User Guide
ARCAD Observer

Version 24.0

Publication Date: January, 2024

Prepared by the ARCAD Documentation Team

North America & LATAM
1 N. State St, 15th Floor
Chicago, IL
USA
1-603-371-9074
1-603-371-3256 (support calls only)
sales-us@arcadsoftware.com

EMEA (HQ)
55 Rue Adrastée – Parc Altaïs
74650 Chavanod/Annecy
France
+33 450 578 396
sales-eu@arcadsoftware.com

Asia Pacific
5 Shenton Way #22-04
UIC Building
Singapore 068808
sales-asia@arcadsoftware.com

Contact ARCAD

Headquartered in France at the foot of the Alps, ARCAD offers global services and has offices and partners all over the world. ARCAD partners with leading-edge companies throughout the world to offer full services, close to home.

Visit our website to Contact Us and find out more about our company and partners, or to request a demo.

The ARCAD Customer Portal is intended for current and potential customers that have full or trial versions of ARCAD software. If you already use or are interested in using an ARCAD product, the portal lets you view all of your current licenses and generate your own temporary license keys for most ARCAD products. It grants you access to the ARCAD product knowledge base (new releases, release notes and current documentation).

Do you have a request for change or have you encountered a bug? Log into the ARCAD Helpdesk and create a ticket.

ARCAD guarantees consultant support 24 hours a day, 5 days a week (24/5) to registered members. Calls received are redirected, according to the hour, to put you in contact with a support team in or near your timezone.

Country	Address	Account Contact	Support Contact
France	ARCAD Software (HQ) 55 Rue Adrastée 74650 Chavanod ARCAD Software 17 chemin de la plaine 07200, Saint-Didier-sous-Aubenas	+33 4 50 57 83 96 sales-eu@arcadsoftware.com	Worldwide 24/7: +1 603 371 3256 France only: +33 450 57 28 00 support@arcadsoftware.com ARCAD Helpdesk
Germany	ARCAD Software Deutschland GmbH c/o Pramex International GmbH Savignystr. 43, 60325 Frankfurt am Main		
China	ARCAD Software #2035, Yuehai Plaza, 180 Wanbo 2nd Road, Nancun, Panyu District, Canton	+86 (020)22324643 +86 (020)22324649 sales-asia@arcadsoftware.com	
India	ARCAD Software D-280/281/282, Vibhuti Khand Gomti Nagar, Lucknow		
Singapore	ARCAD Software 5 Shenton Way #22-04 UIC Building Singapore 068808		
USA	ARCAD Software 1 N. State St, 15th Floor Chicago, IL	+1 (603) 371-9074 +1 (603)-371-3256 (support calls only) sales-us@arcadsoftware.com	

Table 1: Contact ARCAD

Preface

Document purpose

This document is intended to guide you through using ARCAD Observer.

Intended audience

This document is intended for all ARCAD Observer users.

Related documentation

ARCAD technical documentation can be accessed from the product's online help or by logging into the Customer Portal on our website.

ARCAD Release Notes
ARCAD Glossary
ARCAD Installation Guide
ARCAD Observer Getting Started Guide

Table 2: Related documentation

Product information

ARCAD Observer 24.0 Copyright © 2024 ARCAD

Visit the website (ARCAD Observer)

Unless stated otherwise, all content is valid for the most current version of ARCAD Observer listed as well as every subsequent version.

Product version	Document version	Publication date	Update record
≥ 24.0	1.3	January, 2024	General documentation update
23.0	1.2	January, 2023	No functional changes

Table 3: ARCAD Observer User Guide publication record

Contents

CONFIGURATION

ANALYSIS

DIAGRAMS

DOCUMENTATION

Tables

Figures

ABOUT ARCAD OBSERVER

1 About ARCAD Observer

Cross-platform, graphical application analysis, along with a diagram generator and automatic generation of technical documentation
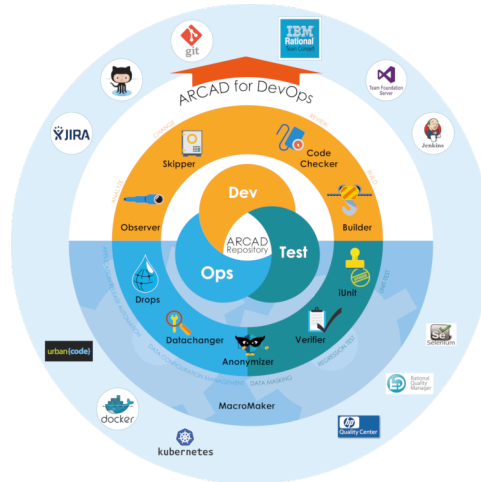
Figure 1: ARCAD Observer in the ARCAD product suite

Whether you are an experienced IT professional on IBM i or a newcomer to the platform, the task of understanding a legacy application before making further changes can be a time-consuming and daunting task.

ARCAD Observer automates application analysis to improve productivity and accuracy in software maintenance and modernization.

ARCAD Observer is the solution to three key issues:

- Re-documenting the Information System;
- Improving productivity during a project's impact analysis phase;
- Capitalizing on and disseminating knowledge of the Information System.

This is crucial for IT departments that:

- make modifications to their applications.
- develop new applications around their existing system (client/server or web).
- migrate to other systems.

1.1 Business context

Software development teams spend at least 50% of their time trying to understand an application's internal architecture before modifying it. Many applications on IBM i are years, even decades old. They have been modified many times by various developers who may no longer be with the company, and documentation is often missing or out-of-date. Sometimes changes have been made which were easy rather than well-done- which creates a heavy "technical debt" that increases the cost of making further changes. If you are involved in application maintenance, enhancement, modernization, or re-engineering, you need an efficient way to

quickly find dependencies within complex applications and understand their structure.

1.2 Functional description

ARCAD Observer is an Eclipse plug-in available for RDi and RTC, and pre-packaged in the ARCAD Client RCP. It is part of a plug-in package for Application Analysis.

ARCAD Observer application analysis features

Navigator: Point and click application navigator by view

Impact analysis: "Where-used", down to source line and field level

Multi-level, version specific view: Dynamic viewing and Query relative to a specific version

LPEX integration: Context-sensitive cross-referencing as developers edit, compile, debug, …

Diagrams: Observe and analyze your programs via I/O diagrams, workflow diagrams, database relationship models and program/procedure calling chains

- Database relationship diagram: Extraction of your underlying database relationship model
- I/O diagram, workflow diagram: View of dependencies between programs and files
- Calling chain diagram: View of calling chains, both programs and ILE procedures
- Macroscopic views: High-level functional view

ARCAD Observer application documentation feature

List manager: List generation, for managing application artifacts in groups

Documenter: Auto-generation of technical documentation in HTML

2 Main concepts

1. Configure documentation models.
2. Analyze your application: call chains, cross-references, search for components, etc.
3. Create work lists of components, generate diagrams, everything that will be used to generate documentation.
4. Create and configure your documentation project.
5. Generate your documentation and export it to the publication format of your choice (HTML, PDF, Confluence markdown).

INSTALLATION

3 Installing the ARCAD Observer Eclipse plug-in

ARCAD Observer is an Eclipse plug-in that can be installed on RDi or any compatible Eclipse IDE. It is installed via the ARCAD RDi plug-ins standard installation package.

3.1 System Requirements

1. ARCAD Server ≥24.0
2. RDi v9.6 or v9.8 or Eclipse IDE ≥v4.2
3. IBM i ≥7.1

> ⓘ **Note**
> To install the plug-in on RDi v9.8, please note that you must install the *RDi 9.8 dependencies* bundle with the plug-in.

3.2 Prerequisites

In order to use the tools provided in the ARCAD Observer perspective, you must have active licenses for the ARCAD Server, ARCAD Observer and the ARCAD Repository.

3.3 Installing

Follow the subsequent steps to install the ARCAD Observer plug-in(s).

Step 1 Download the installation package to your Local Site Repository. ARCAD delivers Eclipse plug-in packages in a .zip file. Save this .zip to your local environment. This location is the local site repository.

> ⓘ **Note**
> The PC on which your Eclipse tool is installed must be able to access this location. It is recommended to save the installation files on the same PC or a shared network that the PC is connected to. There is no need to unzip this file after you save it to your local environment.
>
> After installation, it is recommend that you keep this location. Eclipse will continue to recognize it, so when updates are available, you can copy them into this folder.

Step 2 From the **Help** menu of the Eclipse IDE, select **Install New Software**.

> ⚠️ **Important!**
> You must be logged in as an administrator.

Step 3 From the **Install** dialog, click the **Add** button.

Step 4 In the **Add Repository** dialog, enter a recognizable name, then click **Archive...** This creates a link to the Local Site Repository that you can come back to and that will be used for automatic updates.

Step 5 Navigate to and select the ARCAD plug-ins installation .zip file that you saved to your local environment. Click **OK**.

> **Result** The **Location** field in the **Add Repository** dialog is set to the .zip file's path. You can add new versions to this path later to update.

Click **OK**.

Step 6 Select the plug-in(s) to install.

Click **Next >** to continue.

Step 7 Review the selected features in the **Install Details** dialog.

Click **Next >** to continue.

Step 8 Review and accept the license agreement

The **License Agreement** page presents the ARCAD license agreement for you to review. Please read it carefully.

When you have reviewed the agreement, select **I accept the terms of the license agreements**.

Step 9 Review and confirm the trusted certificates.

Click **OK**.

Step 10 Click the **Finish** button to launch the installation.

Step 11 Restart the workbench. Click **Yes** to restart.

Result ARCAD Observer is installed and available for use.

3.4 Updating

Updating ARCAD plug-ins is similar to the original installation process. The prerequisites are the same and you must already have a previous version of the plug-in installed.

> ℹ️ **Note**
> When there is a major upgrade (as opposed to a simple cumulative patch) or a name change in one of the ARCAD plug-ins, then it is recommended to uninstall the current plug-in and restart your platform before installing the new plug-ins.
>
> Upgrading ARCAD plug-ins only works correctly when the names of the installation files have not changed.

> ℹ **Note**
> If you try to re-install plug-ins instead of search for updates, as long as the correct .zip archive is selected, the plug-ins will update instead of reinstall. However, it is suggested to follow the following procedure to update plug-ins.

Follow the subsequent steps to update the ARCAD Observer plug-in(s).

Step 1 Save the new version of the installation *.zip* in your Local Site repository (created when you first installed the plug-ins).

Step 2 From the **Help** menu of the Eclipse IDE, select **Check for Updates**.

> ⚠ **Important!**
> You must be logged in as an administrator.

Step 3 The **Available Updates** dialog displays all of the plug-ins found in your update location that are out of date. Select the plug-in(s) to update.

Click **Next >** to continue.

Step 4 Review and confirm the updates.

Click **Next >** to continue.

Step 5 Review and accept the license agreement

The **License Agreement** page presents the ARCAD license agreement for you to review. Please read it carefully.

When you have reviewed the agreement, select **I accept the terms of the license agreements**.

Step 6 Review and confirm the trusted certificates.

Click **OK**.

Step 7 Click the **Finish** button to launch the installation.

Step 8 Restart the workbench. Click **Yes** to restart.

Result ARCAD Observer is installed and available for use.

> ℹ **Note**
> Once the update is completed, the IBM i should be updated as well to avoid version compatibility issues. Each user must upgrade their plug-ins, however the upgrade to the server is only performed once.

3.5 Uninstalling

When there is a major upgrade (as opposed to a simple cumulative patch) or a name change in one of the ARCAD plug-ins, then it is recommended to uninstall the current plug-in(s) and restart your platform before installing the new plug-in(s).

Follow the subsequent steps to uninstall the ARCAD Observer plug-in(s).

Step 1   From the **Help** menu of the Eclipse IDE, select **Installation Details**.

> ⚠ **Important!**
>   You must be logged in as an administrator.

Step 2   Click the **What is already installed** link in the lower right-hand corner of the **Install** dialog.
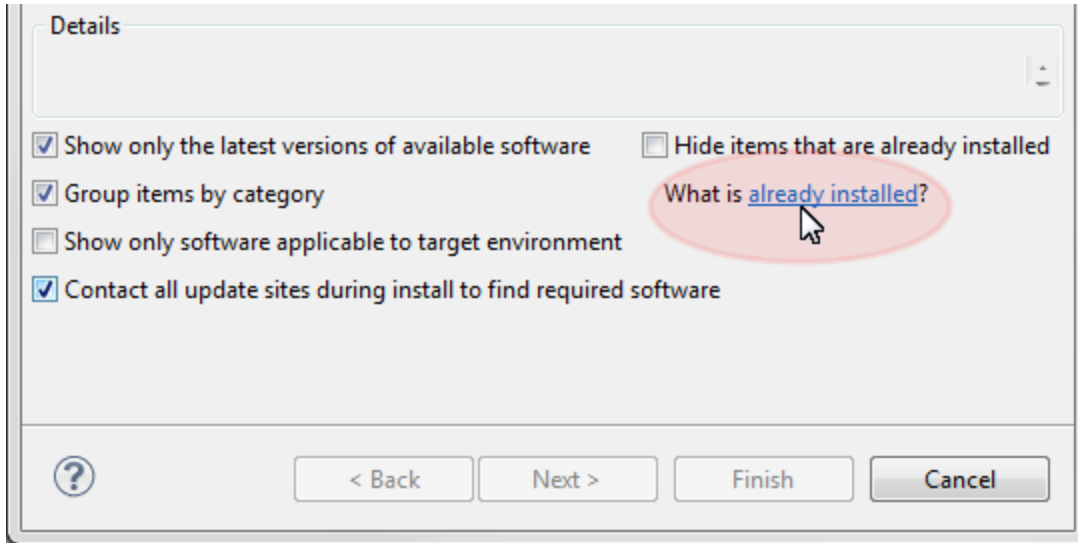
Step 3   Select the plug-in(s) to remove and click the **Uninstall** button.

Step 4   Click **Finish** to confirm.

Step 5   Restart the workbench.

Result   The selected plug-in(s) is/are uninstalled.

GETTING STARTED

4 Overview of ARCAD Observer

ARCAD Observer is an Eclipse plug-in available for RDi and RTC, and pre-packaged in the ARCAD Client RCP. No matter what Eclipse platform you use, the plug-in provides access to the ARCAD Observer perspective.
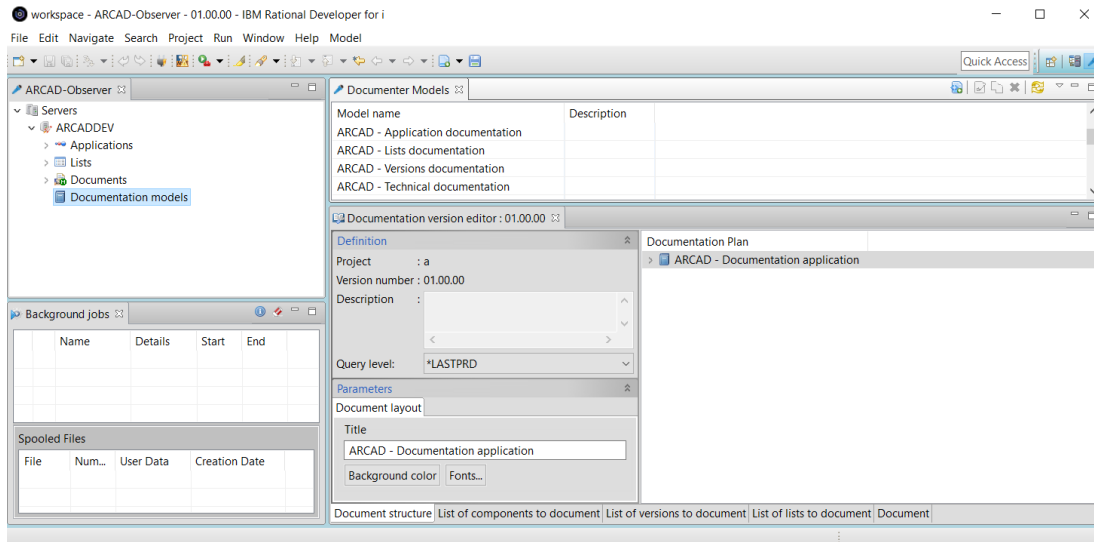
Figure 2: The ARCAD Observer perspective in RDi

4.1 About perspectives

A perspective defines the initial set and layout of views in the workbench window. Within the window, each perspective shares the same set of editors. Each perspective provides a set of capabilities aimed at accomplishing a specific type of task or working with specific types of resources. For example, the JavaTM perspective combines views that you would commonly use while editing Java source files, while the Debug perspective contains views that you would use while debugging a program. Perspectives contain views and editors and control what appears in certain menus and tool bars.

