OBEO
7 boulevard Ampère
44470 Carquefou
France
support@obeo.fr

Team for Capella

INSTALLATION GUIDE

This document is the property of THALES.

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

CONTENTS

1 Scope....6
1.1 Typographic and notation rules....6
1.2 Identification....6
1.2.1 Description of the product....6
1.2.2 Deployment modes....9
1.2.3 Applicability....9
1.3 Document overview....9
2 Referenced documentation....10
3 Team for Capella Server installation....11
3.1 Installation Architecture....11
3.2 Recommended system requirements....12
3.2.1 Server computer recommended system requirements....12
3.2.2 Client computer recommended system requirements (only for local client installation) 13
3.3 Deployment recommendations....14
3.3.1 Network....14
3.3.1.1 Latency: Client and Team Server....14
3.3.1.2 Latency: Team server and DB server....14
3.3.1.3 Network stability....14
3.3.1.4 Server isolation....14
3.3.2 Scalability and size of models....14
3.3.3 Disclaimer....15
3.4 Installation procedure....15
3.4.1 Team for Capella Server installation procedure....15
3.4.1.1 Installation....15
3.4.1.2 Extensions installation....16
3.4.1.3 Jenkins scheduler installation....16
3.4.1.4 Administration features installation....17
3.4.2 License server installation....17
3.4.2.1 Server installation....17

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

3.4.2.2 Client configuration....17
3.4.3 Installation verification....18
3.5 Team for Capella server configuration....19
3.5.1 How to Reuse previous server configuration when configuring v6.1.x server....19
3.5.1.1 Procedure....19
3.5.1.2 Server CDO....19
3.5.1.3 Scheduler....20
3.5.1.4 Tools....20
3.5.1.5 Client (in remote clients installation)....20
3.5.1.6 License server....21
3.5.1.7 User Accounts Management....21
3.6 Uninstallation procedure....21
4 Team for Capella Client installation....23
4.1 Requirements....23
4.1.1 Recommended system requirements....23
4.2 Installation Procedure....23
4.2.1 Team for Capella Client installation procedure....23
4.2.1.1 Default client installation....23
4.2.1.2 Manual client client installation....24
4.2.2 Team for Capella Client configuration....25
4.2.3 Verification installation procedure....25
5 Release Notes....28
6 Migration of existing projects....29
6.1 Version compatibility....29
6.2 Model migration from previous version to v6.1....30
6.3 Model migration from an older version....30
7 Migraton from V6.1.0 to V6.1.0_202509....31
8 Performance consideration feedback....32
8.1 Reference time....32
8.1.1 Case Studies....32
8.1.2 Measures....33
8.1.2.1 Computer used for the tests....33
8.1.2.2 Reference elapsed time....33
8.1.3 Analysis and conclusion....34

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

8.2 Dedicated tests on *Combined IFEs*.....35
8.2.1 Server.....35
8.2.2 Client.....35
8.2.3 conclusion.....35
8.3 How to get characteristics of your model.....36

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

LIST OF TABLES

Table 1 - Reference Documents....10
Table 2 - Version Compatibility (sorted by delivery dates)....29
Table 3 - Reference models....33
Table 4 - Measured time on test models (5.0.0)....34

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

1 SCOPE

1.1 TYPOGRAPHIC AND NOTATION RULES

To improve legibility, some text elements are identified by specific typographic rules in according to their tutorial purpose.

- **Emphasis** font is applied to emphasize words which designs controls (Click on Cancel),
- `Fixed` font is applied to expressions and texts of ASCII (`C:\My documents\Path`),
- *Terminology* font is applied on expressions referenced in Terminology table.

	Points out- information useful for the user. Clarifies a detail.		Indicates warning information.
	Indicates a potential pitfall or operating risks.		Gives information in answer to expected user's question.
	Stop and read before going on.		Refers to an external document.

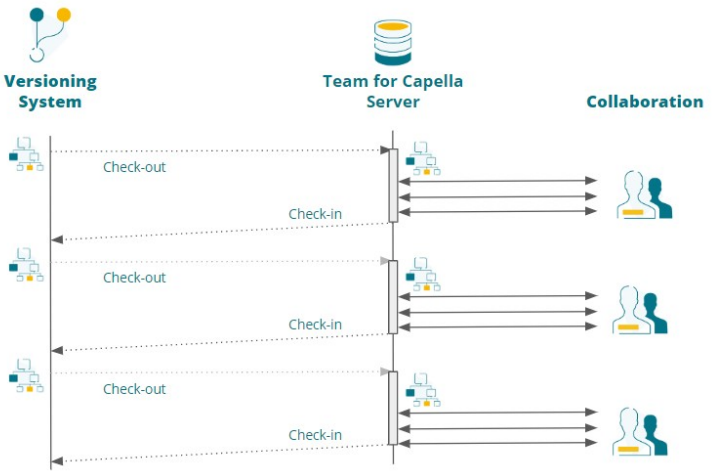
1.2 IDENTIFICATION

1.2.1 DESCRIPTION OF THE PRODUCT

Team for Capella is a collaborative solution to have several contributors working on the same model, with the granularity as fine as a model element and diagrams.

It decouples the versioning issue which is ensured by a Source Control Management (SCM) tool from the concurrent accesses issue.

It introduces a shared repository which is populated from the SCM tool and which enables several users to work on the same model.

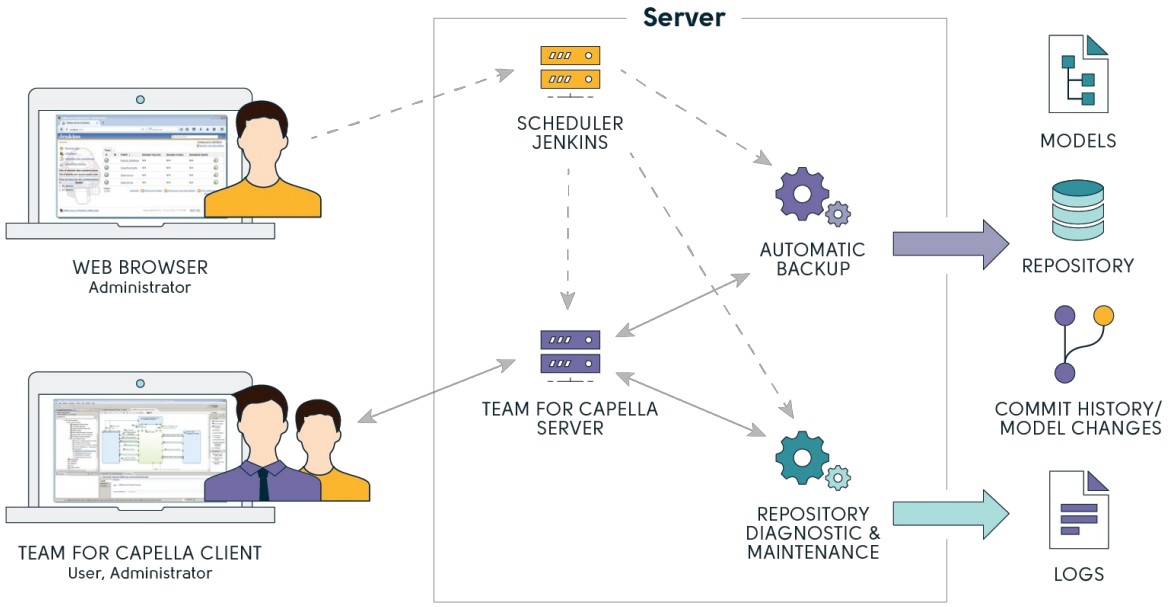


This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

Users can simultaneously edit the same Capella model without conflicts. Only modified elements are locked, not the whole model, and other users can see in live modifications made by connected teammates. It is no longer necessary to split the model into fragments or to merge concurrent modifications.

It is recommended to version exported models with a SCM tool, for instance Git. Please refer to the Capella embedded documentation at the section **Capella Guide > User Manual > Version Control with GIT** for more details.

Here is an overview of the Team for Capella architecture:



Team for Capella is composed of three parts:

- A server, to manage the model repositories, and associated features (such as locks, etc.);
- An administration module, to schedule automatic backups (models, changes and database) and trigger diagnostic and maintenance tasks;
- An add-on, packaged as an update site, to bring the multi-user functionalities on top of the standard Capella rich client.

Please refer to the Team for Capella documentation at the section **Team for Capella Guide > Rationale and Concepts** for more information.

