://raw.githubusercontent.com/eclipse-oomph/oomph/m...
- Eclipse Ini --add-opens=java.base/java.lang=ALL-UNNAMED
- Eclipse Ini -Djava.security.manager=allow**

---

< >

```
Changing C:\Users\...\.PROD\Model RealTime-ex\eclipse\eclipse.ini (-Doomph.update.url:
Performing Eclipse Ini -Doomph.redirection.index.redirection=index:/->https://raw.githubusercontent.com/eclipse-oomph/oomph/m...
Changing C:\Users\...\.PROD\Model RealTime-ex\eclipse\eclipse.ini (-Doomph.redirection
Performing Eclipse Ini --add-opens=java.base/java.lang=ALL-UNNAMED
Changing C:\Users\...\.PROD\Model RealTime-ex\eclipse\eclipse.ini (--add-opens=java.b
Performing Eclipse Ini -Djava.security.manager=allow
Performing post bootstrap tasks
Launching the installed product...
Took 115 seconds.
Press Finish to close the dialog.
```

---

Scroll lock  Dismiss automatically  Launch automatically

---

? 🏠 🔑 🏆 📄 🌐 🔄 1.34.0 Build 693 < Back Next > **Finish** Cancel

---

## 5 Shared Installations and Pre-Configured License Setup

If you would like to share one installation among a group of users, e.g. using a network installation, some special considerations concerning the license setup are needed. The main issue with this scenario is how to apply a valid license at the time of installation, to avoid that each user has to do this when they start Model RealTime.

The license information for Model RealTime is edited in the preference page *RealTime Development – License*. Contrary to most other preferences, this information is not stored in the workspace metadata but in the Model RealTime installation (more precisely in the file `<install-dir>/configuration/.settings/com.hcl.rtist.license.core.prefs`). This means that the person who creates a shared installation can enter the license information right after the installation is completed. All users of that installation will then have the license information already pre-configured and can immediately start to use Model RealTime.

---

### 5.1 Read-Only Installation

In a shared installation scenario it is common to have the installation folder to be read-only for regular users. In general this works for an Model RealTime installation but there are two constraints:

- The license information must be set-up before making the installation read-only, since it is stored within the installation folder.
- Features that provide libraries, such as the TargetRTS, Connexis and TCPServer features, cannot be installed into their usual locations in the `rsa_rt` folder if the Eclipse installation is read-only. However, the libraries are also installed into their corresponding plugins which in the case of a read-only Eclipse installation will be placed in the user configuration area. Another option could be to install these features on a different machine where the Eclipse installation is writable, and let all users access the libraries from there.

---

## 6 Post-Installation Setup

After installation you should do some post-installation setup tasks before starting to use Model RealTime. This is needed because the standard Eclipse settings with regards to memory etc. are usually not sufficient when working with Model RealTime.

---

### 6.1 Modifications in eclipse.ini file

Please refer to [this article](#) for information about setup tasks that are commonly needed for performance and other reasons.

**NOTE:** Starting from version [4.19 of Eclipse](#), the Standard Widget Toolkit (SWT) will report the following error for every non-disposed widget.

```
java.lang.Error: SWT Resource was not properly disposed
```

The errors will be visible in the Error Log view and workspace log file. To disable reporting of such errors a user should locate the following line in eclipse.ini file

```
-Dorg.eclipse.swt.graphics.Resource.reportNonDisposed=true
```

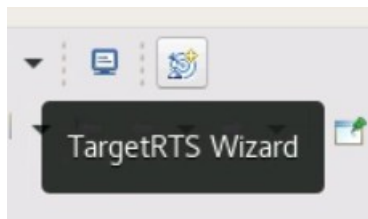
and replace true with false, so it looks like this:

```
-Dorg.eclipse.swt.graphics.Resource.reportNonDisposed=false
```

---

### 6.2 Configuring GTK Theme

On Linux some GTK themes (for example Adwaita on RHEL8) can set tooltip backgrounds to be dark like in the image below.



This should be avoided, because it can cause issues with tooltips Model RealTime will be showing in diagrams. To address this a user can either look for a theme with a light tooltips background or override the theme's settings by creating a file `~/.config/gtk-3.0/gtk.css` with the following contents:

```
tooltip { text-shadow: none; }
tooltip.background { background-color: rgba(255, 255, 255, 1.0);
border: 1px solid rgba(0, 0, 0, 1.0); }
tooltip decoration { background-color: transparent; }
tooltip * { background-color: transparent; color: black; }
```

After saving the file Model RealTime should be restarted for the changes to become effective.

---

## 7 Installing without Internet Access

As explained in the chapter [Install](#) it may be necessary during the installation of Model RealTime to download additional Eclipse plugins, that are needed by Model RealTime but are missing in your Eclipse installation. Usually this happens automatically (provided you have marked the option **Contact all update sites during install to find required software**). However, if you don't have access to the internet this obviously won't work. For example, this can happen if your machine is behind a firewall which blocks access to some internet sites such as eclipse.org.

If you get errors like "missing required plug-in" it's likely that Eclipse failed to download the necessary Eclipse plugins. If you cannot temporarily enable the internet access on the installation machine, you can workaround the problem in the following way: install Model RealTime on a different machine which has access to internet and then copy the installation folder to the target machine.