Views in ARCAD Observer are context sensitive. To see a view that is not currently active, open it by accessing the **Show View** dialog (*Window > Show view > Other*) and selecting the view. Re-position views by dragging and dropping them to the desired location in the studio. Click an edge and drag to re-size a view. Double-click on any view's toolbar to maximize it. This is helpful when a view contains a lot of information; maximizing the view makes it fill the entire window. Double-click again to minimize the view.

To re-set the perspective to the default layout, click *Window > Reset Perspective*.

4.2 The ARCAD Observer perspective

The ARCAD Observer perspective consists of these panes: The navigation pane on the left, and the working and editing panes on the right. The views in the perspective change depending on what you are doing. Related views may overlap each other in tabbed layers. In general, the navigation pane always remains, while

the views in the working and editing panes will change depending on what you've chosen in the navigation pane.

The views in the ARCAD Observer perspective are context sensitive. If you are creating a document, one set of views is active; if you edit a diagram, another set of views is active.

4.2.1 The ARCAD Observer navigator

The ✏ **ARCAD Observer** navigator displays the applications, lists, and documents (diagrams) for the connected 🐾 ARCAD Server. From the navigator, there is a direct link to the documentation models view.
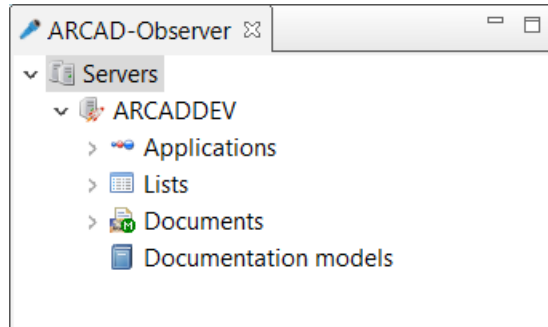
Figure 3: The ARCAD Observer navigator

4.2.2 The applications

For an application, you find the actual application **Repository**, where the elements to documents are listed, and the **Diagrams** (**Documents**) and **Documentation** projects for the application.
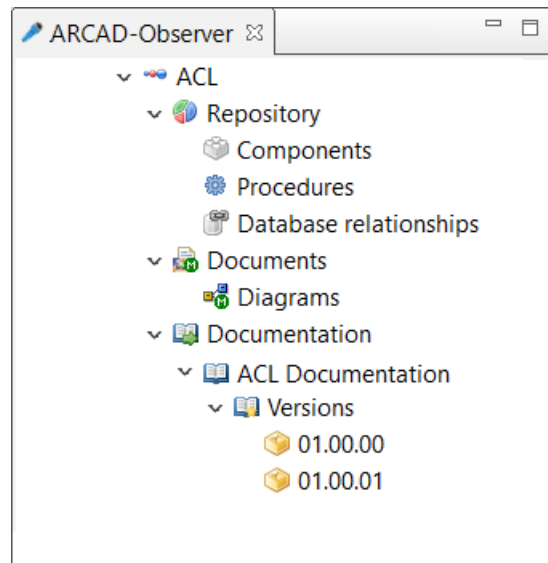
Figure 4: Applications in the ARCAD Observer navigator

Application repository

From the repository, you can access the following views:

- Component repository
- ILE Procedure repository
- Database relationships
- Field repository (from the 🪩 ARCAD Observer DB perspective)

Application documents

The Diagrams view is accessed from the Documents node.

Application documentation

All the versions of all the documentation projects for the application are accessed from the Documentation node.

4.2.3 The Lists

The **Lists** displays all the ARCAD lists available in the current server. This feature is also called ARCAD List Manager and is available for several other ARCAD products.

4.2.4 The Diagrams editor

The **Diagram** view displays the diagram management tools and a preview of the diagram.

4.2.5 The Documentation version editor

The **Documentation version editor** displays the documentation structure editor, the component management views, the versions, the lists to document, and a preview of the final documentation.

4.2.6 The Documenter Models view

The **Documenter Models** view displays the list of all the documentation models available. ARCAD Observer provides a set of documentation models by default: ARCAD - *name of the model.* All documentation models can be edited to suit your needs.

4.3 Global search

The global search option allows you to go search through all the available components, regardless of their nature and the application they are in.

4.3.1 Accessing the global search window...

... from the global search view

Step 1   In the Window tab in RDi, click **Show view** and then **Other....**

The Show view dialog opens, with a list of the installed software.

Step 2   Open the ARCAD **Software** folder and select 📁 **Global Search**, then click on **OK** to open the Global Search view. Double clicking on 📁 **Global Search** also opens the Global Search view.

Result   The 📂 Global search view opens.

> ℹ️ **Note**
>   The search box is empty and you can search for any keyword.

… from an ARCAD Observer component

Step 1   In the LPEX editor, right-click on a selected field.

Step 2   Hover the **ARCAD-Observer** option, then click on **Global search**.

Result   The 📂 Global search view opens.

> ℹ️ **Note**
>   The application box and the search box are already filled with the
>   information concerning the component you opened the **Global search** view
>   from. The search results panel displays more information about the
>   **Description**, **Qualifier** and **Nature** of said component.

4.3.2 Using the global search

The Global Search view contains two panels, the search filters on the left and the search results on the right. Search filters can be set to refine the search results, but you can use a wildcard character (*****) to retrieve all possibly relevant results.

Criterion	Description
Query level	Sets a query level for the global search. The available query levels are the following: • *LASTPRD: the level used is the one of the application's last transfer to production. • Version number: the specified level is used. Click on the ⬚ browse button to select a version number from the existing ones. > ⚠️ **Important!** > This parameter is mandatory.

Table 4: Additional Search Criteria section parameters

Criterion	Description
Nature	Sets the nature of the elements searched for, if they are known. Tick the corresponding nature among the ones available. The available natures are the following: • Component • Procedure • Field • Literal • Used objects • Usage • Message id • Source
Object type	Sets the object type for the **Object repository** or for the **Object XRef**. The **Component** case has to be checked in the **Nature** option above for this filter to be used.
Source type	Sets the source type for the **Object/Source repository** or the **Included source XRef**. The **Component** case has to be checked in the **Nature** option above for this filter to be used.
Source file	Sets the source file for the **Object repository**. The **Component** case has to be checked in the **Nature** option above for this filter to be used.
Module	Sets the module name for **Procedure repository** or the **Procedure Xref**. The **Procedure** case has to be checked in the **Nature** option above for this filter to be used.
Text	Sets the **Object** or **Source** text. The **Component** case has to be checked in the **Nature** option above for this filter to be used.

Table 4: Additional Search Criteria section parameters

Once the search filters are set and the search launched, the search results appear on the right panel.

You can right-click on any results to display the ARCAD Observer menu, and for example launch an analysis directly from the **Global search** view. It is also possible to display the source file or display the calling chains of a program.

5 Defining the server connection

Chapter Summary

ARCAD Observer can be accessed from RDi or from ARCAD Client. A connection to the ARCAD Server is required.

5.1 Accessing ARCAD Observer perspective

Follow the subsequent steps to to access the ARCAD Observer perspective in RDi or ARCAD Client.

Step 1   Click the ⊞ **Open Perspective** icon in the toolbar or open the **Window** menu then select **Open Perspective** > **Other...**.

Step 2   Select 🖊 ARCAD Observer in the dialog, then click **OK**.

Result   The 🖊 ARCAD Observer perspective is displayed.

5.2 Connecting to the ARCAD Server

When launching ARCAD Observer, the connection to the ARCAD Server must be defined.

The connection dialog opens automatically when opening the ARCAD Observer perspective.

Step 1   Define the following fields in the **ARCAD Connection** dialog.

Server

> Enter the name or IP address of the IBM i partition hosting the server.

Login/Password

> Enter a valid user login and password. This user must be declared on the server and will be the user that will execute any operations on the IBM i. The password should be encoded in Base64.

Instance

> Enter the ID of the instance to reach. If you are using ARCAD the instance is **AD** by default.

Language

> Select the language to use on the server (FRA or ENG).

CCSID

> Enter a valid CCSID to connect to the server.

Use Secure Socket Layer (SSL)

Tick this checkbox if the server requires a secured TLS/SSL connection.For more information about ARCAD secure connections, refer to the *ARCAD SSL Configuration Guide*.

Save connection parameters

[Optional] Tick this checkbox to remember the password for the current user.

Bypass Arcad Connection Dialog

[Optional] The last connection information is automatically saved. Tick this checkbox to hide the connection dialog and always use this connection information.

Step 2 Click **OK**.

Result The connection to the ARCAD Server is defined.

CONFIGURATION

6 Preferences

Perspective	🖊 ARCAD Observer
Access	Window > Preferences

Chapter Summary

The preferences for ARCAD Observer are managed in the **Preferences** window. To access this view in RDi, click **Window** in the top menu and select **Preferences**.

6.1 Confluence documentation upload
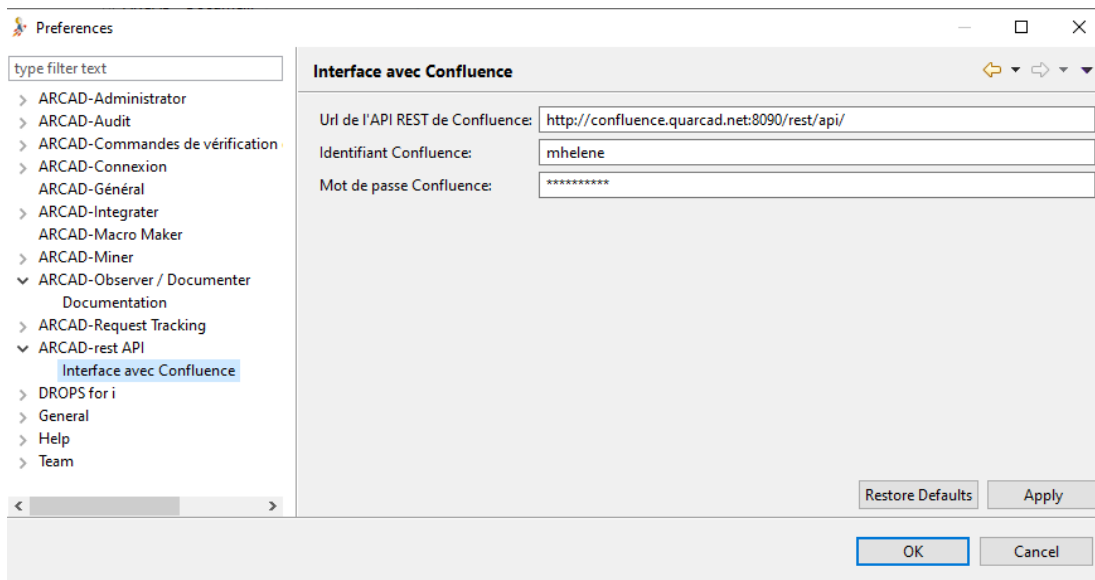
Figure 5: Confluence Documenter preferences

Confluence REST API URL

Sets the REST API URL leading to the Confluence space where the documentation are exported to.

Confluence user name

Sets the user name of the Confluence account.

Confluence password

Sets the password corresponding to the user name, to access the Confluence account.

Click **Apply** to validate the changes made, or click **Restore Default** to return to the default value of the preference.

6.2 Documenter preferences
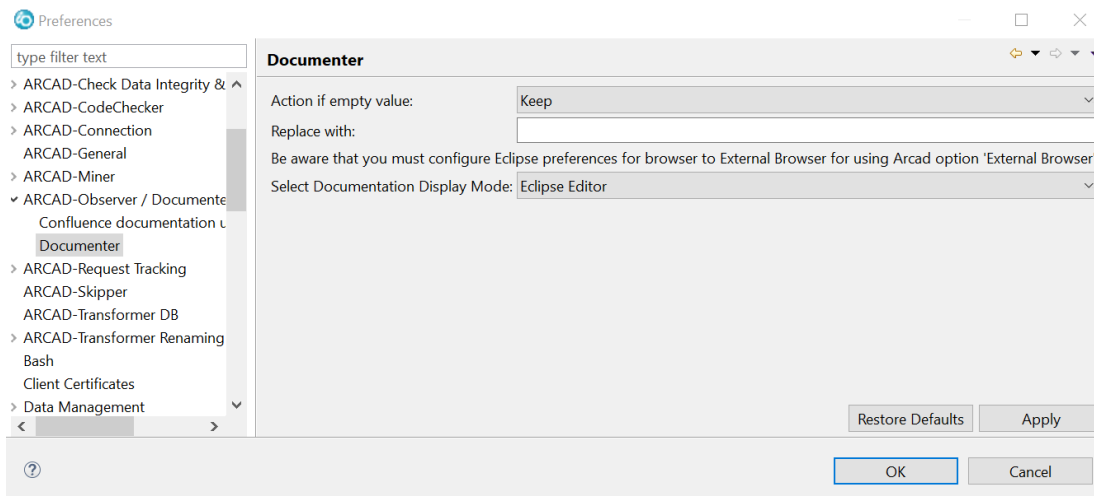
Figure 6: Documenter preferences

Action if empty value

Keep: when selected, all the sections are displayed in the documentation, even if they are empty.

Hide: when selected, empty sections are hidden in the documentation.

Replace: when selected, empty sections are displayed in the documentation, but the content of the section is replaced by custom text set in the **Replace with** option.

Replace with

Enter text if **Replace** is chosen for the **Action if empty value** option.

Select Documentation Display Mode

Select what tool you wish to use to display the documentation preview. Choose the **Eclipse editor** in the drop-down list to display the documentation preview in RDi or the **External browser** to display it in the default navigator set on your machine.

> ⓘ **Note**
>
> To use the External Browser in the ARCAD Observer Documenter feature, the Eclipse **Web Browser** preferences must be correctly configured.

> ℹ️ In the *Window > Preferences > General > Web Browser* preferences dialog, the **Use external web browser** option must be activated.

Click **Apply** to validate the changes made, or click **Restore Default** to return to the default value of the preference.

6.3 Component preferences

Figure 7: Component preferences

Background color, Lines colors, Font color

Define the colors of the **Components** shapes of the diagrams. Click to select a color from the color selection dialog, then click **OK**.

Font

Define the **Font** and font size of the **Components** shapes of the diagrams.

Click **Apply** to validate the changes made, or click **Restore Default** to return to the default value of the preference.

6.4 Connection preferences

Figure 8: Connection preferences

Connections color

Define the default color of the connectors in the diagrams. Click to select a color from the color selection dialog, then click **OK**.

Selected connections color

Define the color of the connectors when they are selected in the diagrams. Click to select a color from the color selection dialog, then click **OK**.

Click **Apply** to validate the changes made, or click **Restore Default** to return to the default value of the preference.