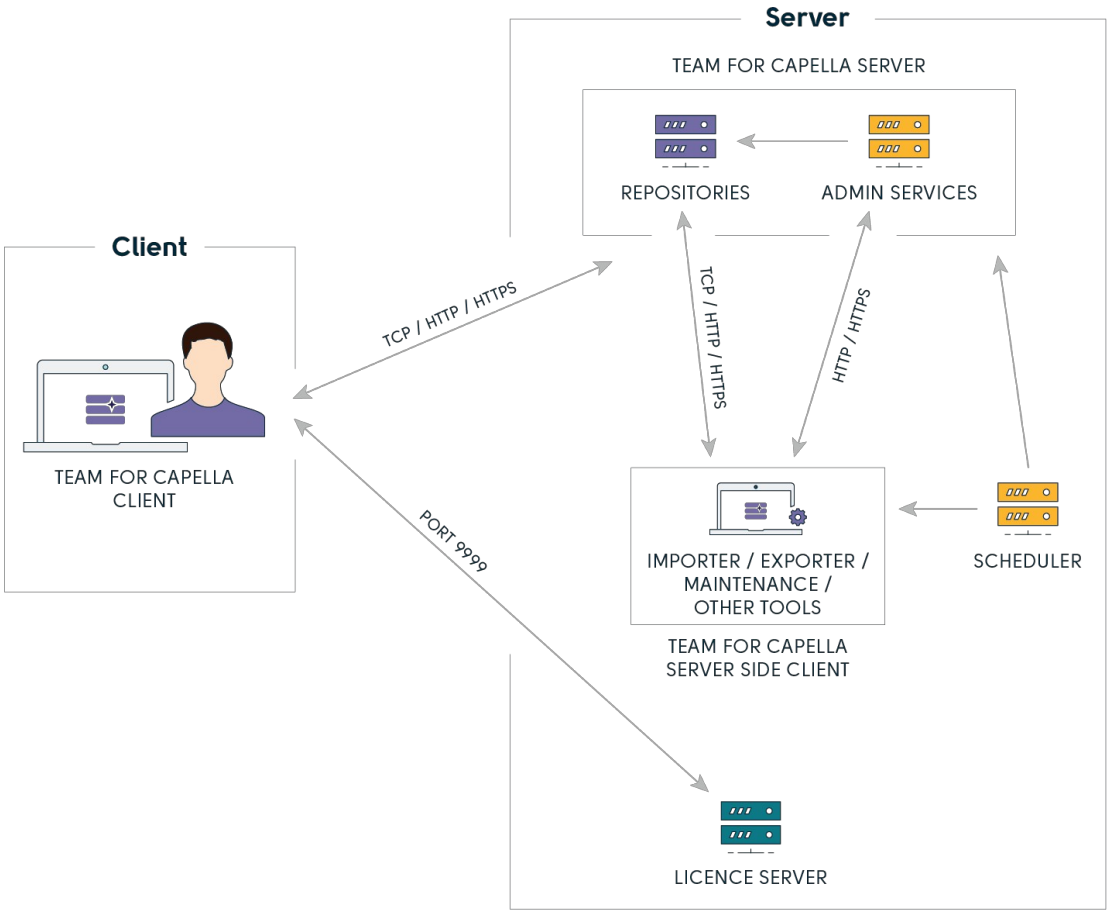
Team for Capella is available in 64 bits version for Windows and for Linux.

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

The Team For Capella server is composed of the CDO repositories server and an HTTP Jetty server. By default, the Jetty admin server is automatically started with the CDO server on the port 8080.

The REST admin server is used:

- to manage repositories with the REST Admin API
- by applications (importer, maintenance application, console application) to execute code on server



Please refer to the Team for Capella documentation at the section **Team for Capella Guide > System Administrator guide > Server Configuration** for more information.

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

1.2.2 DEPLOYMENT MODES

Team for Capella can be deployed on different modes:

- **Cloud:** With the Cloud deployment mode, Team for Capella server is installed and administrated by Obeo, and Team for Capella clients are accessible through a remote desktop technology.
- **On-Premise**: With the On-Premise deployment mode, Team for Capella is installed and administrated on the client's infrastructure. The multiple ways to install/deploy Team for Capella are described in section $3.1 - Installation Architecture.

1.2.3 APPLICABILITY

This guideline is applicable to following versions of Capella and Team for Capella:

Value name	Value
AppChk	[INSTALLDIR]\capella\capella.exe;
Product Name	Capella x64 / Team for Capella x64
Product Version	Capella 6.1.0 / Team for Capella 6.1.0_202509

This can be followed and adapted to the Linux version of Capella / Team for Capella.
Path are equivalents, the Linux bundle provides .sh scripts which are equivalent to the Windows .bat scripts described in this guide.

1.3 DOCUMENT OVERVIEW

This document is intended for persons in charge of installing Team for Capella.

It describes the nominal installation, configuration and uninstallation of Team for Capella.

Please refer to the Team for Capella documentation for more details and advanced configuration. The **Team for Capella Guide** is available from the documentation embedded in the client, but also online, see Referenced Documentation section.

To read the embedded documentation, launch a Team for Capella client (after installation), click on the *Help* menu in the top menu bar. Then click on the *Help Contents* button. A web page will appear with all documentation and especially the **Capella Guide** and **Team for Capella Guide** sections.

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

2 REFERENCED DOCUMENTATION

Title	Version
Team for Capella Guide: • Embedded documentation within Team for Capella • HTML extraction of Team for Capella embedded documentation • PDF extraction of Team for Capella embedded documentation	6.1.0_202509
REST API documentation of the experimental Administration Server	6.1.0_202509
Capella Online Installation Guide	6.1.0
Capella Release Notes	6.1.0
Team for Capella Release Notes	6.1.0

Table 1 - Reference Documents

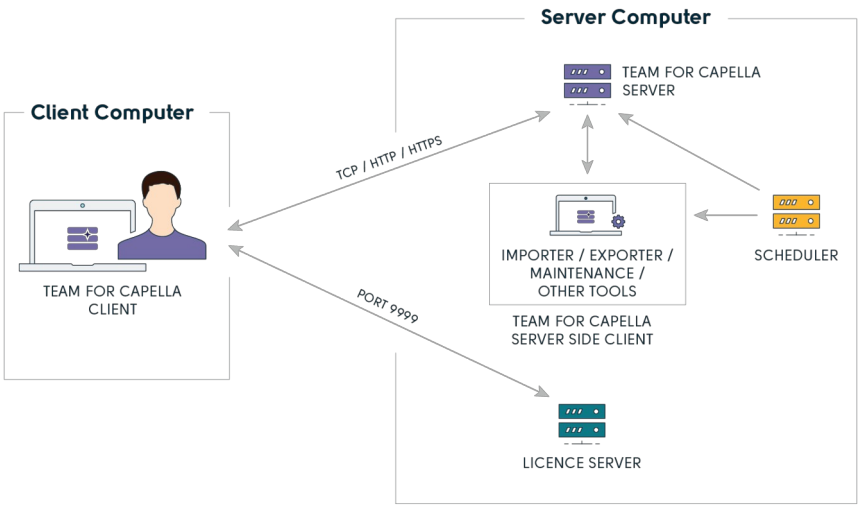
This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

3 TEAM FOR CAPELLA SERVER INSTALLATION

3.1 INSTALLATION ARCHITECTURE

There are two main ways to install Team for Capella:

- **Local clients:**
  - 1 server computer runs the Team for Capella Server and Scheduler,
  - n client computers run the Team for Capella Clients,

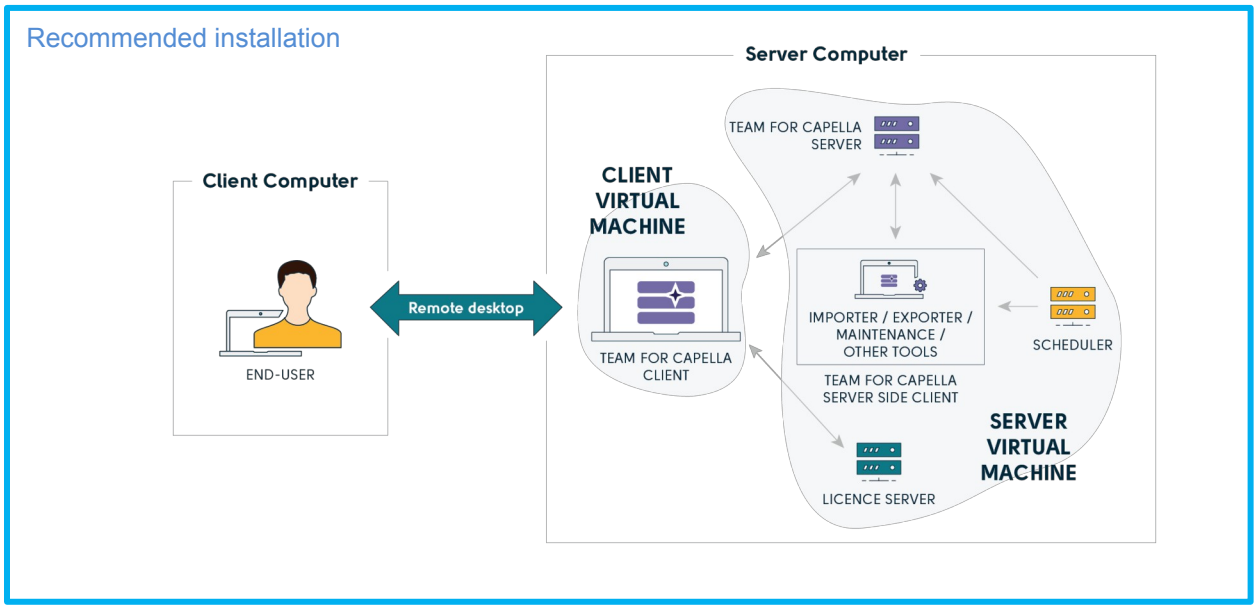


- **Remote clients:**

  The same computer runs the Team for Capella Server and Scheduler and n Team for Capella Clients (users connect to this computer using RDP). This allows to make the exchanges between the clients and the server independent of the network's bandwidth.