6.5 Diagram options preferences
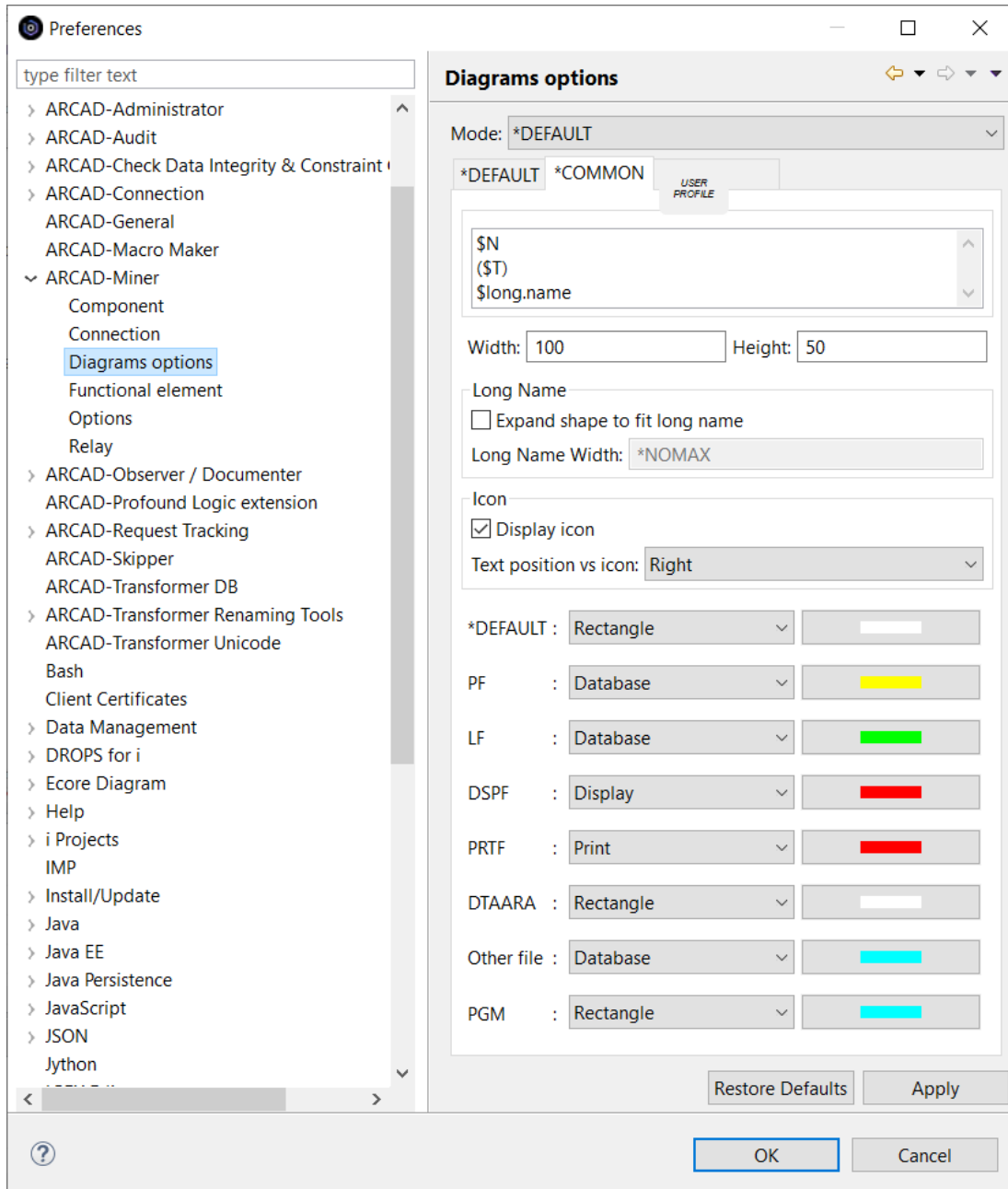
Figure 9: Diagram options preferences

The diagram option preference dialog is used to set default diagram display option depending on the profile.

Mode

Select the display mode to use:

*DEFAULT: the default mode is using the standard display options set in ARCAD Observer. These values are not editable.

*COMMON: the common mode is using the custom display options set for all the profiles.

USER PROFILE: the current user mode is using the custom display options set for an individual user profile.

Content

The information displayed in the shapes of the diagram: by default, the variable used are *$N, ($T), ($long.name)*.

Width & Height

The width and height of the diagram shapes in px. The default values are 96 x 96 px.

Long Name

Tick **Expand shape to fit long name** to fit the shapes size to the text so that the text is not truncated.

Long Name Width: set the max. width of the long name in the diagram. The default value is ***NOMAX**.

Icon

Display icon: Tick to display the icon corresponding to the type of component.

Text position vs icon: The text is positioned on the **Left**, **Right**, **Top** or **Bottom** of the icon.

Shapes and colors

Define the default shape and color associated to the component types (source type or object type).

	Rectangle: by default, this shape is applied to *PGM, *CMD, *MODULE, *SRVPGM and other components (DTAARA).
	Database: by default, this shape is applied to *FILE (PF) and *FILE (LF).
	Display: by default, this shape is applied to *FILE (DSPF).
	Print: by default, this shape is applied to *FILE (PRTF).

Table 5: Diagram options preferences

Click **Apply** to validate the changes made, or click **Restore Default** to return to the default value of the preference.

6.6 Functional element preferences
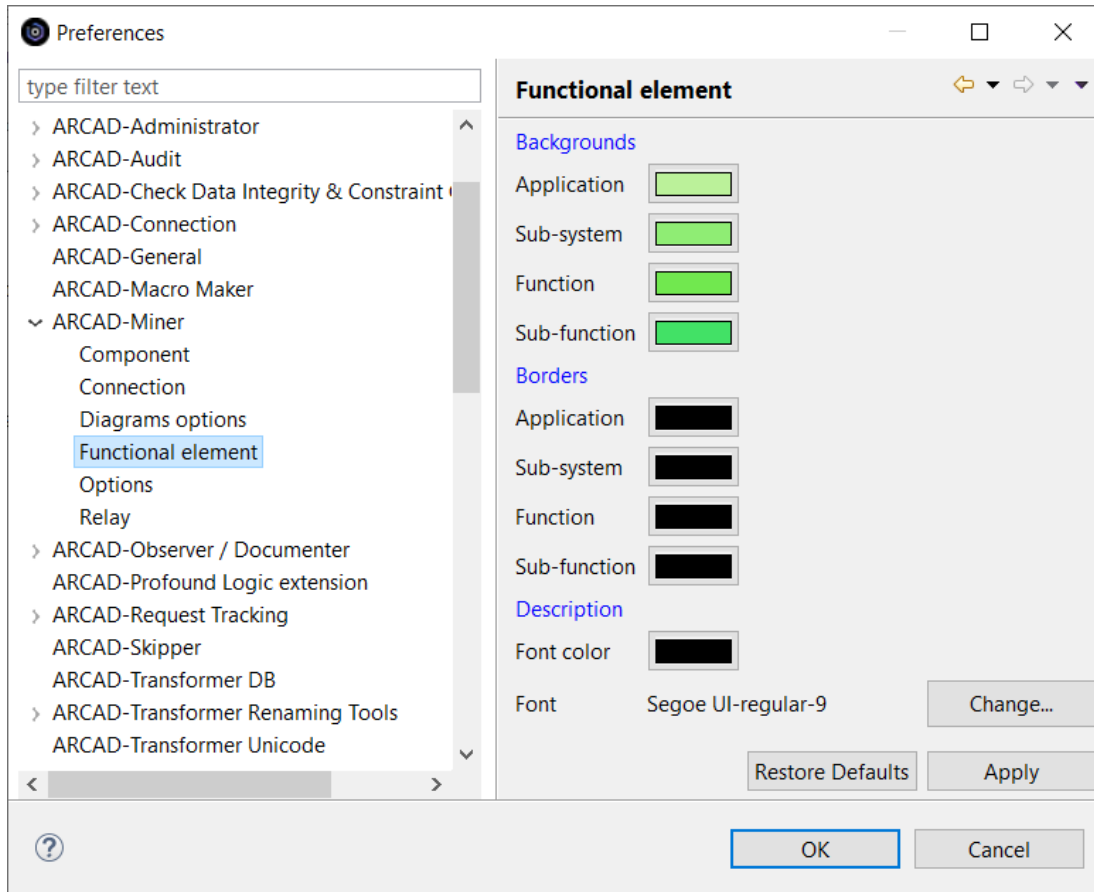
Figure 10: Functional element preferences

Backgrounds

Define the background color of the diagram shapes for **Application**, **Sub-system**, **Functions** and **Sub-function**.

Borders

Define the borders color of the diagram shapes for **Application**, **Sub-system**, **Functions** and **Sub-function**.

Description

Define the **Font**, font size and **Font color** of the descriptions. Click to select a color from the color selection dialog, then click **OK**.

Click **Apply** to validate the changes made, or click **Restore Default** to return to the default value of the preference.

6.7 Options preferences
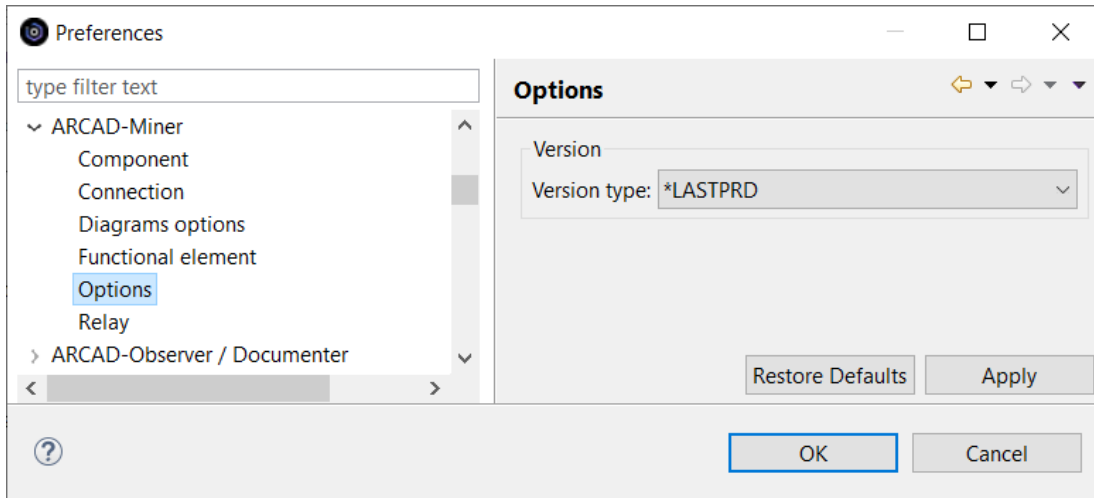
Figure 11: Options preferences

Version type

   *LASTPRD: The level used will be the one of the application's last transfer to production.

   *LAST: The level used will be the one of the last update of the cross-references for each component.

Click **Apply** to validate the changes made, or click **Restore Default** to return to the default value of the preference.

6.8 Relay preferences
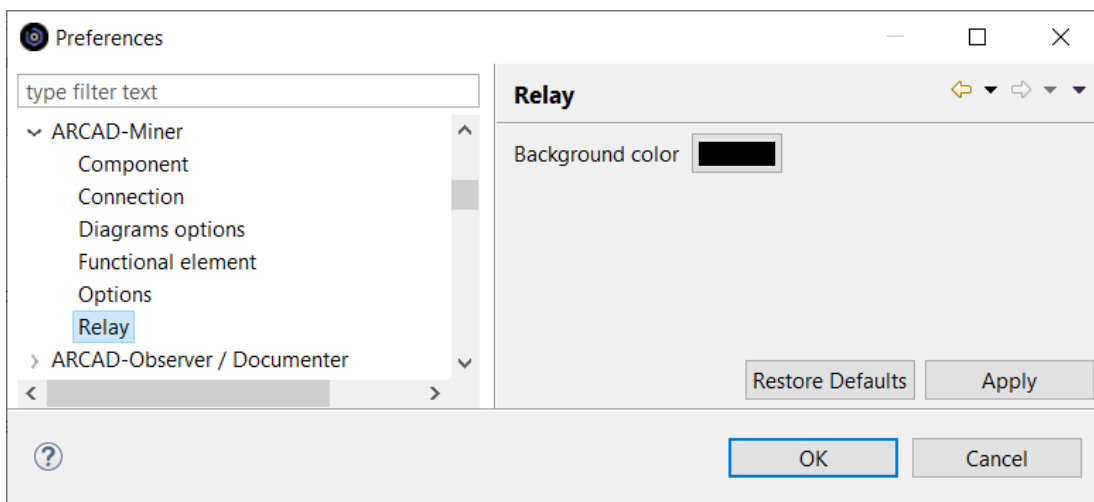
Figure 12: Relay preferences

Background color

> Define the background color of the relays. Click to select a color from the color selection dialog, then click **OK**.

Click **Apply** to validate the changes made, or click **Restore Default** to return to the default value of the preference.

6.9 Calling chain parameters preferences
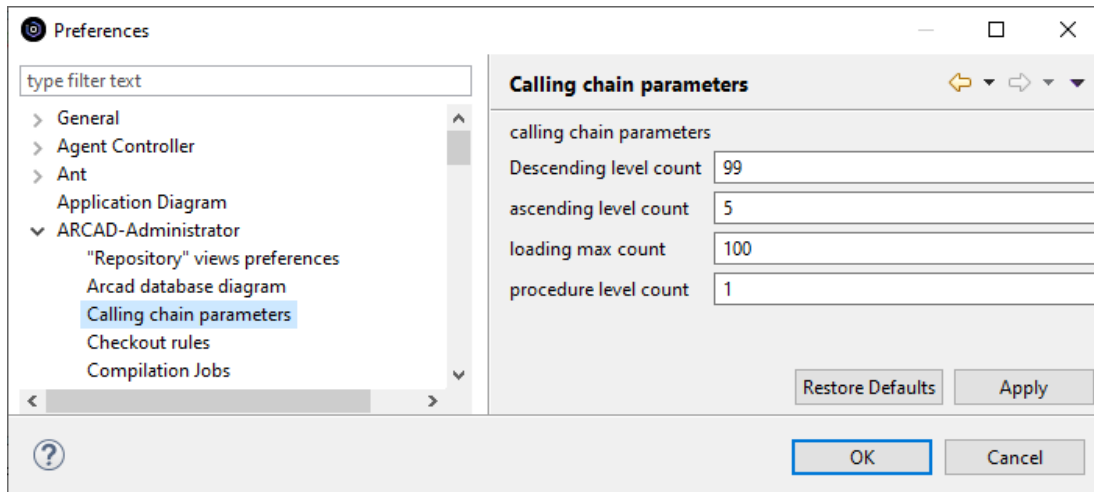
Figure 13: Call chains parameters preferences

Define the default call chain parameters.

Descending level count

> Define the max. level count for descending call chains analysis.

Ascending level count

> Define the max. level count for ascending call chains analysis.

Loading max. count

> Define the loading max. count for call chains analysis.

Procedure level count

> Define the max. level count for procedures.

Click **Apply** to validate the changes made, or click **Restore Default** to return to the default value of the preference.
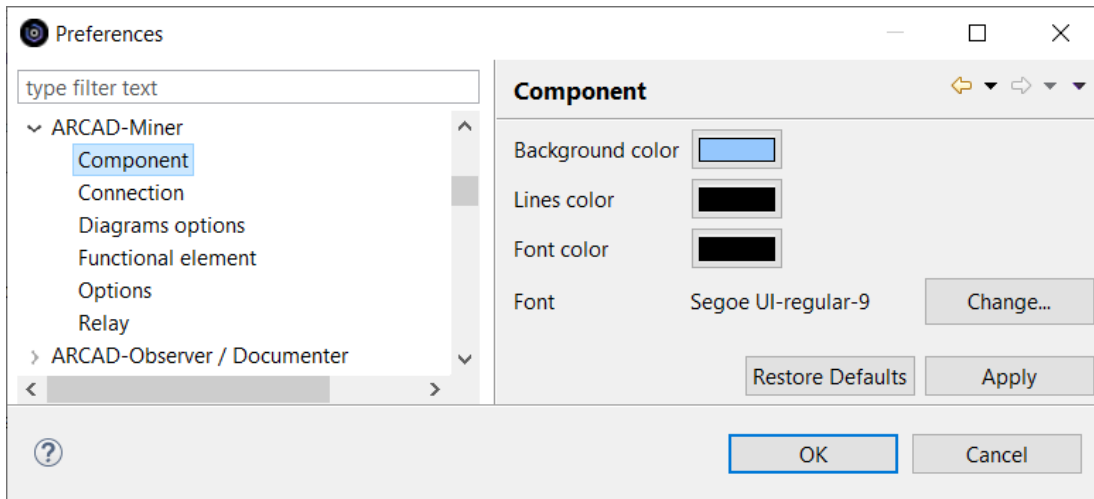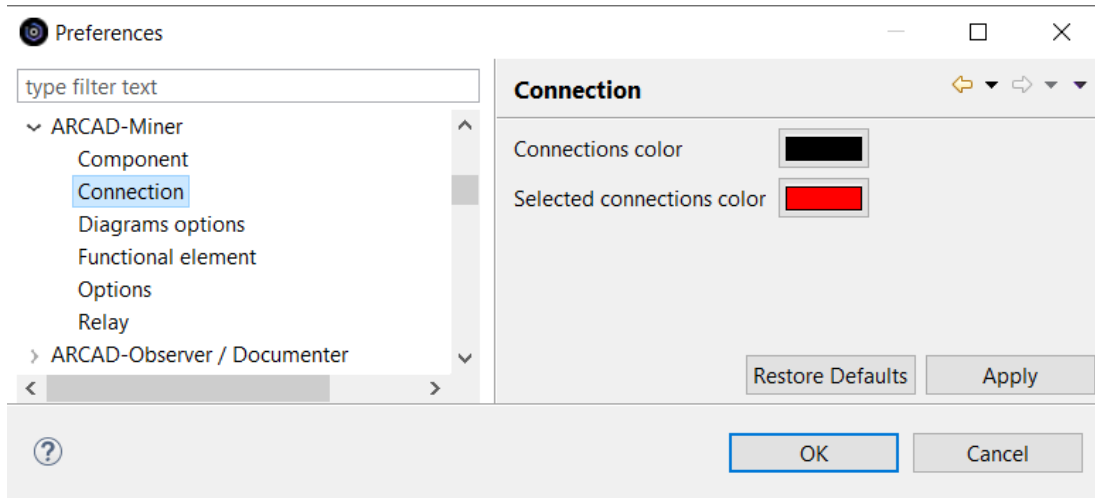
> 🔖 *Reference*
>
> > For more information about call chain analysis, refer to Calling chains on page 44.