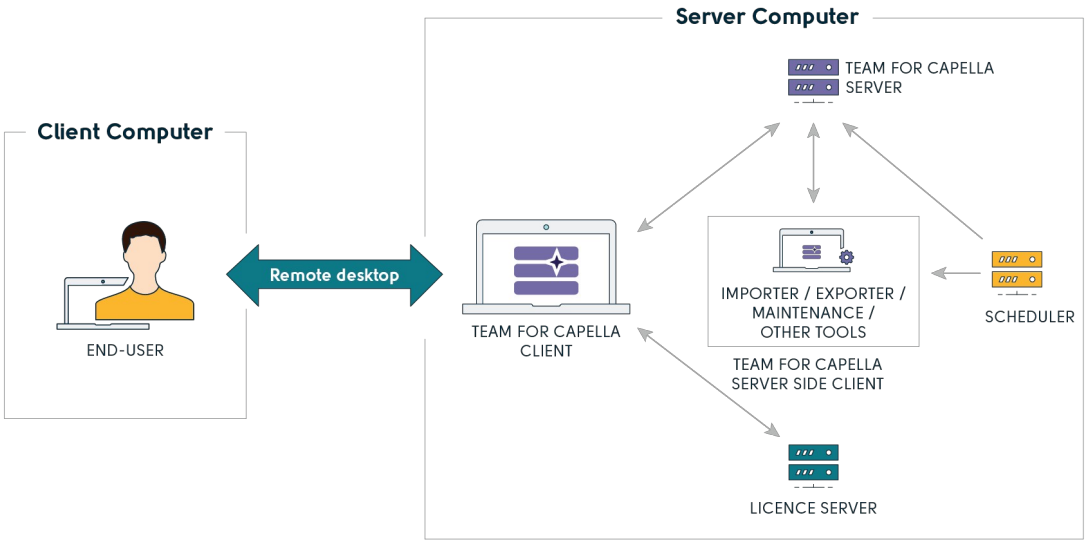
  In this configuration, the client and server can either be installed on the server computer or separated in two virtual machines. The latter is the recommended installation:

  - Client and server separated in two virtual machines:



This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

- Client and server on the same server machine:



3.2 RECOMMENDED SYSTEM REQUIREMENTS

3.2.1 SERVER COMPUTER RECOMMENDED SYSTEM REQUIREMENTS

If you installed a tool to act as a scheduler, it is mandatory to run it on the same computer as the Team for Capella Server.

For successful installation of Team for Capella Server, your computer must meet the following requirements:

- Local client installation (server side only):
  - 2 GHz processor,
  - RAM: 4 GB for the Team for Capella server + 3GB for the importer client
  - 15 GB of available hard disk space

- Remote client installation:
  - Multicore processor (2GHz)
    - 2 cores for Team for Capella Server, scheduler and license server
    - 1 core per running Team for Capella Client,
  - RAM:
    - 4 GB for the Team for Capella server + 3GB for the importer client
    - 3 GB per Team for Capella Client,
  - 15 GB of available hard disk space + 2 GB per Team for Capella Client,
  - 2 hard drives are recommended:
    - The first containing system files and software installation files (a SSD hard drive is mandatory if more than 8 users),
    - The second containing the Team for Capella Server file.

- System requirements:
  - Microsoft Windows 10/11 64 bits
    - no known issues with Windows 7/8
  - Microsoft Windows Server 2019/2022 64 bits

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

        - no known issues Windows Server 2008/2012/2016
    - Compatibility with Open JDK 17, see Capella Online Installation Guide
        - Capella and Team for Capella are configured to use the JRE provided by Capella (Eclipse Adoptium Open JDK 17.0.6, JDK)
    - Jenkins LTS 2.375.3 installed as a service on the server computer.

- Team for Capella database must be stored on a local hard drive,
- Security policies:
    - Virus scanner:
        - Team for Capella Server database files must not be scanned (`*.db`).
        - In addition, it should not be activated either on Capella models files: `*.aird`, `*.capella`, `*.airdfragment`, `*.capellafragment` and `*.afm`,
        - The license server hosts a collection of licenses stored in several encrypted `.ols` files. Those licenses pools must not be scanned.
        - Periodic analyses should not be launched when users are working (launch them at night),
    - Firewall:
        - At least 2 ports must be opened: the Team for Capella Server port (by default 2036) and the license server port (by default 9999),
        - In addition, the Scheduler port (by default 8036), the license server monitoring port (8086) and the Admin server port (8080)

- The computer should be fully dedicated to Team for Capella.

3.2.2 Client computer recommended system requirements (only for local client installation)

For successful installation of Team for Capella Client, your computer must meet the following requirements:

- 2 GHz processor
- 3 GB for Team for Capella client
- Microsoft Windows 10/11 64 bits
    - no known issues with Windows 7/8
- Compatibility with Java Runtime Environment 17 (Eclipse Adoptium Open JDK 17.0.6 is provided by Capella)
- Security policies:
    - Virus scanner:
        - It should not be activated on Capella models files: `*.aird`, `*.capella`, `*.airdfragment`, `*.capellafragment` and `*.afm`.

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

3.3 DEPLOYMENT RECOMMENDATIONS

3.3.1 NETWORK

3.3.1.1 Latency: Client and Team Server

It is recommended to provide a network with the lowest possible latency between the client and the server: in the order of 1 to 10 ms for a round-trip.

3.3.1.2 Latency: Team server and DB server

It is strongly recommended that the Team server and the DB server are located on the same physical server as latency between the Team server and DB server will impact greatly the overall performances of the solution. As such the best performing deployment is achieved by using the H2 database in embedded mode with its .db database file located on the same disk than the Team server.

If there is a requirement on the database that prevents from using H2, make sure that the latency is as low as possible.

3.3.1.3 Network stability

VPN are not recommended (it is a latency factor) as well as other network elements that could drop connections which are more or less inactive. As such wireless connection are also not recommended as any loss of connectivity might lead to instability in the product and loss of data. However, if a network element of this kind is mandatory, an SSH tunnel could be used as a workaround to avoid client/server disconnections.

3.3.1.4 Server isolation

It is strongly discouraged to deploy the server on a public WAN. Team for Capella should be the only way to edit the information stored in the database.

3.3.2 SCALABILITY AND SIZE OF MODELS

Scalability and performances are highly dependent on the design of the domain metamodel, the implementation of this metamodel and the Viewpoint Specification Models. The following figures are given with an Ecore model and the EcoreTools tooling which applies the Sirius best practices.

The minimum physical memory dedicated to the Team server is 4 GB for a deployment where the expected model size is in the order of 300 000 model elements. The heap memory available for the server should be increased to support bigger models: 8GB should support 600 000 model elements.

The memory usage of the clients will increase when the model which is shared among the clients grows as such these resources might need to be increased for larger models with 8GB being expected for models with 600 000 elements (the exact value might vary depending on the amount of information each model element holds).

The latency of end user operations requiring a full analysis of the model increase as the model grow, this includes: opening and closing a project, deleting model elements and representations, launching a transformation or a code generation. Opening a project (and hence collecting the model from the network) might take up to 1 min for a model with 500K elements.

Models having 1 000 000 model elements are the considered the upper limit for a Collaborative Server deployment.

A given server is expected to be used by 10 to 20 users simultaneously depending on their level of activity.

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

3.3.3 DISCLAIMER

Notwithstanding what was stated previously, Team for Capella product is not warranted to run without any error or interruption. Obeo does not make any warranty regarding the statements that are under the chapter «Deployment Recommendations», this chapter is provided for information purposes.

You acknowledge and accept the risks involved in using these products which could include without limitation, down time, loss of connectivity or data, system crashes, bad performances or performance degradation.

3.4 INSTALLATION PROCEDURE

The date/time and the time zone of the server must be correct to make the scheduler work as expected.

To use existing models in a new version of Team for Capella, copies of these models have to be kept (in files format) before removing the old version. Once the new version is installed, the migration procedure will be performed on these models.

3.4.1 TEAM FOR CAPELLA SERVER INSTALLATION PROCEDURE

3.4.1.1 Installation

Preparation steps:

1. Extract the archive `TeamForCapella-6.1.0_202509-win32.win32.x86_64.zip` in a given directory. It contains a `TeamForCapella` directory with 5 sub-folders and 1 file:
   - `lic-server`: contains a floating license server which allows several users to share the same product licenses. Each license can be used by only one user simultaneously.
   - `server`: the Team for Capella server
   - `tools`: contains some scripts, properties files and pre-configured jobs to configure a Jenkins as `scheduler`
   - `updateSite`: the Team For Capella update site for the client and an update site with feature patches for Capella
   - `license.html`
2. Download the Capella 6.1.0 bundle from https://www.eclipse.org/capella/download.html (Windows).

Client Installation

1. Unzip Capella bundle.
2. Move the `capella` and `samples` folders into the `TeamForCapella` directory.
3. Resulting structure of `TeamForCapella`
   - `capella`
   - `lic-server`
   - `samples`
   - `server`
   - `tools`
   - `updateSite`
4. Navigate to the `tools` folder and execute `installTeamForCapellaInCapella.bat`

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

The installation script will install the Team for Capella features in Capella, update the splash screen and update some properties in `capella.ini` and `config.ini`.