6.10 Cross references preferences

Tick the **Synchronize information panel automatically** box to ensure that the information panel displays dynamically the information about the component selected in the **Cross-reference** analysis view.

Request options
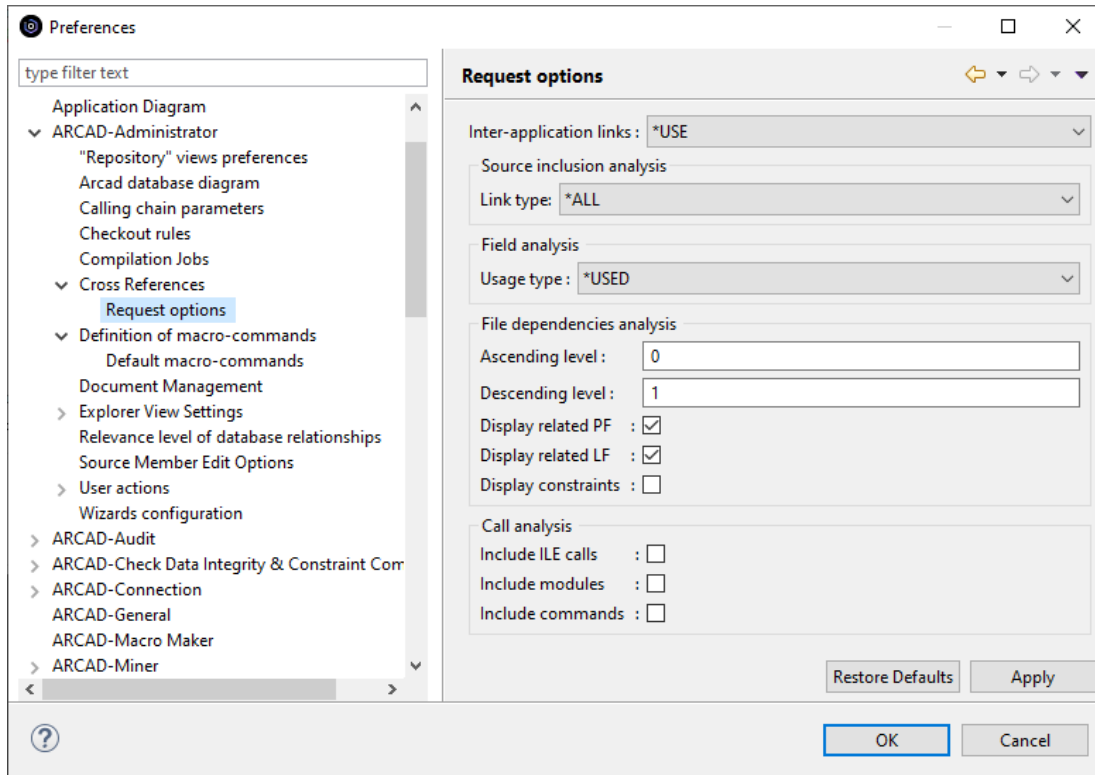
Figure 14: Request options preferences

Select the default **Request options** for the cross-references analysis.

> **Reference**
> For more information about cross-reference analysis, refer to Cross-references on page 51.

Inter-application links

Specifies whether or not to use inter-application links for cross-reference usage. The list of applications linked to the current application is obtained by examining the links defined in the application's descriptive parameters (found using the command ADSPAPPINT). The possible values are:

- ***NONE**: The inter-application links are not used. Only the components present in the interrogating application are known.
- ***USE**: The inter-application links are used to simulate the real usage context for components. The link is made towards the applications used or with a reciprocal link. In the case of a component homonym, only the first one found is taken.

- ***ALL**: The inter-application links are used to determine all the applications concerned by a component. The link is made in all directions (applications used or using the current application or reciprocal link). In the case of a component homonym, all the applications of this component are displayed, in the order defined in the inter-application links.

Link type

*SIMILAR displays links resulting from component copies.

*INCLUDE displays COPY clause type links

*L4GL3G displays the 3rd and 4th generation languages links types.

*ALL Displays all link types for the selected component.

Usage type

*USED Only downloads the explicitly used fields from among the referenced fields.

*UPD Only downloads the updated fields from among the referenced fields.

*IMPL Only downloads the explicitly and implicitly used fields from among the referenced fields.

*ALL Downloads all referenced fields.

Sort

*ALPHA to display Alpha types fields first.

*LINE to display the result in the lines' order.

Ascending level & Descending level

Enter a value to set the levels of ascending and descending file levels you want to display.

Display related PF & Display related LF

Click to set to **True** or **False** the related physical and logical files to display.

Display constraints

Click to set to **True** or **False** the constraints to display.

Include commands

If this option is checked, program cross-references will include commands.

Include ILE Links

If this option is checked, components associated through an ILE link are added to the result list.

Include Module Links

If you check this option, program cross-references will include calling and called modules. This option is only valid for cross-reference calculations starting from a module.

Click **Apply** to validate the changes made, or click **Restore Default** to return to the default value of the preference.

ANALYSIS

# Introduction to application analysis

This section will guide you through the application analysis features of ARCAD Observer.

- Calling chains on page 44
- Cross-references on page 51
- Database relationships on page 69
- Work lists on page 75

7 Calling chains

Perspective	✏️ ARCAD Observer
Access	🖳 *Server >* 🌐 Applications > 🌐 *Application Name >* 🌐 Repository

Chapter Summary

ARCAD Observer allows you to navigate all the dependencies of programs and other components in your application.

7.1 About call chain analysis

For a given program:

Calling

>   The programs that call this program.

Called

>   The programs that are called by this program.

Descending

>   The program's called program or the descendants of one of its called programs.

Ascending

>   The program's calling program or the ascendants of one of its calling programs.

Final ascending

>   A program with no calling program.

Final descending

>   A program with no called program.

Level

>   The generation number included between this program and one of its ascendants (ascendance level) or one of its descendants (descent level).

Calling chain

>   A multi-level hierarchical view of calls between programs.

Branch

> A group made up of a component and all of its descendants.

7.2 Launching the call chain analysis on programs

Follow the subsequent steps to launch the call chain analysis.

Step 1   To display a program's call chain, right-click on the item in the Components view and select Calling chain > Programs > …

Step 2   Select the types of calling chain.

Ascending Chains

> This chain shows all calling programs for a given program, up to the final ascendants or up to a defined ascendance level.

Descending Chains

> This chain shows all called programs for a given program, down to the final descendants or down to a defined descent level.

Ascending / Descending Chains

> This chain shows the ascending or descending chain for a given program.

Result   The call chain analysis result view is displayed

> ⓘ **Note**
>
> > Call chains can be produced using several points of entry. To do this, select a number of components before launching the calling chain analysis.

7.3 Launching call chain analysis on other components

Call chain analysis is possible on files. Some files contain triggers that launch programs execution. Therefore you can do call chain analysis on those files. In that case, the procedure is the same as call chain analysis on programs.
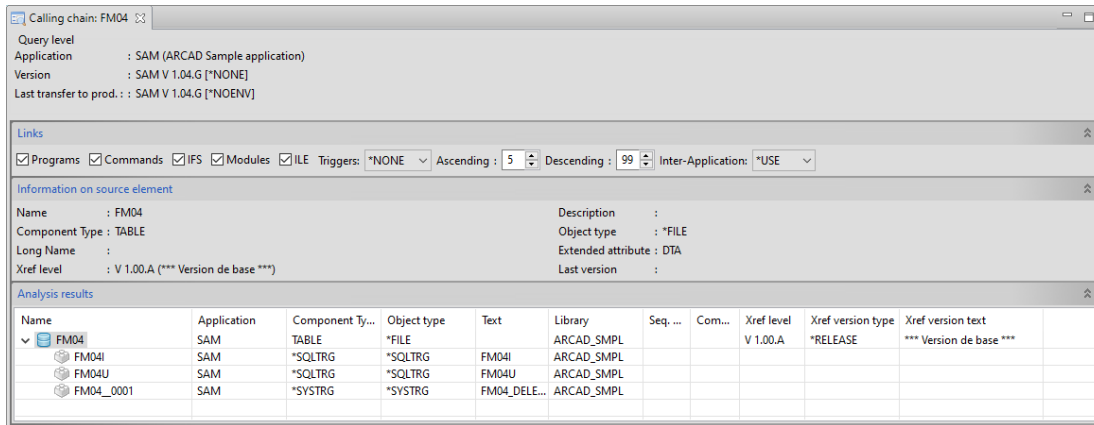
Figure 15: Call chain analysis on files

7.4 Call chain analysis results
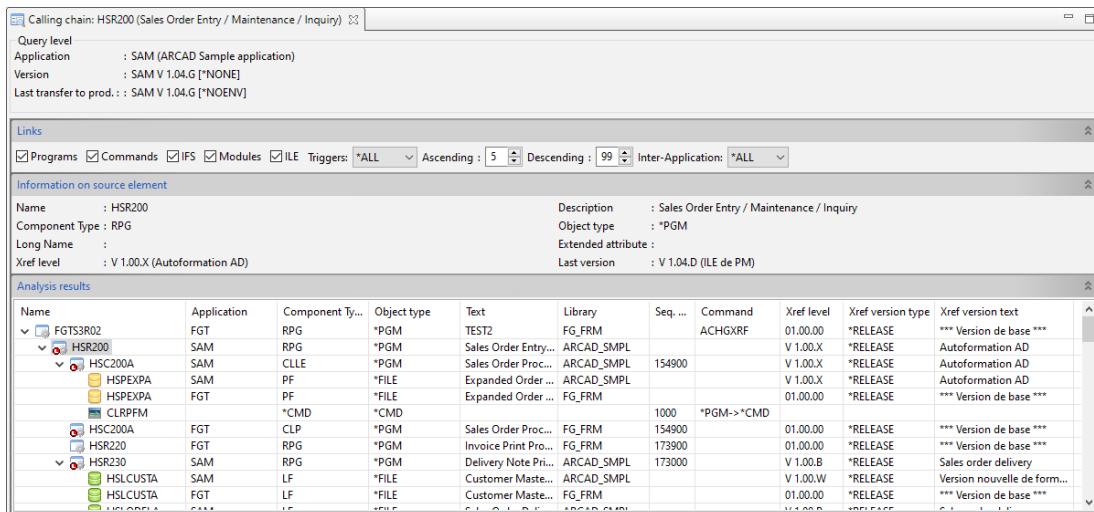
Figure 16: The call chain analysis results view

The **Query level** information panel contains information on the analyzed element's owner application and on the query level.

The **Information on source element** panel contains information about the selected component.

In the call chain **Analysis Results** view, the call chain is folded by default. Use the Expand options to display the components of the call chain. To expend and the branches of the analysis results' tree, right-click on the node in the **Analysis result** section and select Expand in the contextual menu. there are three options:

- Select **Expand...** to manually define the levels to expand. In the dialogue, enter the number of levels and click **OK**.
- Select **Expand all levels** to unfold all the are there nodes of the analysis results tree.
- Select **Expand up to default level** to unfold the notes of the analysis result tree to the level of the analyzed component.

Alternatively, select a node in the **Analysis result** section and click the ⚃ Expand or ⚃ Collapse icons in the toolbar to expand or collapse all the node below the selected node.

Each node in the call chain **Analysis Results** view shows an icon (specific to the type of component it represents) and all the details of the component.

> ⓘ **Note**
>> If an analyzed component has a CLP source type, the called elements are shown in their calling order within the program.

Removing a branch from the result view

You may want to focus on a specific branch of the call chain and remove the unnecessary branches from the results view. To remove a branch, right-click on the item in the **Analysis result** section and select **Remove branch**.

> ⓘ **Note**
>> To display the branch again, you need to launch a new analysis of the component.

Analysis results display filters

Links

Tick the corresponding box to display the chosen types of components' links: **Programs**, **Commands**, **IFS**, **Modules**, **ILE**.

Triggers

Indicates if you wish to display calling chains with files and their triggers or FIELDPROC calls.

- ***NONE**: Omits all *File* to *Program* links.
- ***TRG**: Selects all trigger types (I + U + D) or file update types.
- ***ALL**: Selects all *File* to *Program* links.
- ***TRGI**: Only selects *Insert* type triggers (Insert).
- ***TRGU**: Only selects *Update* type triggers (Update)
- ***TRGUD**: Selects *Update* and *Delete* type triggers.
- ***TRGID**: Selects *Insert* and *Delete* trigger types.

Ascending

Use the increment selector to set the number of ascending levels of links to display (value from 0 to 9).

> ⓘ **Note**
>> If you display calls between procedures (and any subroutines), it is considered that there is a new level between each component or procedure (or calling subroutine).

Descending

Use the increment selector to set the number of descending levels of links to display (value from 0 to 99).

> **ⓘ Note**
>
> If you display calls between procedures (and any subroutines), it is considered that there is a new level only when the source component changes.

Inter-application links

Specifies whether or not to use inter-application links for cross-reference usage. The list of applications linked to the current application is obtained by examining the links defined in the application's descriptive parameters (found using the command ADSPAPPINT). The possible values are:

- ***NONE**: The inter-application links are not used. Only the components present in the interrogating application are known.
- ***USE**: The inter-application links are used to simulate the real usage context for components. The link is made towards the applications used or with a reciprocal link. In the case of a component homonym, only the first one found is taken.
- ***ALL**: The inter-application links are used to determine all the applications concerned by a component. The link is made in all directions (applications used or using the current application or reciprocal link). In the case of a component homonym, all the applications of this component are displayed, in the order defined in the inter-application links.

7.5 Searching options in calling chains

It is possible to have access to advanced search options on ascending and descending calling chains.

Step 1   Right-click on a component.

Step 2   Click on **Calling chain** > **Programs** > **Ascending and descending**.

The corresponding calling chain opens in a new tab.

Step 3   In the **Analysis results** tab, right-click anywhere, then click on the 🔎 **Search...** option.

Step 4   Complete the selection criteria dialog.

Columns

Sets the column of the Analysis results where the search has to be made.

Select a column in the drop-down list.

Operator type

Sets the operator to use to compare the value set in the **Search value** option below, with the available results.

Select an operator type in the drop-down list among:

Like	Searches for a value that contains the same value as the one entered in the **Search value** option. With this operator type, the value entered in the **Search value** option below has to be a string.
Not Like	Searches for a value that does not contain the same value as the one entered in the **Search value** option. With this operator type, the value entered in the **Search value** option below has to be a string.
=	Searches for a value that equals to the value entered in the **Search value** option. With this operator type, the value entered in the **Search value** option below has to be an integer.
<	Searches for a value that is less than the value entered in the **Search value** option. With this operator type, the value entered in the **Search value** option below has to be an integer.
>	Searches for a value that is greater than the value entered in the **Search value** option. With this operator type, the value entered in the **Search value** option below has to be an integer.
>=	Searches for a value that is greater than or equal to the value entered in the **Search value** option. With this operator type, the value entered in the **Search value** option below has to be an integer.
<=	Searches for a value that is less than or equal to the value entered in the **Search value** option. With this operator type, the value entered in the **Search value** option below has to be an integer.
<>	Searches for a value that does not correspond exactly to the value entered in the **Search value** option. With this operator type, the value entered in the **Search value** option below has to be an integer.