It is configured by default to retrieve the Team For Capella update site in the folder:

`TeamForCapella\updateSite`

The `-repository` property can be updated in the script to reference it from another location.

This Capella client (`capella` folder) should be used only for the Scheduler jobs: it must not be moved or renamed as its .exe files are referenced from the pre-configured jobs (Scheduler) and scripts (`tools` folder).

It can also be zipped and provided to user in case of local client installation, see § 4.2.1 Team for Capella Client installation procedure.

In remote clients installation, you need to **copy the full** `capella` **folder** and **rename it into** *`capella_client`*. Then this client can be started on Windows Server and accessed with Remote Desktop. If you want to install additional functionality, it will have to be done on `capella_client` and will not impact the *`capella`* folder.

3.4.1.2 Extensions installation

If meta-model extensions or add-ons are needed, use **one** of the following ways to install them:

- Use the Help > Install New Software… wizard if the extension are provided as update sites
- Otherwise if they are provided as dropins
    - Either unzip/copy their binary files in the folder

      `TeamForCapella\capella\dropins`
    - Or:
        - Unzip/copy them in any folder (it can be a shared folder between this server installation and client installations)
        - Modify the configuration file `TeamForCapella\capella\capella.ini` by adding the following parameter, **after -vmargs** :

```
-Dorg.eclipse.equinox.p2.reconciler.dropins.directory=<ExtensionFolder>
```

Exactly the same extensions have to be installed on **and on all clients** and on the server (`capella` and `capella_client`).

3.4.1.3 Jenkins scheduler installation

Team for Capella provides applications to manage the CDO repositories with the shared projects.

These applications can be triggered from Capella, but Team for Capella also provides Jenkins jobs in order to manage them with a web interface.

See how to install Jenkins and our specific jobs in the documentation ***Team for Capella Guide > System Administrator Guide > Jenkins Installation***.

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

3.4.1.4 Administration features installation

For system administrators, it is useful to install administration features such as the "Durable Lock Management" and "User Management" views and the "User Profiles" feature to manage users access rights.

Two ways to install these features in the Team for Capella Client:

- (recommended) Execute the script `TeamForCapella\tools\installAdminFeatures`
- install them manually from the Team for Capella client
    - "Help > Install New Software"
    - select the T4C update site and check the features under "Team for Capella - Administration".

For more details about administration views and User Profiles, please refer to the documentation in ***Team for Capella Guide > System Administrator Guide > Server Administration > Administration Views*** and ***Team for Capella Guide > System Administrator Guide > Access Control (User Profiles)***.

3.4.2 LICENSE SERVER INSTALLATION

3.4.2.1 Server installation

1. The license server is provided in the `TeamForCapella-6.1.0_202509-win32.win32.x86_64.zip` archive.
   After the preparation steps (see § 3.4.1.1), it is located in the folder
   `TeamForCapella\lic-server`

2. Unzip the `OLS.zip` archive in `TeamForCapella\lic-server\OLS`, the `OLS` folder should contain 4 `.ols` files.

3. You can choose to:
    - either launch the license server from the sheduler's job `License Server - Start` (disabled by default)
    - or directly launch the tool using the command `lic-server -keys ./OLS -verbose`

Additional parameters and documentation can be found in the `DOCUMENTATION.txt` file located in `TeamForCapella\lic-server\`.

The server configuration and more especially the `.ols` files should not be modified, moved or even accessed while the server is operating. Stop the license server before doing such operation.

3.4.2.2 Client configuration

In order to connect Team for Capella instances with the license server, a connection key must be used to retrieve its address and port.

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

You should have received this key with the license server bundle and your `.ols` files. If not please contact the support and provide the IP address and port to use to connect to the machine that will host the server in your local network.

The `capella\capella.ini` file has to be modified with the following line to define the configuration of the server (after `-vmargs`);

```
-DOBEO_LICENSE_SERVER_CONFIGURATION=<connection key>
```

This must be done on all clients.

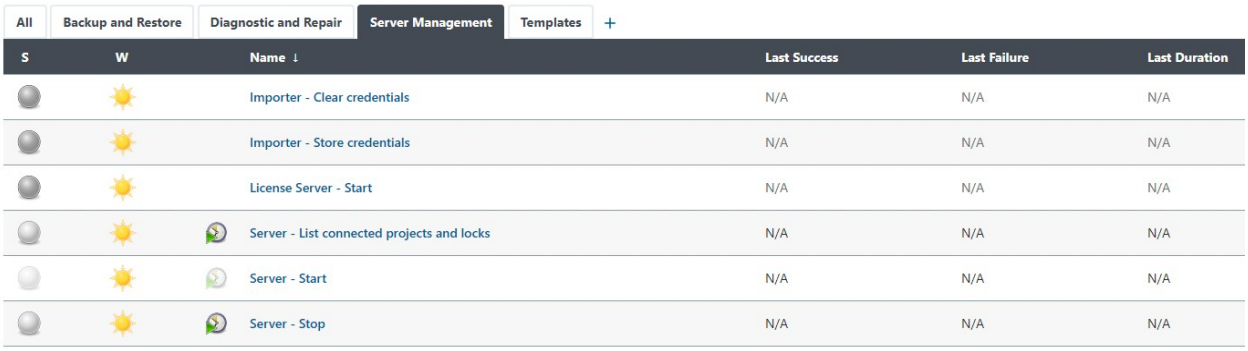
A license token is retrieved at the first connection attempt done by Team for Capella. It is then revoked when the last connected Capella project is closed or when Team for Capella closes. This license is verified from times to times with the server while you are using Team for Capella.

Clients are programmatically throttled and will not send requests to the server more frequently than once every minute. This throttling means some of the client actions might have a delay before being distributed to the server, for instance if one user stops using a given feature, 2 minutes of delay at most can be necessary for another user to be able to get the token.

The client/server communication is request based, no connection is kept alive for longer than just a request.

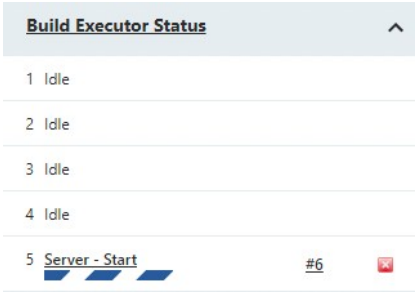
3.4.3 INSTALLATION VERIFICATION

1. Connect to the scheduler admin page using the default URL http://localhost:8036,
2. Check that the job `Server - Start` has been automatically launched.



On the scheduler main page, the job `Server - Start` should appear the Build Executor Status section:

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023



3. Once the server is started, it can be used to check the Team for Capella Client installations (see the Team for Capella Client installation verification procedure in §4.2.3),
4. Launch the `Server - List connected projects` and check its Console output,
5. Launch the `Database – backup` job from the `Backup and Restore` tab,
6. Stop the server using the `Server - Stop` job → the `Server - Start` job should be stopped.

3.5 TEAM FOR CAPELLA SERVER CONFIGURATION

The Team for Capella Server is provided with a basic configuration. You can find all the documentation to configure it further in ***Team for Capella Guide > System Administrator Guide >Server Configuration.***

To manage your repository with its shared project it is recommended to check the documentation of the Jenkins scheduler in ***Team for Capella Guide > Project Administrator Guide > Jenkins Configuration.***

3.5.1 HOW TO REUSE PREVIOUS SERVER CONFIGURATION WHEN CONFIGURING V6.1.X SERVER

When installing a new version of Team for Capella, some parts of the server/client configurations of the previous version can be reused.

3.5.1.1 Procedure

- Manually import your data or launch the backups jobs.
- Launch the `Server – Stop` job, terminate the `License Server - Start` job or end tasks `server.exe` and `lic_server.exe` in task manager (processes sheet)
- If you were using a Team for Capella version before 6.1.x, stop `TeamForCapellaScheduler` (service sheet) and remove it (see §3.6 - Uninstallation procedure).
- Keep the entire Team For Capella installation which contains all the file you may reuse.
- Choose among the following configuration you want to reuse

3.5.1.2 Server CDO

- Copy part of the `C:\xxx\T4C\server\configuration\`**`cdo-server.xml`** file which contains
  - the acceptor tag can be reused
  - the repository configurations can be reused except for *userManager*, *securityManager* tags.

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

- if the server was configured with the authenticated configuration, the content of `C:\xxx\T4C\server\configuration\**users.properties**` can be kept as is.
- If the server was configured with the user profile configuration, the user profile model can be reused. Follow the steps described in **Team for Capella Guide > System Administrator Guide > Access Control > User Profiles > Import/Export User Profile Model**.
- You may also consider all properties or parameters you may have set in the server.ini file.
- If you are doing a new installation with the same version, you can reuse the database. Copy the `C:\xxx\T4C\server\eclipse\db` folder. Note that the database cannot be reused if the *Audit mode* activation changes between both installations.