Search value

Sets the value to search for in the available results. You can enter a complete value, it has to be exact. If you do not know the exact value, you can enter at least one letter and the ***** character.

Step 5 Click **Add** to put additional search criteria. Select a criterion and click **Remove** if you no longer need it.

Step 6 Tick the **Case sensitive** option to refine the search.

Step 7 Click the **Search** button to launch the search. Click the **Cancel** button to close the search dialog.

Result   In case of a component corresponding to the search you launched, this one is highlighted in pink.

8 Cross-references

Perspective	✏️ ARCAD Observer
Access	🗄️ *Server* > 📟 Applications > 📟 *Application Name* > 🌐 Repository

Chapter Summary

ARCAD Observer allows you to navigate all the cross-references in your application and to find detailed information about them. It is possible to select several item before starting a cross-reference analysis. The result of this analysis can be used as a starting point of a new cross reference analysis.

8.1 Calls analysis

Shows the called or calling components for a given component. This request is only available for components that allow calling of another component (program, command, etc.).

Step 1 Select one or several program components in the view then right-click and select **Analyze** > **Calls**.

A 📈 **Cross-references: Calls** view is displayed showing the analysis results.

> ℹ️ **Note**
> The results list first shows called components, then the calling components. The list of called components is sorted by increasing order of the resulting component's call sequences in the source of the analyzed component.

Figure 17: Calls cross-references analysis result view

Step 2 Use optional parameters to analyze the results in the call parameter section, then click on the component to refresh the results.

Inter-application links

Specifies whether or not to use inter-application links for cross-reference usage. The list of applications linked to the current application is obtained by examining the links defined in the application's descriptive parameters (found using the command ADSPAPPINT). The possible values are:

- ***NONE**: The inter-application links are not used. Only the components present in the interrogating application are known.
- ***USE**: The inter-application links are used to simulate the real usage context for components. The link is made towards the applications used or with a reciprocal link. In the case of a component homonym, only the first one found is taken.
- ***ALL**: The inter-application links are used to determine all the applications concerned by a component. The link is made in all directions (applications used or using the current application or reciprocal link). In the case of a component homonym, all the applications of this component are displayed, in the order defined in the inter-application links.

Include commands

If this option is checked, program cross-references will include commands.

Include ILE Links

If this option is checked, components associated through an ILE link are added to the result list.

Include Module Links

If you check this option, program cross-references will include calling and called modules. This option is only valid for cross-reference calculations starting from a module.

The **Direction** column contains the relationship type between the analyzed component and the result component. This type can take the *Called* or *Calling* values in the case of a call; or *Used By* or *Using* values in the case of an ILE link.

8.2 Utilization analysis

Shows the components that use or are used by a given component.

Step 1 Select one or several components in the view then right-click and select **Analyze** > **Utilization**.

A ✖ **Cross-references: Utilization** view is displayed showing the analysis results.

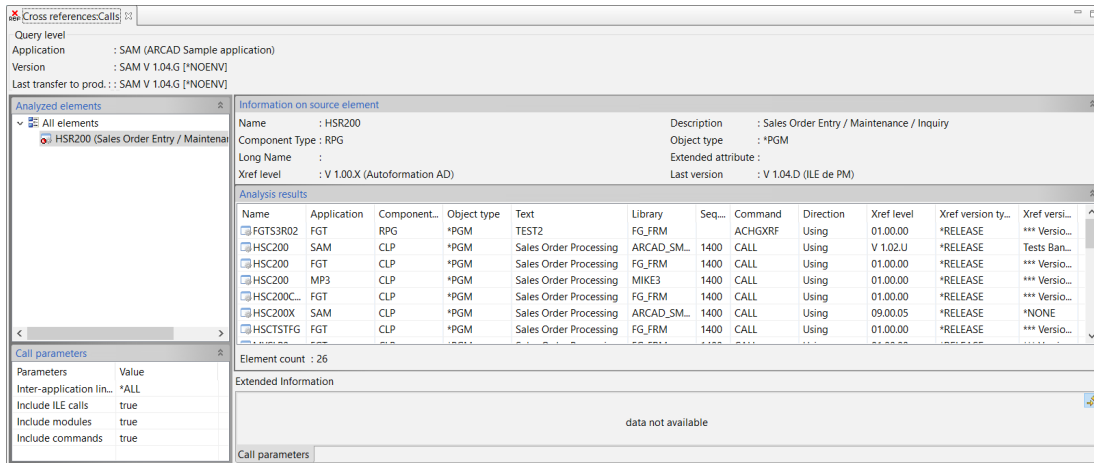*Figure 18: Utilization cross-references analysis result view*

Step 2 Use optional parameters to analyze the results in the call parameter section, then click on the component to refresh the results.

Inter-application links

Specifies whether or not to use inter-application links for cross-reference usage. The list of applications linked to the current application is obtained by examining the links defined in the application's descriptive parameters (found using the command ADSPAPPINT). The possible values are:

- ***NONE**: The inter-application links are not used. Only the components present in the interrogating application are known.
- ***USE**: The inter-application links are used to simulate the real usage context for components. The link is made towards the applications used or with a reciprocal link. In the case of a component homonym, only the first one found is taken.
- ***ALL**: The inter-application links are used to determine all the applications concerned by a component. The link is made in all directions (applications used or using the current application or reciprocal link). In the case of a component homonym, all the applications of this component are displayed, in the order defined in the inter-application links.

The **Usage** column contains a description of how the resulting component is used by the analyzed component. The possible values are: [**Not Specified**, **Input**, **Output**, **Input/Output**, **Update**, **Input/Update**, **Output/Update**, and **Input/Output/Update**].

8.2.1 Displaying access keys

You can consult the keys (or set of keys) used by a program to access a file.

Step 1 Verify that the ⟳ Auto-refresh icon is engaged.

Step 2 Select a program from the analyzed component tree.

Step 3 Select a file from the result list.

Result The keys used by the program to access the file are displayed in the **Extended information** section of the result view.
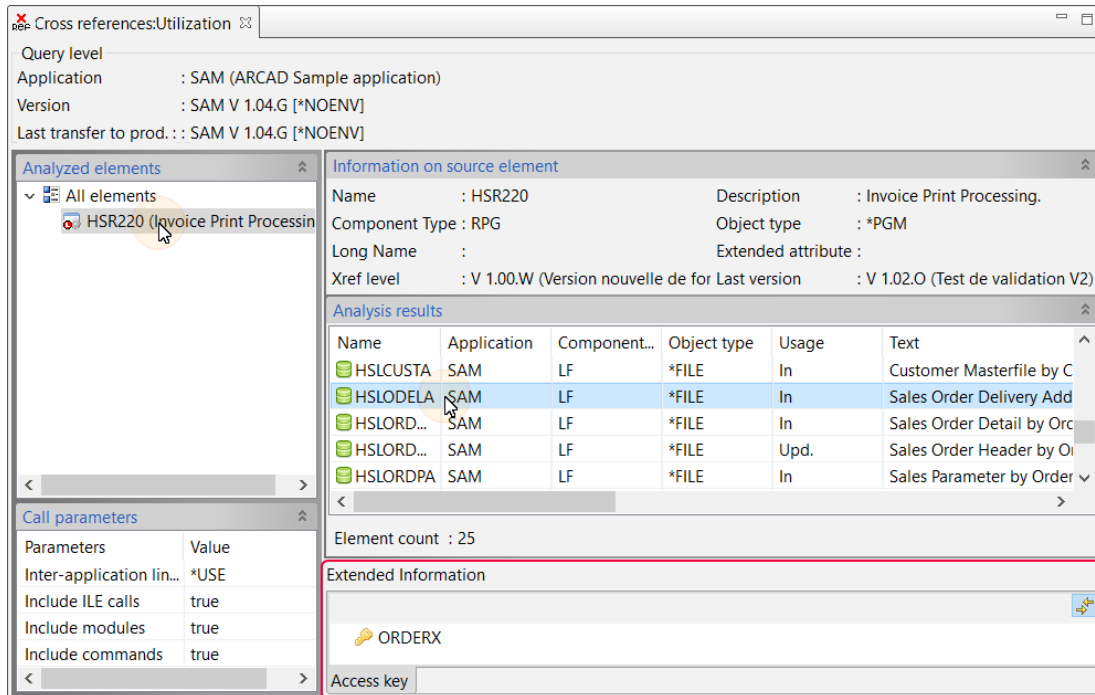
Figure 19: Access keys

8.3 Dependent files analysis

Shows a given file's dependent files. If the selected file is a **physical file**, the dependent files will be the associated logical files or the physical files dependent by format. If the selected file is a **logical file**, the dependent file will be the master physical file.

Step 1 Select one or several components in the view then right-click and select **Analyze** > **Dependent files**.

A 🔀 **Cross-references: Dependent files** view is displayed showing the analysis results.

Figure 20: Dependent files analysis result view

Step 2 Use optional parameters to analyze the results in the call parameter section, then click on the component to refresh the results.

Inter-application links

Specifies whether or not to use inter-application links for cross-reference usage. The list of applications linked to the current application is obtained by examining the links defined in the application's descriptive parameters (found using the command ADSPAPPINT). The possible values are:

- ***NONE**: The inter-application links are not used. Only the components present in the interrogating application are known.
- ***USE**: The inter-application links are used to simulate the real usage context for components. The link is made towards the applications used or with a reciprocal link. In the case of a component homonym, only the first one found is taken.
- ***ALL**: The inter-application links are used to determine all the applications concerned by a component. The link is made in all directions (applications used or using the current application or reciprocal link). In the case of a component homonym, all the applications of this component are displayed, in the order defined in the inter-application links.

Ascending level & Descending level

Enter a value to set the levels of ascending and descending file levels you want to display.

Display related PF & Display related LF

Click to set to **True** or **False** the related physical and logical files to display.

Display constraints

Click to set to **True** or **False** the constraints to display.

8.4 Fields analysis

Shows the list of fields that are declared or (directly or indirectly) manipulated by the selected component.

Step 1 Select one or several components in the view then right-click and select **Analyze** > **Fields**.

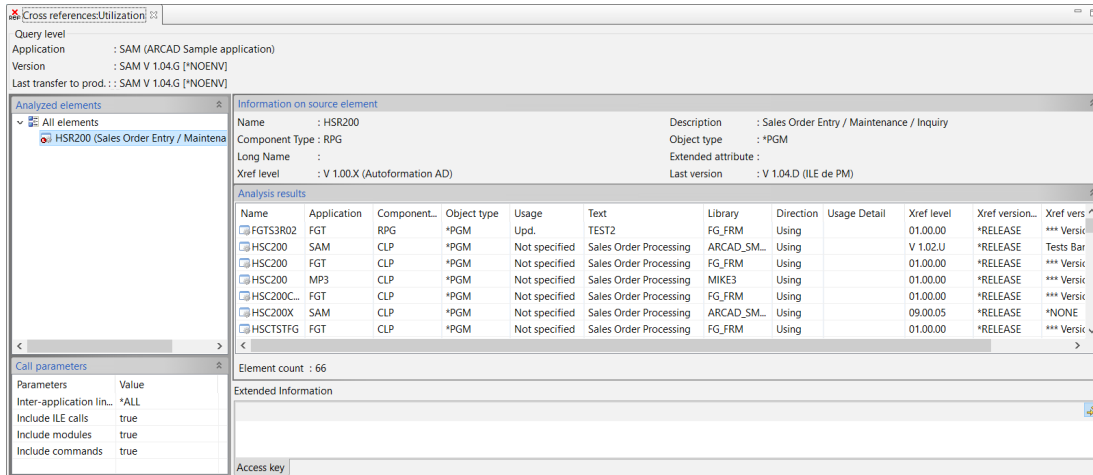A **Cross-references: Fields** view is displayed showing the analysis results.
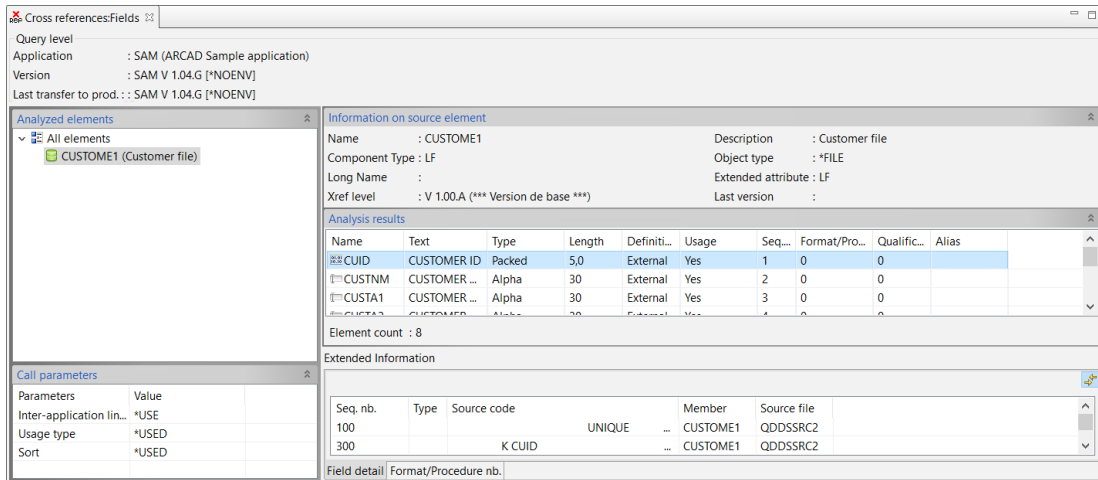
Figure 21: Fields analysis result view

Step 2   Use optional parameters to analyze the results in the call parameter section, then click on the component to refresh the results.

Inter-application links

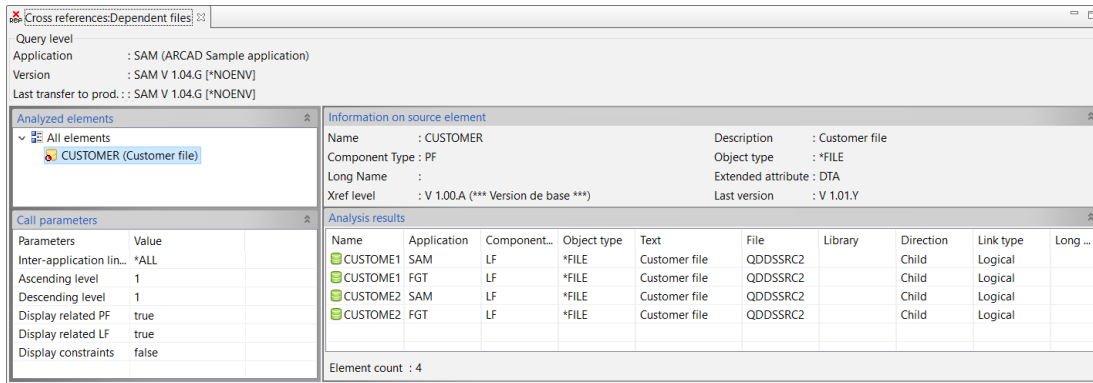Specifies whether or not to use inter-application links for cross-reference usage. The list of applications linked to the current application is obtained by examining the links defined in the application's descriptive parameters (found using the command ADSPAPPINT). The possible values are:

- ***NONE**: The inter-application links are not used. Only the components present in the interrogating application are known.

- ***USE**: The inter-application links are used to simulate the real usage context for components. The link is made towards the applications used or with a reciprocal link. In the case of a component homonym, only the first one found is taken.

- ***ALL**: The inter-application links are used to determine all the applications concerned by a component. The link is made in all directions (applications used or using the current application or reciprocal link). In the case of a component homonym, all the applications of this component are displayed, in the order defined in the inter-application links.

Usage type

*USED Only downloads the explicitly used fields from among the referenced fields.

*UPD Only downloads the updated fields from among the referenced fields.

*IMPL Only downloads the explicitly and implicitly used fields from among the referenced fields.