3.5.1.3 Scheduler

- In previous versions (before 5.1.0, included), the scheduler was packaged inside Team for Capella, in the `C:\xxx\T4C\scheduler\jenkins_home` folder
  - `config.xml` contains the global Jenkins configuration which can be reused
  - `jobs\` contains the jobs definition with the build history. `jobs\<xxxjob>\config.xml` can be reused modulo some parameter changes that may be needed (for example for importer executable call).
- Since 5.2.0, Team for Capella does not pack a third party tool to manage the repository content (such as Jenkins). You will need to access its installation folder or *Home* directory.
  -
- Customization made on those files/folders might be reported on your new custom installation made for this version.
- Check the modification done in each job with the **Job configuration history** available in each job .
- It is possible to create a `v5.2` folder in the Scheduler interface, move the existing jobs in this folder, and check the Jenkins scheduler installation section to update Jenkins to the recommended version and install the new jobs. Then compare your v5.2 jobs with v6.1 jobs.

3.5.1.4 Tools

- You may consider all system properties or parameters you may have set in the **importer.bat** file (or **importer.ini** for older versions) . That information have to be transferred and adapted
  - From `C:\xxx\T4C14\capella\eclipse\**importer.bat**` or `C:\xxx\T4C52\tools\**importer.bat**`
  - To `C:\xxx\T4C\tools\**importer.bat.**`
- Do the same for other scripts you have modified In `C:\xxx\T4C52\tools\`: **`maintenance.bat,command.bat`**.

3.5.1.5 Client (in remote clients installation)

- You may consider all properties or parameters you may have set in the eclipse.ini file. That information have to be transferred to `C:\xxx\T4C\capella_client\**capella.ini.**`
- If the IP adress of the server did not change, you can keep the `OBEO_LICENSE_SERVER_CONFIGURATION` property system (just copy `-DOBEO_LICENSE_SERVER_CONFIGURATION=116168566473385529985114844.`)

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

3.5.1.6 License server

- The license server version is now provided in the Team for Capella archive.
- The `*.ols` files may have been thought up to be reusable for your new Team For Capella version. In this case, you can keep your `*.ols` files.

3.5.1.7 User Accounts Management

Several modes of access control can be used for each repository on the server:

- Identification (default mode): each user defined in the file `user.properties` is authorized to read and/or modify all models present on the repository.
- User Profiles: discriminating user rights are defined in a User Profiles model stored in the database.
- LDAP Authentication: this mode allows to authenticate with a LDAP server. It can be also used with authenticated or with user profiles.
- Not Authenticated Access: anyone can read and/or modify all models on the repository.

For the complete access control configuration documentation, refer to the documentation **Team for Capella Guide > System Administrator Guide** in chapters:

- ***Access Control***
- ***Server Configuration***
- ***Server Configuration > Not Authenticated Configuration***
- ***Server Configuration > Authenticated Configuration***
- ***Server Configuration > Activate LDAP Authentication***
- ***Server Configuration > Activate OpenID Connect Authentication***
- ***Server Configuration > Activate WebSocket Connection***
- ***Server Configuration > Activate SSL Connection***
- ***Server Configuration > User Profile Configuration***

In the current version, Team for Capella is configured with the “Identification” access control mode, ie. passwords are not encrypted. Refer to ***Team for Capella Guide > System Administrator Guide > Access Control*** if you need Authentication or Authorization mechanisms.

3.6 UNINSTALLATION PROCEDURE

To uninstall Team for Capella:

1. Do a backup of projects stored on the Team for Capella Server,
2. Stop the server (see How to stop the Server),
3. Stop the scheduler (see How to stop the Scheduler),
4. If a Windows Service was created, remove it:
   a. Using a Windows Command Line (as administrator), go to the scheduler directory,
   b. Execute the following command: `sc delete TeamForCapellaScheduler` (alternative command: modify the `winservice.bat` to comment the installation steps and uncomment the uninstallation steps).

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

5. If you installed a third party tool to manage your repository, such as Jenkins, uninstall it.
6. Remove the installation directory.

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

4 TEAM FOR CAPELLA CLIENT INSTALLATION

4.1 REQUIREMENTS

4.1.1 RECOMMENDED SYSTEM REQUIREMENTS

For successful installation of Team for Capella Client, your computer must meet the following requirements:

- 2 GHz processor
- 3 GB for (for big models, 4GB RAM)
- Microsoft Windows 10/11 64 bits
  - no known issues with Windows 7/8
- Compatibility with Java Runtime Environment 17
  (Eclipse Adoptium Open JDK 17.0.6 is provided by Capella)
- Virus scanner should not be activated on Capella models files:
  `*.aird`, `*.capella`, `*.airdfragment`, `*.capellafragment` and `*.afm`.

4.2 INSTALLATION PROCEDURE

4.2.1 TEAM FOR CAPELLA CLIENT INSTALLATION PROCEDURE

4.2.1.1 DEFAULT CLIENT INSTALLATION

You may take your Team For Capella client as it has been installed during the server installation. In that case, you simply need to make an archive of the `capella` folder and distribute it to users. Do not forget to include the `samples` folder if you want to ship the example model.

For Windows and Linux, it possible to follow the server installation procedure without the server nor Jenkins deployement and keep only the `capella` folder in the end.

Before archiving the Capella client that will be deployed on others computers, don't forget to remove the workspace used for the installation from the recent workspaces list (`Window > Preferences > General > Startup and Shutdown > Workspaces`).

It is also possible to manually install Team for Capella but note that the installation with the `tools/installTeamForCapellaInCapella.bat` is recommended.

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

4.2.1.2 MANUAL CLIENT CLIENT INSTALLATION

To manually install Team for Capella client in a Capella bundle (Windows/Linux/macOS), please follow the next steps:

1. Extract the archive TeamForCapella-6.1.0_202509-win32.win32.x86_64.zip and keep only the `updateSite` and `tools` folders.
2. Download and unzip Capella 6.1.0 bundle (https://www.eclipse.org/capella/download.html)
3. Launch Capella (capella.exe for Windows, capella for Linux, Capella for macOS)
4. Click Menu *Help/Install New Software…* , add the archive file `com.thalesgroup.mde.melody.team.license.update-xxx.zip` in the `updateSite` folder and select features in the *Team for Capella* category:
   - Team for Capella
   - User Interface to request a License
   - Other features from *Team for Capella* and *Team for Capella – Administration* categories are optional and not installed by the installation script.
5. Click Menu *Help/Install New Software…* , add the archive file `com.thalesgroup.mde.melody.team.license.update.patch-xxxx.zip` in the `updateSite` folder and select the feature patches for Capella and open source components.
6. Then you need to do the additional tasks performed by the installation script:
   - change the splashscreen:
     - replace the value of `osgi.splashPath` by `platform\:/base/plugins/com.thalesgroup.mde.melody.collab.ui` in `capella\configuration\config.ini` (`Capella.app/Contents/Eclipse/configuration/config.ini` for macOS)
   - copy the `license.html` file from `tools to capella` folder
   - avoid performance impact of CDO internal tracing:
     - add `-Dorg.eclipse.net4j.util.om.trace.disable=true` at the end of the `-vmargs` section in `capella\capella.ini` (`Capella.app/Contents/Eclipse/capella.ini` for macOS)
   - avoid third parties components logging noise
     - add `-Dlogback.configurationFile=configuration/logback.xml` at the end of the `-vmargs` section in `capella\capella.ini` (`Capella.app/Contents/Eclipse/capella.ini` for macOS)
     - copy the `logback.xml` file from `tools/resources/client_rootfiles` to `capella/configuration` folder (`Capella.app/Contents/Eclipse/configuration/` for macOS)
   - add possibility to override default repository connection information:

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

- add `-pluginCustomization pluginCustomization.ini` before the `-vmargs` section in `capella\capella.ini` (`Capella.app/Contents/Eclipse/capella.ini` for macOS)
- copy the `pluginCustomization.ini` file from `tools/resources/client_rootfiles` to `capella` folder (`Capella.app/Contents/Eclipse/` for macOS)

- add the `license.html` in capella folder, copied from `tools\license`.

7. Add the license server configuration key as described in §3.4.2.2 - Client configuration.
8. If you have installed Team for Capella in a Capella bundle already completed with some add-ons, do not forget to check if they provide a “CDO Feature” for Team for Capella compatibility.

Steps 4 and 5 are done by the installation script.