*ALL Downloads all referenced fields.

Sort

*ALPHA to display Alpha types fields first.

*LINE to display the result in the lines' order.

The **Definition** column indicates if the field is *Internal* or *External*.

The **Usage** column indicates how the field is used by the component. The possible values are *Yes*, *Implicit*, *No*.

The **Extended information** section give you information about the source lines for a given component where a given field is used.

8.5 Further field analysis

From a field analysis result, or from the field repository in the ARCAD Observer DB perspective, you can access further field analysis tools.

8.5.1 Reference files analysis

Shows the list of directories that contain the field referenced by the fields in the selection.

Step 1 Select one or several fields in the view then right-click and select **Analyze > Reference files**.

A ⚡ **Cross-references: Reference files** view is displayed showing the analysis results.
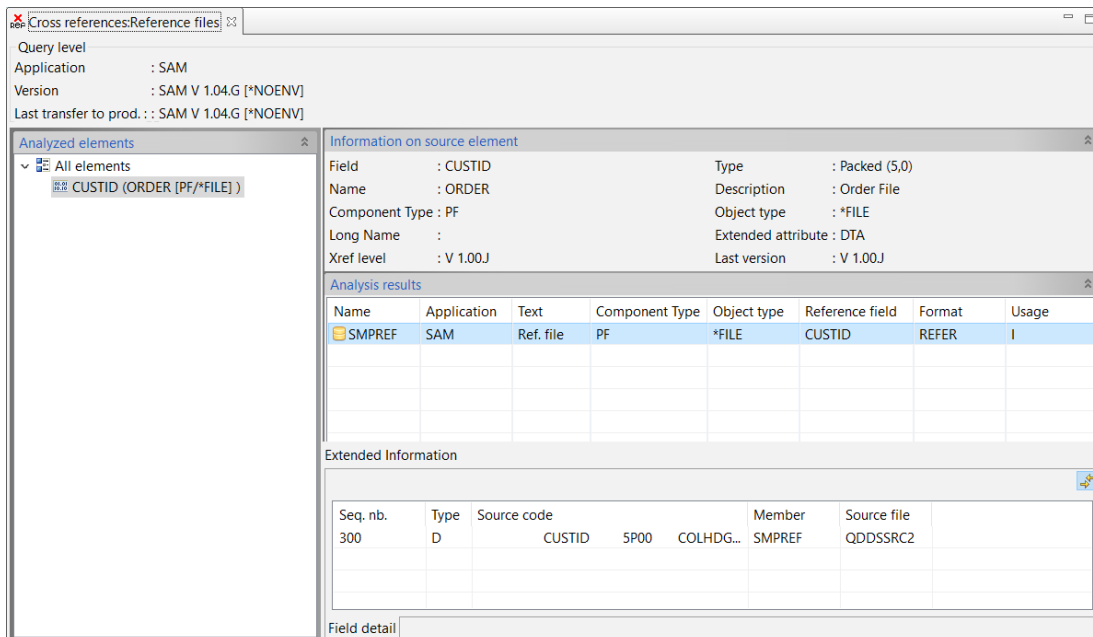
Figure 22: Reference files analysis result view

The **Reference Field** column displays the name of the directory's referenced field.

The **Extended information** section give you information about the source lines for a given component where a given field is used.

8.5.2 Composite fields analysis

Shows the list of composite fields for fields in the selection.

Step 1 Select one or several fields in the view then right-click and select **Analyze Composite fields**.

A 〰 **Cross-references: Composite fields** view is displayed showing the analysis results.
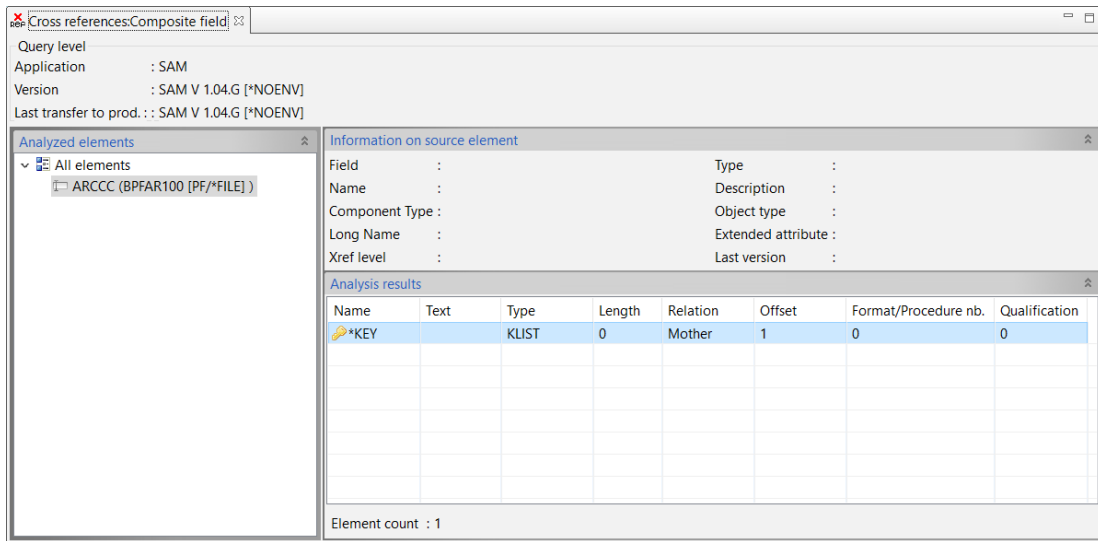
Figure 23: Composite fields analysis result view

The **Relation** column displays the value that identifies the type of hierarchical relationship between the selected field and the composite field. The possible values are *Mother* or *Daughter*.

The **Type** column displays the value that identifies the type of link between the selected field and the composite field.

8.5.3 Components' field and file analysis

Shows the list of components that process the fields in the selection. This search is performed using the selected field/file pair that contains the field.

Step 1   Select one or several fields in the view then right-click and select **Analyze** > **Component** > **Field and file**.

A 〰 **Cross-references: Field and file** view is displayed showing the analysis results.

Figure 24: Fields and files analysis result view

Step 2 Use optional parameters to analyze the results in the call parameter section, then click on the component to refresh the results.

Inter-application links

Specifies whether or not to use inter-application links for cross-reference usage. The list of applications linked to the current application is obtained by examining the links defined in the application's descriptive parameters (found using the command ADSPAPPINT). The possible values are:

- ***NONE**: The inter-application links are not used. Only the components present in the interrogating application are known.
- ***USE**: The inter-application links are used to simulate the real usage context for components. The link is made towards the applications used or with a reciprocal link. In the case of a component homonym, only the first one found is taken.
- ***ALL**: The inter-application links are used to determine all the applications concerned by a component. The link is made in all directions (applications used or using the current application or reciprocal link). In the case of a component homonym, all the applications of this component are displayed, in the order defined in the inter-application links.

Usage type

*USED Only downloads the explicitly used fields from among the referenced fields.

*UPD Only downloads the updated fields from among the referenced fields.

*IMPL Only downloads the explicitly and implicitly used fields from among the referenced fields.

*ALL Downloads all referenced fields.

Sort

> ***ALPHA** to display Alpha types fields first.

> ***LINE** to display the result in the lines' order.

The **Definition** column indicates if the field is *Internal* or *External*.

The **Usage** column indicates how the field is used by the component. The possible values are *Yes*, *Implicit*, *No*.

The **Extended information** section give you information about the source lines for a given component where a given field is used.

8.5.4 Components' field only analysis

Shows the list of components that manipulate the fields in the selection. This search is performed using only the field name and does not include the file the field belongs to.

Step 1 Select one or several fields in the view then right-click and select **Analyze** > **Component** > **Field only**.

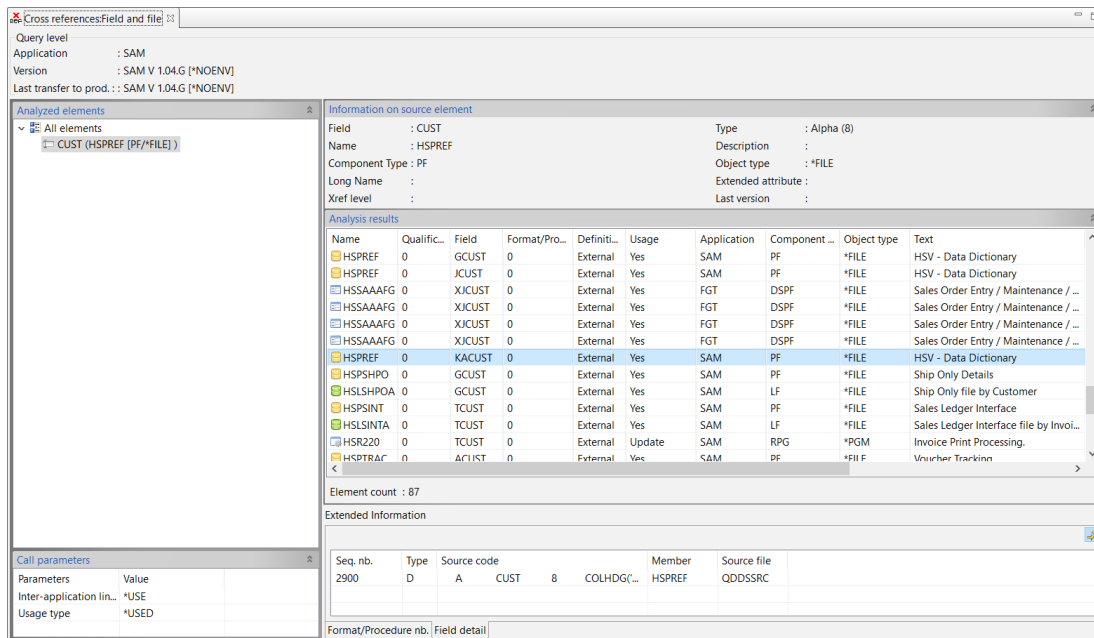> A ✖ **Cross-references: Field only** view is displayed showing the analysis results.

Figure 25: Fields only analysis result view

Step 2 Use optional parameters to analyze the results in the call parameter section, then click on the component to refresh the results.

Inter-application links

> Specifies whether or not to use inter-application links for cross-reference usage. The list of applications linked to the current application is obtained by examining the links defined in the application's descriptive parameters (found using the command ADSPAPPINT). The possible values are:

- ***NONE**: The inter-application links are not used. Only the components present in the interrogating application are known.
- ***USE**: The inter-application links are used to simulate the real usage context for components. The link is made towards the applications used or with a reciprocal link. In the case of a component homonym, only the first one found is taken.
- ***ALL**: The inter-application links are used to determine all the applications concerned by a component. The link is made in all directions (applications used or using the current application or reciprocal link). In the case of a component homonym, all the applications of this component are displayed, in the order defined in the inter-application links.

Usage type

*USED Only downloads the explicitly used fields from among the referenced fields.

*UPD Only downloads the updated fields from among the referenced fields.

*IMPL Only downloads the explicitly and implicitly used fields from among the referenced fields.

*ALL Downloads all referenced fields.

Sort

*ALPHA to display Alpha types fields first.

*LINE to display the result in the lines' order.

The **Definition** column indicates if the field is *Internal* or *External*.

The **Usage** column indicates how the field is used by the component. The possible values are *Yes*, *Implicit*, *No*.
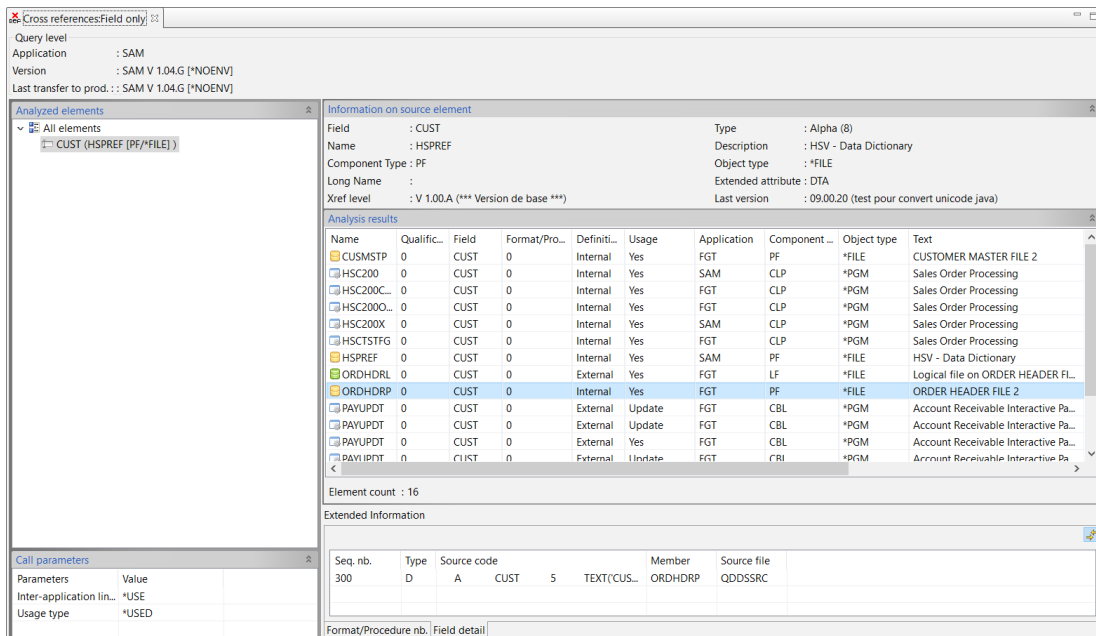
The **Extended information** section give you information about the source lines for a given component where a given field is used.

8.6 Literals analysis

Shows the list of literals that are declared or (directly or indirectly) manipulated by the selected component.

Step 1 Select one or several components in the view then right-click and select **Analyze** > **Literals**.

A **Cross-references: Literals** view is displayed showing the analysis results.

Figure 26: Literals analysis result view

Step 2 Use optional parameters to analyze the results in the call parameter section, then click on the component to refresh the results.

Inter-application links

Specifies whether or not to use inter-application links for cross-reference usage. The list of applications linked to the current application is obtained by examining the links defined in the application's descriptive parameters (found using the command ADSPAPPINT). The possible values are:

- ***NONE**: The inter-application links are not used. Only the components present in the interrogating application are known.
- ***USE**: The inter-application links are used to simulate the real usage context for components. The link is made towards the applications used or with a reciprocal link. In the case of a component homonym, only the first one found is taken.
- ***ALL**: The inter-application links are used to determine all the applications concerned by a component. The link is made in all directions (applications used or using the current application or reciprocal link). In the case of a component homonym, all the applications of this component are displayed, in the order defined in the inter-application links.

Usage type

*USED Only downloads the explicitly used fields from among the referenced fields.

*UPD Only downloads the updated fields from among the referenced fields.

*IMPL Only downloads the explicitly and implicitly used fields from among the referenced fields.

*ALL Downloads all referenced fields.

Sort

*ALPHA to display Alpha types fields first.

*LINE to display the result in the lines' order.