4.2.2 Team for Capella Client configuration

For the complete client configuration documentation, refer to the chapter ***Team for Capella Guide > User Guide > Client Configuration > Client Preferences***

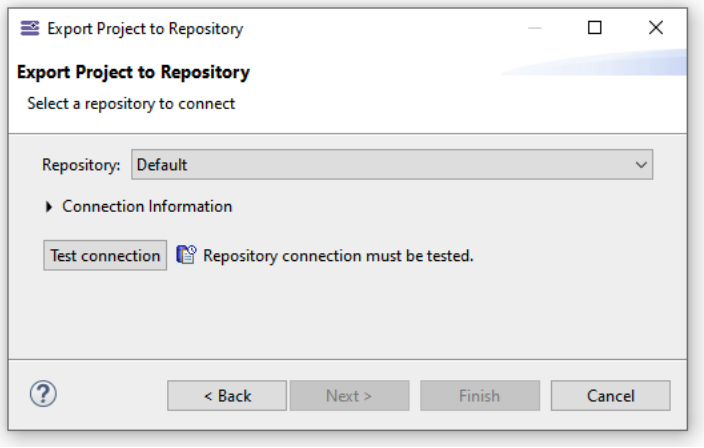
1. Launch Team for Capella (“capella/capella.exe”),
2. Optional: clean the user’s Secure Storage (it contains the save Login/Password, if “Remember Me” option was used):
   a. Go to menu Window > Preferences > General > Security > Secure Storage,
   b. Open the “Contents” tab,
   c. Select “[Default Secure Storage]”,
   d. Click on “Delete”,
   e. Upon request, restart the Team for Capella client.
3. Go to the menu Window > Preferences > Sirius > Team collaboration
   a. Set the server location with the hostname or the IP address of the Team for Capella Server (`localhost` if the server is setup on the same machine),
   b. Click on “Apply”

4.2.3 Verification installation procedure

1. Create a new project → Right click in the Capella Explorer → New → Capella Project
   a. Call it Test for example → Finish
2. Export the project to the remote repository
   a. Right click on the project → Export → Team for Capella → Export model to remote repository→Next,
   b. (Optional) Expand *Connection Information* if the deployed repository with default parameters

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

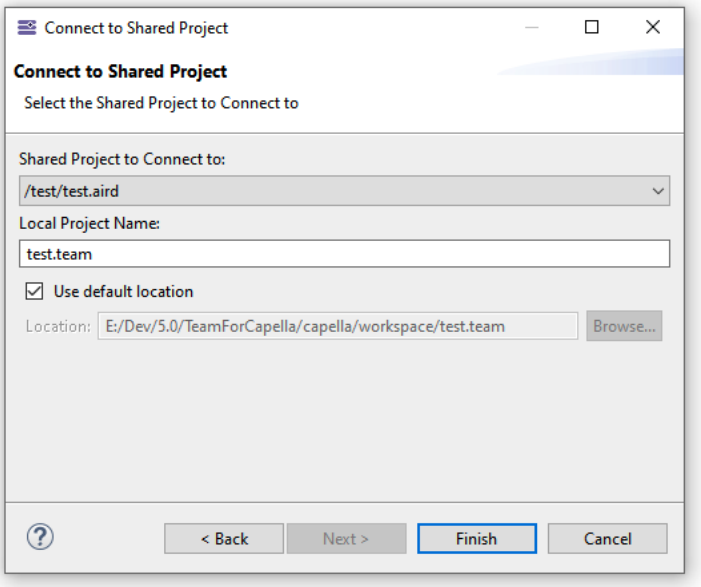
c. Click on Test Connection,



d. Provide the user name and password (by default: user1/user1, user2/user2, user3/user3, admin/admin) → See chapters Default Server Configuration and User Accounts Management to customize users/passwords,

e. Click on Finish.

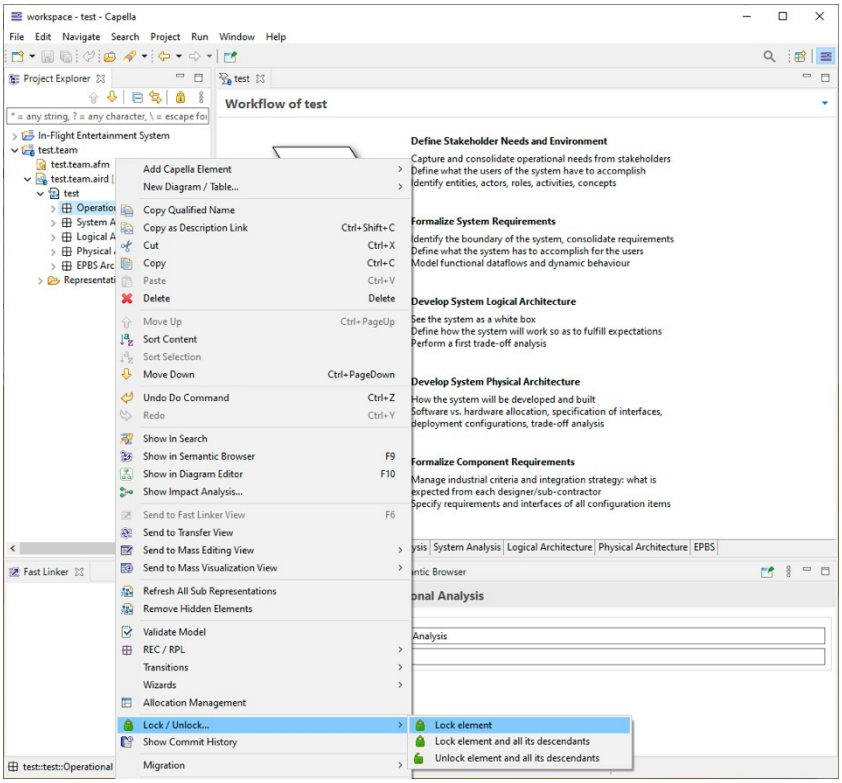
3. Connect to the remote project previously exported,
   a. Right click in the Capella Explorer → New → Capella Connected Project
   b. Test Connection,
   c. Select the Shared project from the list,



d. Click on Finish,

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

4. You should now be able to work on the project on the remote repository.



This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

5 RELEASE NOTES

The Team for Capella 6.1.0_2025-09 release notes are available at:

https://www.obeo.fr/en/team-for-capella-releases - 6.1.0_202509

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

6 MIGRATION OF EXISTING PROJECTS

6.1 VERSION COMPATIBILITY

Team for Capella	(based on) Capella	(third party software) Sirius
1.0.3 (32bits & 64bits)	1.0.3	Sirius 3.1.6
1.1.4 (32bits & 64bits)	1.1.4	Sirius 4.1.9
1.2.2 (64bits)	1.2.2	Sirius 5.1.4
1.3.2 (64bits)	1.3.2	Sirius 6.1.4
1.4.0 (64bits)	1.4.0	Sirius 6.3.0
1.4.1 (64bits)	1.4.1	Sirius 6.3.1
1.4.2 (64bits)	1.4.2	Sirius 6.3.3
5.0.0 (64bits)	5.0.0	Sirius 6.4.0
5.1.0 (64bits)	5.1.0	Sirius 6.5.0
5.2.0 (64bits)	5.2.0	Sirius 6.6.0
6.0.0 (64bits)	6.0.0	Sirius 7.0.1
6.1.0 (64bits)	6.1.0	Sirius 7.1.0
6.1.0_202509 (64bits)	6.1.0	Sirius 7.1.0

Table 2 - Version Compatibility (sorted by delivery dates)

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

6.2 MODEL MIGRATION FROM PREVIOUS VERSION TO V6.1.

To use a previous version model in Team for Capella v6.1, a migration must be done to be compliant with the new metamodel and file extensions.

This model migration is provided by Capella and must be done between each minor version (from v1.4.x to v5.x, for example).

The process to migrate a model to v6.1 from a shared repository follows the following steps:

1) If Team for Capella users have created local diagrams (i.e.: diagrams stored in the local .aird file), they have to move all diagrams they want to keep to a remote `.aird` or `.airdfragment` ("**Move Diagrams**" action),

2) Import locally the model to migrate ("**Import… / Team for Capella / Import model from remote repository**") using previous version (v5.x) of Team for Capella. Make sure that associated team server is running before importing from remote repository. Another way is to use the last valid import of the Scheduler.

   ➔ Make a baseline of the imported model,

3) Migrate the previously imported model ("**Migrate Project toward current version**") in a Team for Capella v6.1. (The migration can also be done in Capella v6.1). To do so, please refer to *Migration* dedicated chapter of Capella online installation guide,

   ➔ Make a baseline of the migrated model,

4) Export the migrated model to the Team for Capella server v6.1 ("**Export… / Team for Capella / Capella Project to Remote**") using Team for Capella v6.1. Make sure that associated team server is running before exporting to remote repository.

After performing these steps, the model in the shared repository is in right version. Team for Capella has to be upgraded on client's computers. Then users can connect and work on the model. They do not need to do any migration.

6.3 MODEL MIGRATION FROM AN OLDER VERSION

The model migration is necessary between each version considering only minor and major version change.

- For the first model migration, you need to reproduce only steps 0, 1 and 2 described above.
- For the following intermediary model migrations, you need to reproduce only step 2.
- For the last model migration, you need to reproduce only steps 2 and 3

For example is you start from v1.4.2 model

- do steps **0**,**1** and **2** with *previous version* = *1.3.2* and new version = 1.4.2,
- do step **2** with *previous version* = *1.4.2* and *new version* = *5.2.0*
- do steps **2** and **3** with *new version* = *6.1.0.*

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

7 MIGRATON FROM V6.1.0 TO V6.1.0_202509

A full new installation is recommended.

Installation:

- An additional update site with the feature patches for Capella/Kitalpha/Sirius/GMF has been included in bundle.
- Installation script has been modified to install the feature patches during Team for Capella features installation in Capella bundle.