The **Definition** column indicates if the literals is *Internal* or *External*.
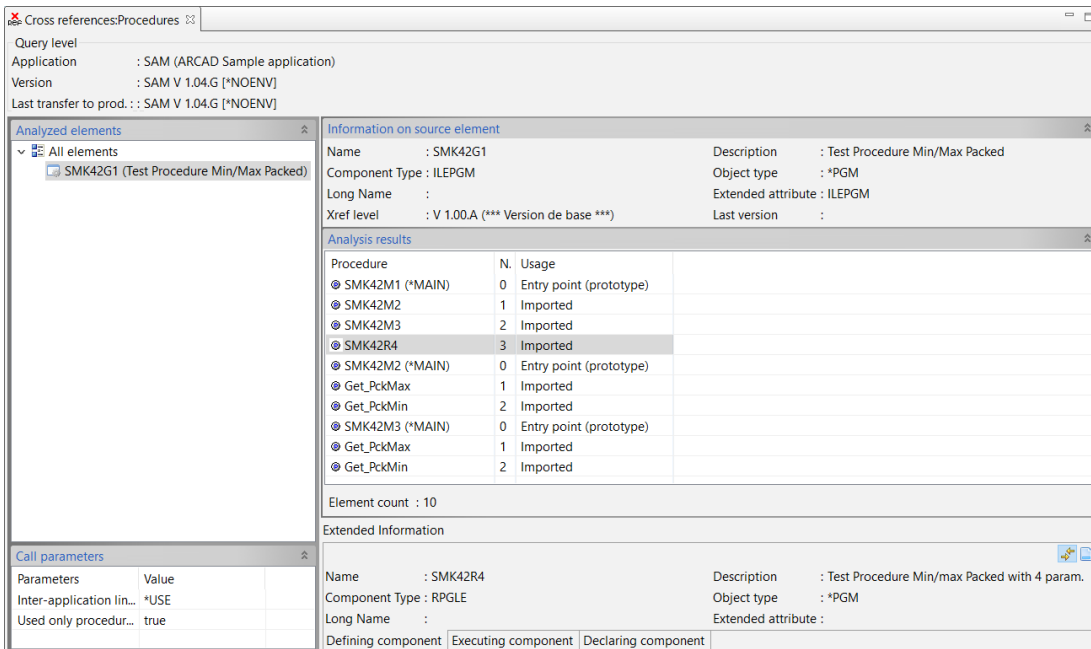
The **Usage** column indicates how the literals is used by the component. The possible values are *Yes*, *Implicit*, *No*.

8.7 Procedures analysis

Shows the list of procedures that are used by the selected component.

Step 1   Select one or several components in the view then right-click and select **Analyze** > **Procedures**.

A **Cross-references: Procedures** view is displayed showing the analysis results.

Figure 27: Procedures analysis result view

Step 2   Use optional parameters to analyze the results in the call parameter section, then click on the component to refresh the results.

Inter-application links

Specifies whether or not to use inter-application links for cross-reference usage. The list of applications linked to the current application is obtained by examining the links defined in the application's descriptive parameters (found using the command ADSPAPPINT). The possible values are:

- ***NONE**: The inter-application links are not used. Only the components present in the interrogating application are known.

- ***USE**: The inter-application links are used to simulate the real usage context for components. The link is made towards the applications used or with a reciprocal link. In the case of a component homonym, only the first one found is taken.
- ***ALL**: The inter-application links are used to determine all the applications concerned by a component. The link is made in all directions (applications used or using the current application or reciprocal link). In the case of a component homonym, all the applications of this component are displayed, in the order defined in the inter-application links.

Parameter	Value
Used only procedures	Click to set to **True** or **False** to display used procedures only.

Table 6: Other cross-reference analysis options

The **Extended information** section give you information about the **Defining**, **Executing** or **Declaring component** of the selected procedure in the result view.

Click the 🖼 Source icon to display the source code of the defining component. The source code of the procedure is displayed in a new view.

8.8 Source inclusion analysis

Shows the list of source inclusions for a given component.

Step 1 Select one or several components in the view then right-click and select **Analyze** > **Source inclusion**.

A ⤴ **Cross-references: Source inclusion** view is displayed showing the analysis results.
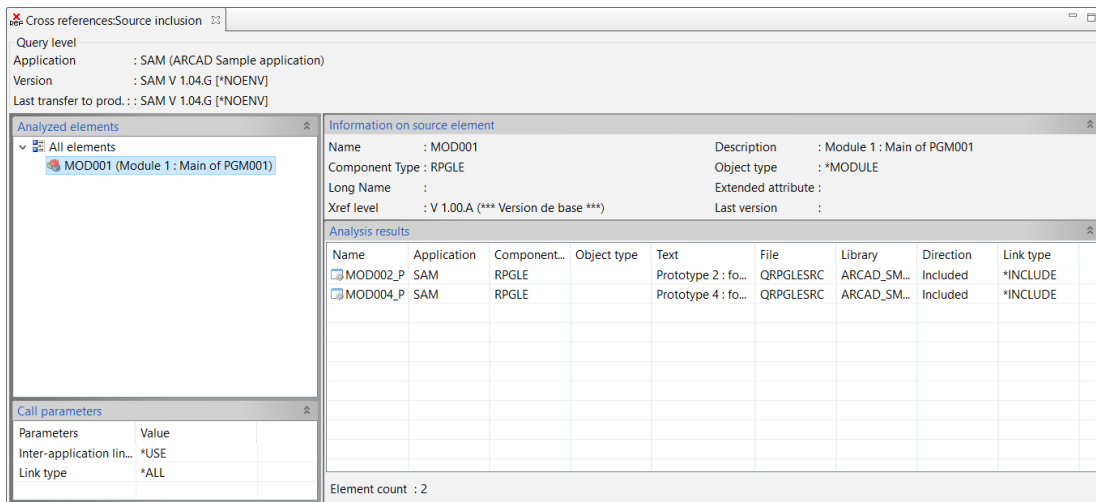
Figure 28: Source inclusion analysis result view

Step 2 Use optional parameters to analyze the results in the call parameter section, then click on the component to refresh the results.

Inter-application links

Specifies whether or not to use inter-application links for cross-reference usage. The list of applications linked to the current application is obtained by examining the links defined in the application's descriptive parameters (found using the command ADSPAPPINT). The possible values are:

- ***NONE**: The inter-application links are not used. Only the components present in the interrogating application are known.
- ***USE**: The inter-application links are used to simulate the real usage context for components. The link is made towards the applications used or with a reciprocal link. In the case of a component homonym, only the first one found is taken.
- ***ALL**: The inter-application links are used to determine all the applications concerned by a component. The link is made in all directions (applications used or using the current application or reciprocal link). In the case of a component homonym, all the applications of this component are displayed, in the order defined in the inter-application links.

Link type

*SIMILAR displays links resulting from component copies.

*INCLUDE displays COPY clause type links

*L4GL3G displays the 3rd and 4th generation languages links types.

*ALL Displays all link types for the selected component.

8.9 ILE encapsulation links analysis

Shows the list of ILE components that use or are used by the selected component. The request also allows you to obtain a graphical representation of the internal calling architecture between modules.

Step 1 Select one or several components in the view then right-click and select **Analyze** > **ILE encapsulation links**.

A ⚡ **Cross-references: ILE links** view is displayed showing the analysis results.

Figure 29: ILE links analysis result view

Step 2 Use optional parameters to analyze the results in the call parameter section, then click on the component to refresh the results.

Inter-application links

Specifies whether or not to use inter-application links for cross-reference usage. The list of applications linked to the current application is obtained by examining the links defined in the application's descriptive parameters (found using the command ADSPAPPINT). The possible values are:

- ***NONE**: The inter-application links are not used. Only the components present in the interrogating application are known.
- ***USE**: The inter-application links are used to simulate the real usage context for components. The link is made towards the applications used or with a reciprocal link. In the case of a component homonym, only the first one found is taken.
- ***ALL**: The inter-application links are used to determine all the applications concerned by a component. The link is made in all directions (applications used or using the current application or reciprocal link). In the case of a component homonym, all the applications of this component are displayed, in the order defined in the inter-application links.

The **Direction** column indicates how the component is linked to the selected component. The possible values are *Used By*, *Using*.

8.10 Messages identifiers analysis

Shows the list of messages identifiers used by a given component.

Step 1 Select one or several components in the view then right-click and select **Analyze > Message Identifiers**.

A ⤳ **Cross-references: Message Identifiers** view is displayed showing the analysis results.

Figure 30: Message identifiers analysis result view

Step 2 Use optional parameters to analyze the results in the call parameter section, then click on the component to refresh the results.

Inter-application links

Specifies whether or not to use inter-application links for cross-reference usage. The list of applications linked to the current application is obtained by examining the links defined in the application's descriptive parameters (found using the command ADSPAPPINT). The possible values are:

- ***NONE**: The inter-application links are not used. Only the components present in the interrogating application are known.
- ***USE**: The inter-application links are used to simulate the real usage context for components. The link is made towards the applications used or with a reciprocal link. In the case of a component homonym, only the first one found is taken.
- ***ALL**: The inter-application links are used to determine all the applications concerned by a component. The link is made in all directions (applications used or using the current application or reciprocal link). In the case of a component homonym, all the applications of this component are displayed, in the order defined in the inter-application links.

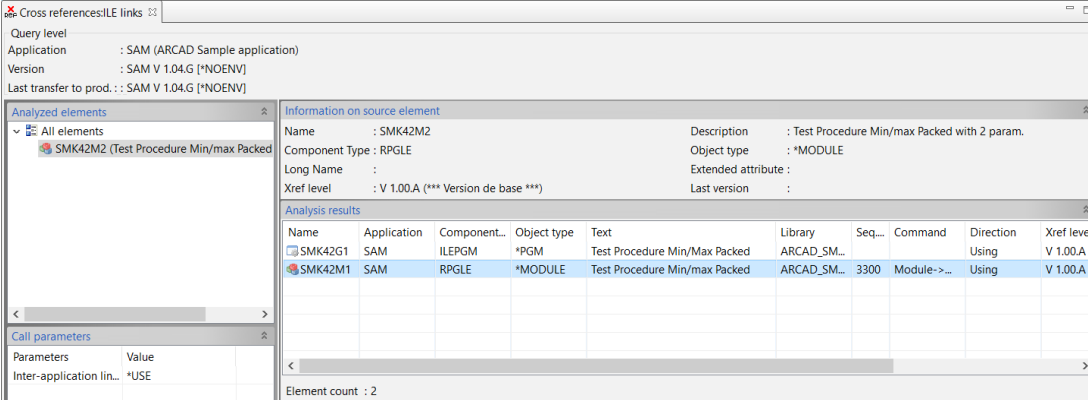The **MSGID** column shows the identifier for the message used.

The **Sequence Number** column indicates at which sequence number the message identifier is used.

8.11 Message use analysis

Shows the list of component using a given message.

> ℹ️ **Note**
> Only available for *MSGF component types.

Step 1 Select one or several components in the view then right-click and select **Analyze > Components using the message**.

A ⤳ **Cross-references: Components using the message** view is displayed showing the analysis results.

Query level
Application   : SAM (ARCAD Sample application)
Version   : SAM V 1.04.G [*NOENV]
Last transfer to prod. : : SAM V 1.04.G [*NOENV]

Analyzed elements
- All elements
  - HSM001 (HSV - System message File)

Call parameters

Parameters	Value
Inter-application lin...	*USE

Analysis results

Name	Application	Component...	Object type	Text	Library	Seq...	MSGID	Xref le...
HSR200	SAM	RPG	*PGM	Sales Order Entry / Maintenance / ...	ARCAD_SM...	197...	HSV0006	V 1.00...
HSR200	SAM	RPGLE	*PGM	Sales Order Entry / Maintenance / ...	ARCAD_SM...	197...	HSV0006	V 1.00...
HSR200E...	SAM	RPGLE	*PGM	V 4.00.E|FOr Web C/S application	ARCAD_SM...	755...	HSV0006	V 1.00...
HSR701	SAM	RPG	*PGM	Inventory Transaction Reports	ARCAD_SM...	248...	HSV0048	V 1.00...
HSR711	SAM	RPG	*PGM	Voucher Tracking Reports	ARCAD_SM...	287...	HSV0048	V 1.00...
HSR721	SAM	RPG	*PGM	Customer Address Reports	ARCAD_SM...	331...	HSV0071	V 1.00...
HSR761	SAM	RPGLE	*PGM	Sales Order Detail Reports	ARCAD_SM...	226...	HSV0048	V 1.00...

Element count : 7

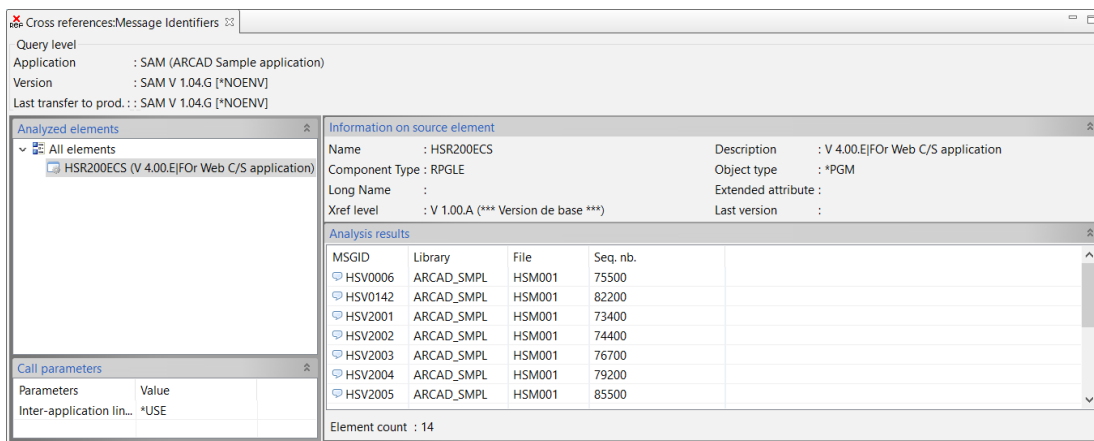*Figure 31: Message use analysis result view*

Step 2 Use optional parameters to analyze the results in the call parameter section, then click on the component to refresh the results.

Inter-application links

Specifies whether or not to use inter-application links for cross-reference usage. The list of applications linked to the current application is obtained by examining the links defined in the application's descriptive parameters (found using the command ADSPAPPINT). The possible values are:

- ***NONE**: The inter-application links are not used. Only the components present in the interrogating application are known.

- ***USE**: The inter-application links are used to simulate the real usage context for components. The link is made towards the applications used or with a reciprocal link. In the case of a component homonym, only the first one found is taken.

- ***ALL**: The inter-application links are used to determine all the applications concerned by a component. The link is made in all directions (applications used or using the current application or reciprocal link). In the case of a component homonym, all the applications of this component are displayed, in the order defined in the inter-application links.

9 Database relationships

Perspective	✏️ ARCAD Observer
Access	🖥️ *Server* > 🖧 Applications > 🖧 *Application Name* > 🌐 Repository

Chapter Summary

ARCAD Observer gives you a detailed overview of all the database relationships of your applications. Create automatically diagrams of the database relationships, and use them to document your application.

9.1 🖳 Prerequisites

Before working with database relationships in ARCAD Observer, you need to run commands to be able to calculate the database relationships of your application.
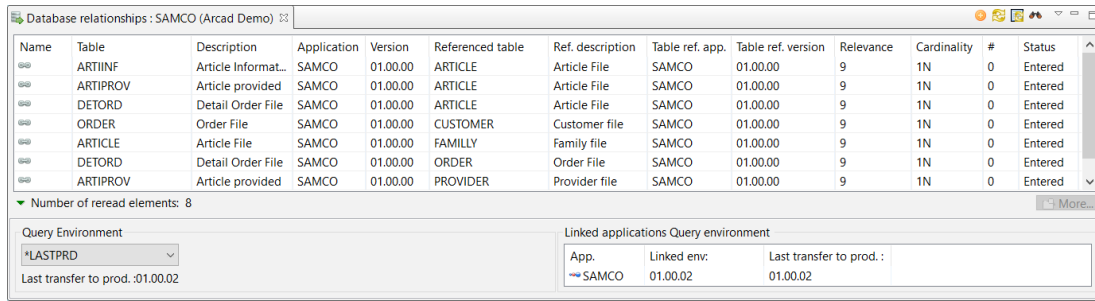
In the ARCAD 5250 emulator or in the command feature or the RDi remote system perspective, run the following ARCAD commands:

- The AUPDPFKEY command is used to generate primary keys that will be used to calculate the database relationships.

- The AUPDFLDDBR command is used to generate database links at the field level.

> 📑 *Reference*
> For more information about specific commands for ARCAD Observer, refer to the appendix.

9.2 ⬢ Accessing database relationships

Figure 32: The Database Relationships view

By default, the ⬢ **Database relationships** view display the 1000 first database relationships. Click the **More…** button to display the next 1000 database relationships.

Click the 🔄 **Refresh selected** or 🔄 **Refresh all** icons to refresh the database relationship view with the latest version of the items.

9.2.1 Searching for database relationships

To refine the list of database relationships displayed, use multiple filters using the **Search Parameters** dialog.

Select the **Query Environment** from the drop-down list to refine the search to a specific environment.

*LASTPRD

The last level used to transfer the application to production.

*LAST

The last level used to update the cross-references for each component.

Version number

Define a specific level to use.