Scheduler:

- It is possible to do reuse current working jobs after updating the TEAMFORCAPELLA_APP_HOME if the new version has been installed at a different location
- Jobs for Linux have been corrected to add Xvnc related configuration by default in jobs which requires Capella runtime to start
- Start repository jobs have been simplified to remove `-httpXXX` which are already in `command.sh/bat` or in secure storage
- `Database - Backup` job (Windows) has been corrected to use Jenkins workspace

Server:

- DB files and dynamic repositories from v6.1.0 can be reused in v6.1.0_202509
- Repositories with OpenID Connect authentication
    - technicalUsers.properties files must be cleared: password are no more stored in clear text but encrypted with BCrypt.
    - Technical ssers can now be added/updated/removed without restarting the repository, via the REST api.
    - New configuration capabilities are available, see release notes.
- REST Admin server realm passwords are now encrypted with BCrypt.
    - Secure storage must be cleaned from all Team for Capella keys: `fr.obeo.dsl.viewpoint.collab.credentials.*`
    - Before launching the server, delete `TeamForCapella/server/configuration/fr.obeo.dsl.viewpoint.collab.server.admin/secret.txt`
- Credentials and passphrases can now be injected via environment variables instead of being stored in clear text in multiple configuration files. See available variables in ***Team for Capella Guide > System Administrator > Server Configuration > Passing parameters using system property or environment.***

Client and server-side client

- Feature patches for Capella and open source components are installed during installation phase.

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

8 PERFORMANCE CONSIDERATION FEEDBACK

This chapter will provide you some useful information about the performance you should expect and about performance on production context (with big sized model and many users)

8.1 REFERENCE TIME

You may wonder, if the performances you have are normal, that is, if the time elapsed, when doing usual action in Team for Capella, is the right one compared to what should be expected.

Ensuring that the elapsed time doing your actions is the expected one, will help you to ensure that there are no issues that could come from other cause like network lag, lack of memory, antivirus process etc..

8.1.1 Case Studies

Some studies have been done on reference Capella models:

- The open source *IFE* model packaged with Capella
- *Combined IFEs*, a modified version of *IFE* which is artificially swollen to have an important sized model.

The next figure presents some characteristics on those models and the expressions used to compute them.

Characteristics	**Project**	
	IFE	**Combined IFEs**
File (size in KB)		
Project	19 179	203 310
.capella	1 475	13 464
.aird	17 705	189 846

Model (number of elements)			**AQL expressions to use in Interpreter view**
Capella elements	5 378	49 389	aql:self.eAllContents()->size() *select the first child of the .aird file in the Project Explorer*
Components	110	1 230	aql:self.eAllContents(cs::Component)->size() *select the first child of the .aird file in the Project Explorer*
Functions	224	2 652	aql:self.eAllContents(fa::AbstractFunction)->size() *select the first child of the .aird file in the Project Explorer*
Diagrams	111	880	aql:self.eResource().getContents()->filter(diagram::DDiagram)->size() *select a diagram in the Project Explorer*
Tables	1	8	aql:self.eResource().getContents()->filter(table::DTable)->size() *select a diagram in the Project Explorer*
Elements displayed in all representations	6 384	70 850	aql:self.eResource().getContents()->filter (viewpoint::DRepresentation).eAllContents (viewpoint::DRepresentationElement)->size() *select a Diagram in the Project Explorer (only for local projects)*

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

Table 3 - Reference models

Refer to See §8.3 - How to get characteristics of your model to see how to get the characteristics of your own model in order to compare it with those reference models.

8.1.2 MEASURES

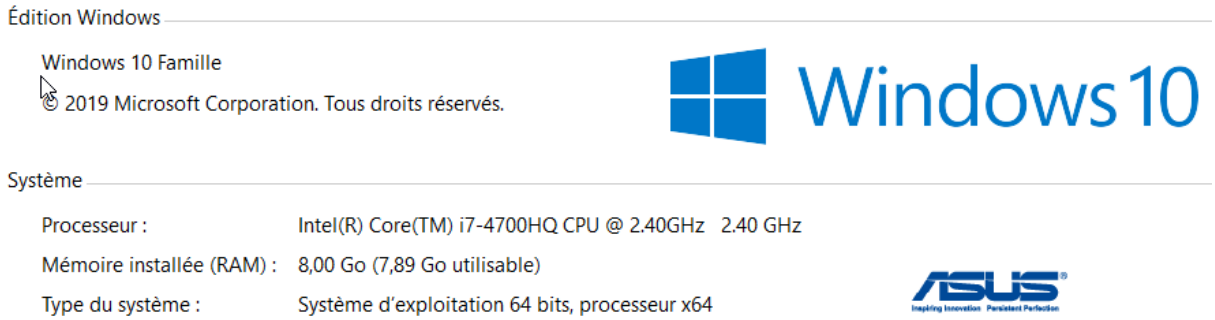
Measures contained in this document give some idea and order of magnitude of the performances expected for comparable environments.

They have been taken on two different deployments:

- Local: single computer with both client and server - *IFE* and *Combined IFEs*
- OVH: two virtual machines, one for the client and one for the server as recommenced in §3.1 -*Combined IFEs*.

8.1.2.1 Computer used for the tests

Local Windows computer



OVH Windows computer

8.1.2.2 Reference elapsed time

The memory allowed to the Team for Capella client can be configured with the *capella.ini* file beside *capella.exe* executable.

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

Elapsed time (s)	***IFE***	***Combined IFEs***
	Xmx 3GB	Xmx 3GB
Export To Server	28,7	114
Open Session (first time)	3,7	9
Test Close Session	1	3
Open Session	2	6,5
Open the diagram *(1)*	4	6
Create a LFBD diagram on Root Logical Function. *(2)*	3	6
Save the session after steps from *(3)*	2	7
Test Close Session	1	3
Test import local for remote	9	122 *(4)*

(1) Open *[PAB] [BUILD] All Pcs, Pfs, Fes* .
It is a diagram containing 486 DRepresentationElement for both *IFE* and *Combined IFEs*.
(aql:self.eAllContents (viewpoint::DRepresentationElement)->size())

(2) The created diagram will contain 121 DRepresentationElement for *IFE* and484 DRepresentationElement for *Combined IFEs*.

(3) Steps to do before save: Save Session + Transition a LF from Logical layer to Physical Layer (with Apply) + create a PFBD on root PF

(4) This used to require 4GB of Ram with 1.4 stream to be in success

Table **4** - Measured time on test models (5.0.0)

8.1.3 Analysis and conclusion

The majority of the scenarios could be done with the minimum recommended memory of 3Go.

Nevertheless, with model of big size (*Combined IFEs*), we need more RAM to work with constant performance for some actions like *Refresh all representations* or *Validate* on the whole model.

If the *Team for Capella* Client reaches, at execution, the maximum allowed memory, the performance may drastically fall particularly for actions that consume a lot of memory such as:

- Import a Capella project locally from the server
- Do the V*alidation* of the whole project
- Start *Refresh All Representation* action
- Save if many objects have been changed or deleted
- Some particular Capella functionality such as *functional transition*

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

8.2 DEDICATED TESTS ON *COMBINED IFES*

The aim of this test session was to ensure that Team for Capella is able to have to keep the right performances after having used for a while.

- *Combined IFEs* is used as the Capella Project. It is a large sized project.
- 20 users are connected to the Capella project on the server
- 8 users are really working on the model. During the test, they will do standard actions such as
    - Open many diagrams
    - Create, Refresh, modify diagrams
    - Make functional transition for some of them
    - Regularly create semantic element through diagram or not
    - Use tool in diagram palette
    - Use the *Activity Explorer,* F8 and F9 to navigate
    - Validate some parts of the model

  But also, other actions that demands significant amount of RAM

- The tests have been done with 21 virtual machines hosted on OVH cloud (20 clients and 1 server):
    - Client Xmx=3Go
    - Server Xmx=4Go

8.2.1 SERVER

Throughout the test, the server never reaches its maximum allowed memory of 4Go RAM and not even 3Go. It keeps constant performance.

8.2.2 CLIENT

The client behaves constantly with good performance throughout the tests.

Nevertheless, the client may reach 3Go of memory if high memory consumption actions are performed such as:

- Import a Capella project locally from the server
- Do the V*alidation* of the whole project
- Start *Refresh All Representation* action

Once the maximum allowed memory is reached:

- The user keeps having good performance for standard actions
- But the performance may fall drastically for high memory consuming actions. For that cases, the performance issue can be solved
    - using more allowed memory for the client (Xmx=4 or more)
    - simply restarting the client

Note that the potential performance issue encountered by one user has no impact on another user

8.2.3 CONCLUSION

The model used is what we consider as a medium-big sized model.

20 users are working on it.

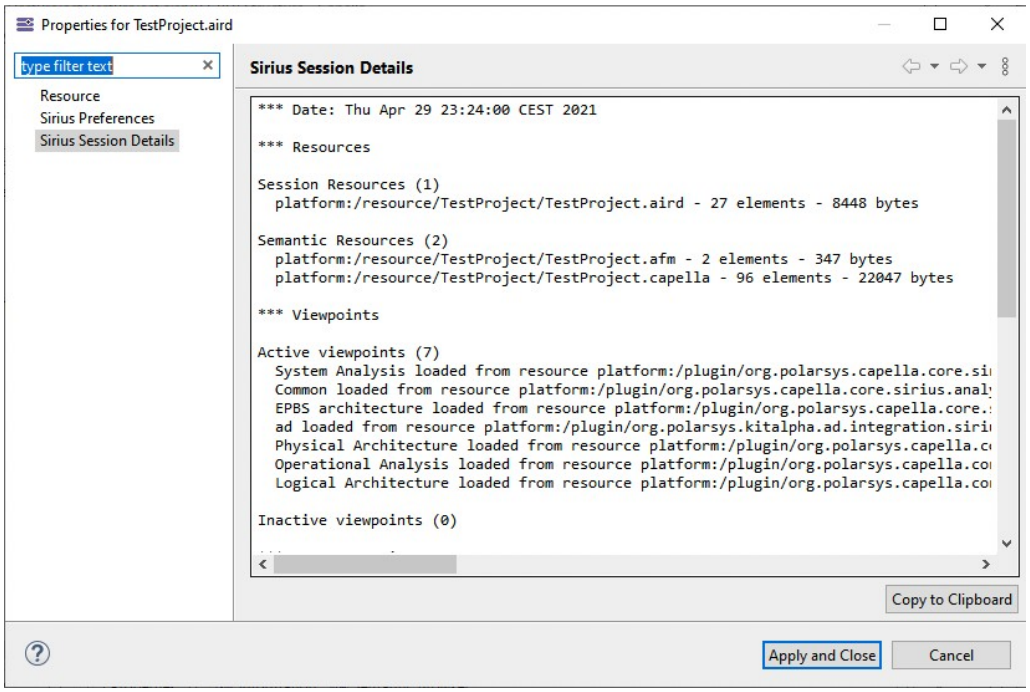
- The server 4Go is well fitted for the use

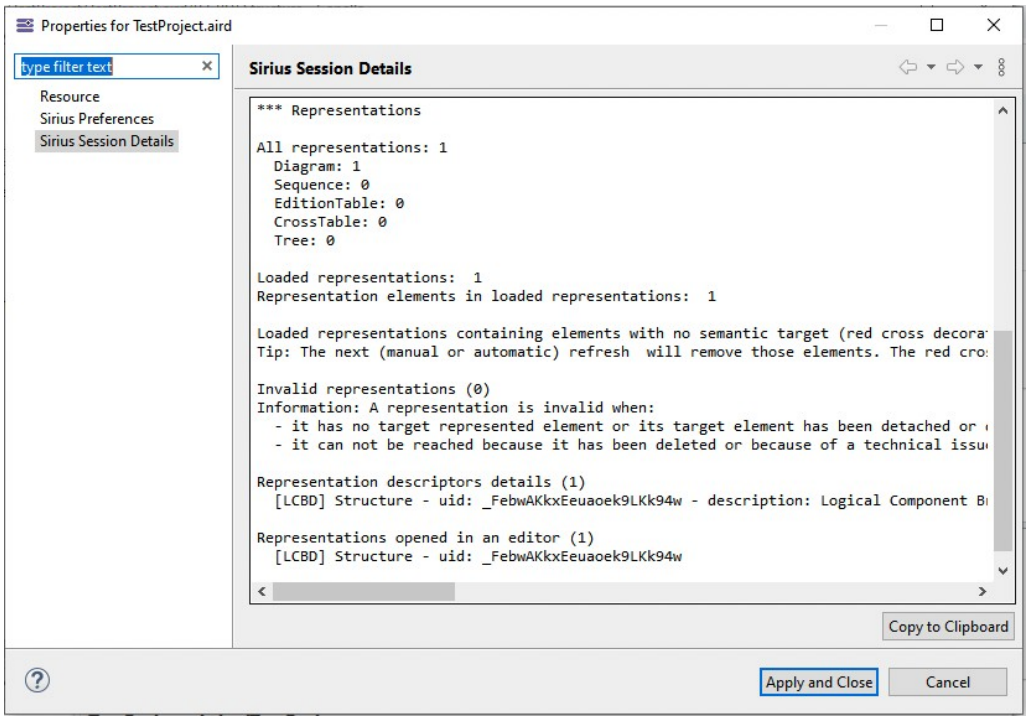
This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

- The client 3Go RAM is sufficient for most cases. It can be increased to have best performance when the user uses high memory consuming action

8.3 HOW TO GET CHARACTERISTICS OF YOUR MODEL

The simplest way to get characteristics of your model is to use the *Sirius Session Details* tab available in the *Properties* contextual menu of the .aird file of your project:



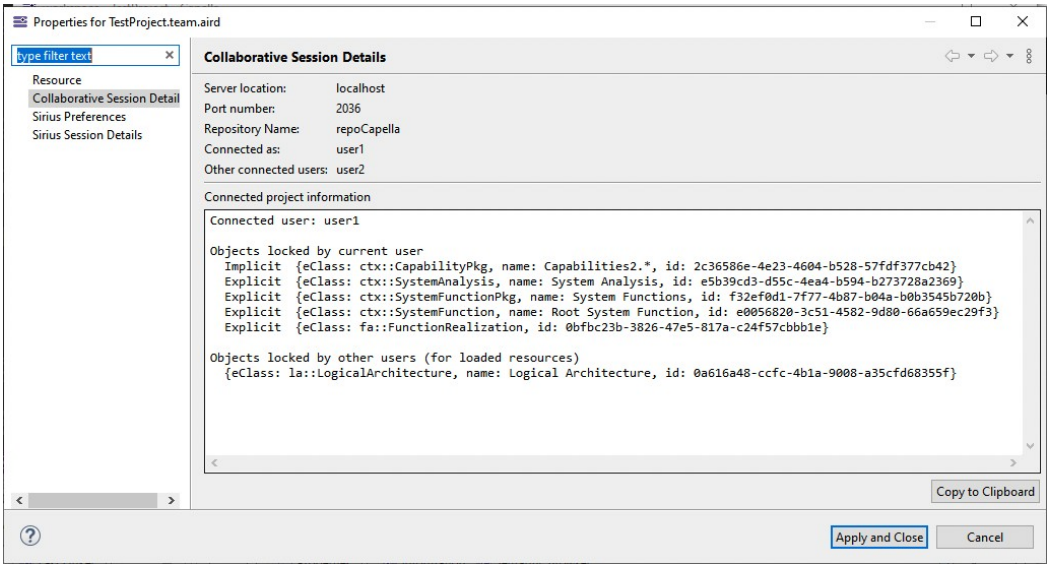


This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023

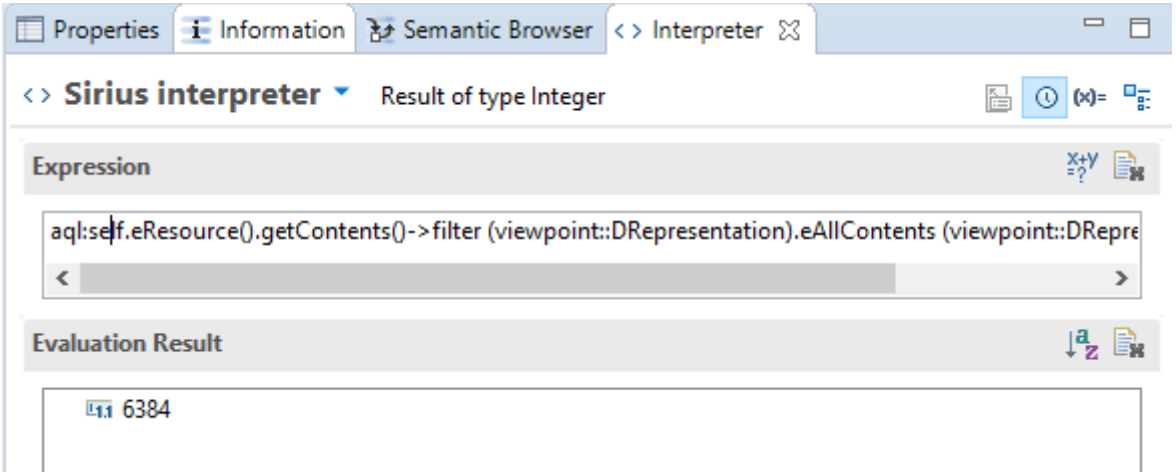
It will give you the following information about your project:

- Number, name, path and number of contained elements of each resources
- Selected Viewpoints
- Number of representations
- Some details about each representation (name type, id, status)
- Number of loaded representations
- Number of representations opened in editor

On shared projects, the *Collaborative Session Details* tab gives information about connected users and locks:



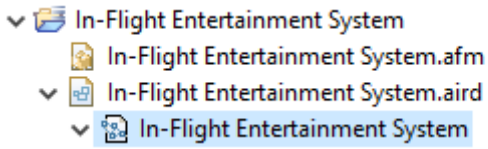
It is also possible to additional model characteristics thanks to the *Interpreter* view:



To get the characteristics of the *.capella* semantic resource :

- select any semantic element

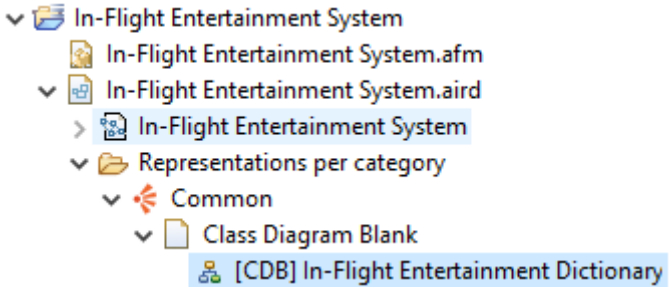
This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023



- write the AQL query in the *Interpreter* view to get the result in the lower part of the view.

To get the characteristics of the representations in the *.aird*

- select any representation



- write the AQL query in the *Interpreter* view to get the result in the lower part of the view

.

This document is not to be reproduced, modified, adapted, published, neither translated in any material form in whole or in part nor disclosed to any third party. © Thales 2023