Click the 🔍 Search icon to display the **Search Parameters** dialog.
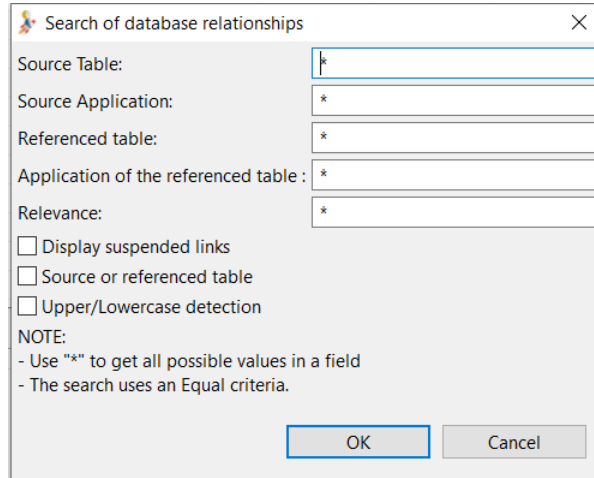
Figure 33: Search Parameters for database relationships

Enter all or part of a database relationship's attribute, then click **OK**.

> ℹ **Note**
>   Use "*" to get all possible values in a field.

Source

Table Filter the components linked to a particular source table. Enter all or part of a table's name.

Source Application

Filter the components linked to a particular source application. Enter all or part of an application's name.

Referenced Table

Filter the components linked to a particular referenced table. Enter all or part of a table's name.

Application of the referenced table

Filter the components linked to a particular application containing a referenced table. Enter all or part of an application's name.

Display suspended links

Tick to display links that are suspended.

Source or referenced table

Tick to search both source or referenced table with the defined search parameters.

Upper/Lowercase detection

Tick to take word case into account for the search.

9.2.2 Displaying database relationships' properties

Follow the subsequent steps to display the selected procedure's properties.

Step 1 Right click on the database relationship item in the view and select **Properties** > ⚙ **Source table** or ⚙ **Referenced table** in the contextual menu.

Result A dialog will open and show the database relationship's properties.
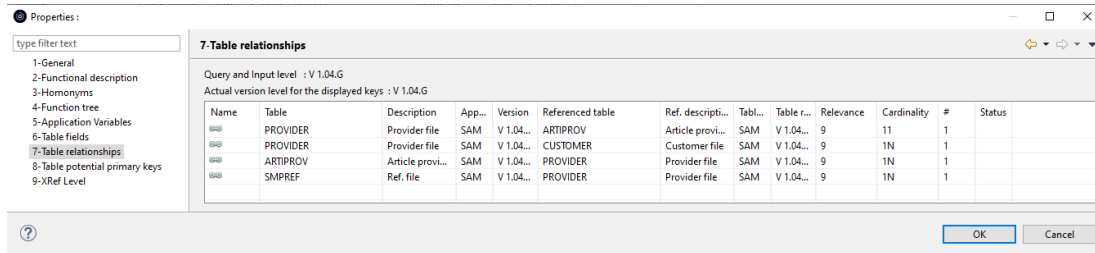
Figure 34: Database relationships' properties

> ⚠ **Important!**
> Depending on the property selection, this section shows the source or referenced table's properties.

General

This section shows the source or referenced table's properties.

Functional Description

This section shows the source or referenced table's functional description.

Homonyms

Shows the list of homonyms in the whole repository.

Functional Tree

Shows a tree displaying the branches of the functional tree to which the table belongs. The node marked with a red M represents the main tree.

Application variable

Shows the list of application variables.

Table Fields

[Only available for tables] Shows the list of table fields.

Table Relationships

[Only available for tables] Shows the list of table relationships.

Table Potential Primary Keys

[Only available for tables] Shows the list of table potential primary keys.

XRef Levels

Shows the source or referenced table's cross-reference level.

You can modify the primary keys from this dialog. Enter your modifications in the **Table Potential Primary Keys** tab.

Click **OK** to save your modifications or **Cancel** to cancel.

9.3 🌐 Creating database relationships manually

The majority of cross-references is automatically calculated by the ARCAD, however you may wish to manually create a link between components.

Follow the subsequent steps to create manually a database relationship between components.

Step 1   To create a database relationship, click the 🌐 Create icon in the database relationship view.

The 🌐 **Database Relationship** editor opens.

Step 2   Enter a **Name** for the database relationship in the **Identification** section.

Step 3   Select the **Relevance** level by entering a relevance level number or selecting it with the increment selector. By default, the relevance level is set to 9.

Step 4   In the **Tables** section, select an **Application** in the drop-down list for the source and referenced tables.

- Click the ⬚ Browse icon to select a table.
- Enter all or parts of the name in the **Table** field of the table selector dialog. A list of tables is displayed.
- Click to select the table then click **OK**.

Step 5   Select the unique key in the drop-down list of the **Fields** tab.

Step 6   Define the relationship condition. Enter the relationship condition in the **Fields** tab.

Step 7   Enter a description in the **Description** tab.

Step 8   Save the changes (💾, Ctrl+S or **File** > **Save**).

Result   The database relationship is created.

9.4 🌐 Editing database relationships

Existing database relationship can be modified. To edit a database relationship, right-click on the item in the database relationship view and select 🌐 Edit Database Relationship in the contextual menu.

The 🌐 **Database Relationship** editor opens. Not all fields can be edited for the database relationships calculated by ARCAD. Only the database relationships created manually can be entirely edited.

9.5 🌐 Suspending or 🌐 releasing database relationships

To suspend a database relationship, right-click on the item in the database relationship view and select 🌐 **Suspend / Release relationship**.

To release the database relationship, right-click on the item in the database relationship view and select again 🔁 **Suspend / Release relationship**.

9.6 🔗 Generating database relationship diagrams

> 📑 *Reference*
> For more information about generating diagrams, refer to Diagrams on page 82.

9.7 🗑 Deleting database relationships

> ⚠ *Warning!*
> Deleted database relationship cannot be accessed or recovered. You need to recreate the relationship manually if needed.

To delete a database relationship, right-click on the item in the database relationship view and select 🗑 Delete. Click **OK** in the dialog to confirm or click **Cancel** to keep the database relationship.

10 Work lists

Perspective	🖊 ARCAD Observer

Chapter Summary

The component work list is a temporary list into which field or component type elements can be transferred. This allows you to create groups of elements originating from different exploration result lists. You can manually add items to the component work list or import into it a whole lot of items from existing ARCAD lists.

> ℹ️ **Note**
>
> Whichever method you use to populate the work list, a uniqueness check is automatically launched. An element cannot be inserted into a work list if it already exists in the work list.

> ⚠️ **Important!**
>
> The component work list is a temporary list. When you close ARCAD Observer, the work list is emptied.

For ARCAD Observer, the 📝 **Components Work List** is useful to group the items to use to document your application via the documentation features.

> 🔁 **Reference**
>
> For more information about documenting applications, refer to Introduction to documenter on page 88.
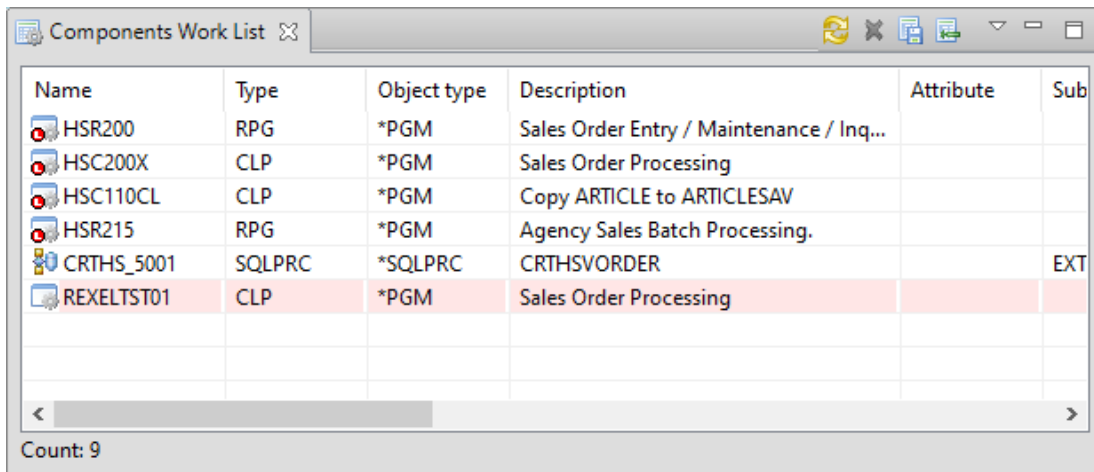
Figure 35: The Components Work List view

10.1 🖼 Adding components to the component work list

You can add any type of component to the work list.

To add a component in this work list, right-click on a component in the **Repository** view, and select 🖼 **Add to work list** in the contextual menu.

The component is added to the work list. The **Component Work List** view opens automatically when you add a component to the list. Click the 🔄 Refresh icon if the component is not displayed in the view.

Figure 36: The Component Work List showing inter-application components

> ⓘ **Note**
> Components in pink are components called by inter-application links. They do not belong to the current application.

10.2 🖼 Removing components from the component work list

You can remove components from the work list one by one or clear the list of all the components.

To remove one or several components from the work list, select one or more components in the view (Ctrl+click to select several components), then right-click and choose 🖼 **Delete from current work list** in the contextual menu.

To remove all the components from the work list, click the 🖼 **Clear the work list** icon in the **Components Work List** view.

> ⓘ **Note**
> Components removed from the work list can still be found in the repository.

10.3 🖼 Importing components from an existing ARCAD list

Follow the subsequent steps to import the components of an ARCAD list into the work list.

Step 1 In the **Components Work List** view, click the 🖼 Import List icon.
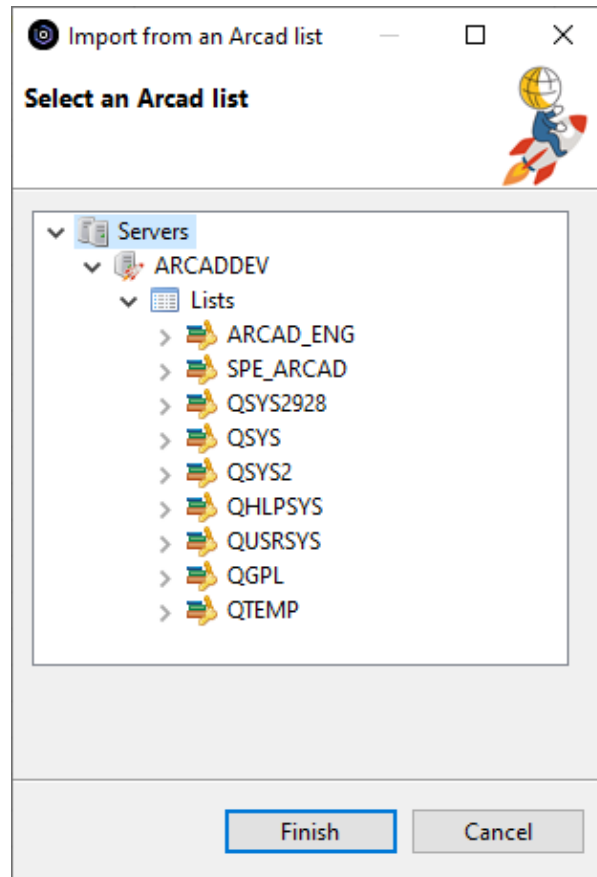
Figure 37: Importing components from an ARCAD list

Step 2 Select the ARCAD list in the dialog. The dialog displays all the lists available in all the libraries added to the list manager for the connected ARCAD Server.

Result The components are added to the work list. If the component is not displayed in the work list, click the 🔃 Refresh icon of the **Components Work List** view.

10.4 🖼 Saving the component work list

The work list is only a temporary list of items and the content of this list is reset when closing ARCAD Observer. To be able to keep the content of the work list, it must be saved as an ARCAD list. You can save the items of the work list to an existing ARCAD list or create a new ARCAD list for that purpose.

> ⓘ **Note**
> Saving items to an ARCAD list will not empty the current work list.

10.4.1 Save the work list to an existing ARCAD list

Follow the subsequent steps to save the work list.

Step 1 In the **Components Work List** view, click the 🖼 Save Work List icon.
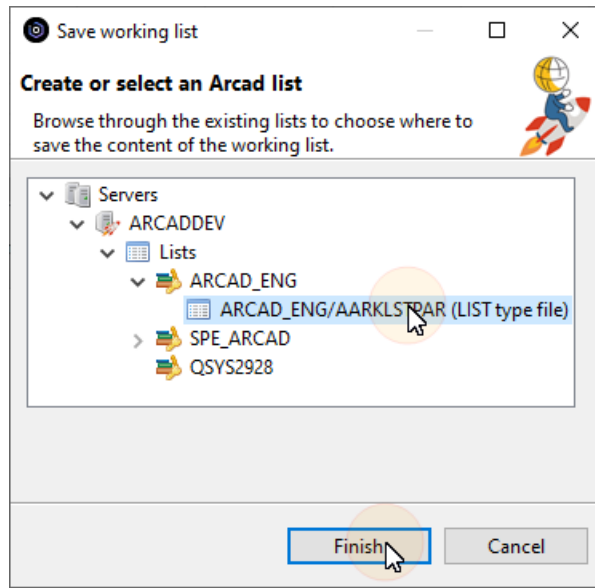
Figure 38: Saving a work list in the library list

Step 2 Browse through the existing lists under the current server's node. Expand the libraries to choose the list where to save the content of the working list.

Step 3 Click **Finish**.

Result The new list is saved to the **Lists** in the navigator.

10.4.2 Save the work list to a new ARCAD list

Follow the subsequent steps to save the work list.

Step 1 In the **Components Work List** view, click the 🖼 Save Work List icon.

Step 2 Expand the Lists node under the current server's node in the dialog. The **Save** dialog allows you to create a specific ARCAD list to save the contents of the work list. if needed, you can also create a new library to save the list.

Create a new library

- To create a new library, right-click on a the lists node and select 🗒 **Create library**.
- Enter a library name. Click **Finish**.

Create a new list

- To create a new list, right-click on a library and select 📄 **Create an empty list**.
- Enter a list name and description in the **Empty list creation wizard**. Click **Next**.
- Click the **Prompt command** button. Define the parameters to use for the list creation command. Click **OK**.
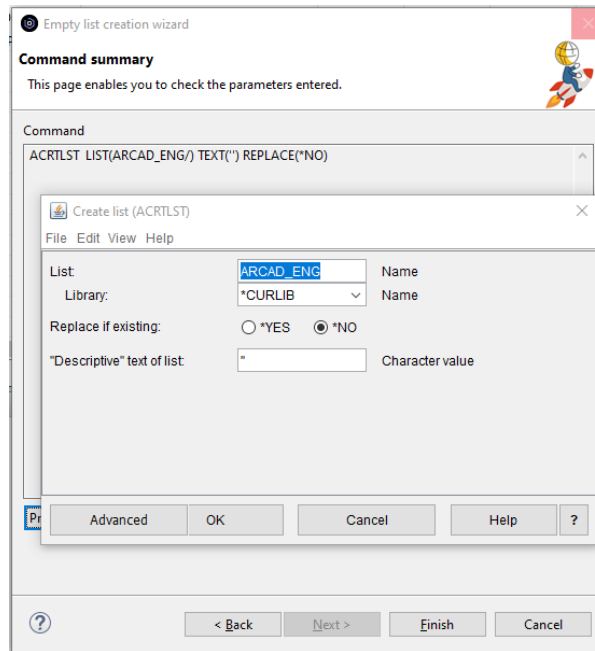
Figure 39: The list creation command parameters

Step 3   Click **Finish**.

Result   The new list is saved to the **Lists** in the navigator.

DIAGRAMS

Introduction to diagrams

This section will guide you through the diagram features of ARCAD Observer.

11 Diagrams

Perspective	✎ ARCAD Observer
Access	🖳 Server > 🖧 Applications > *Application Name* > 🖼 Documents > 🕸 Diagrams
	🖳 Server > 🖼 Documents > 🕸 Diagrams

Chapter Summary

ARCAD Observer handles data representation in diagram form. To facilitate learning, all diagrams share some common interface qualities.

11.1 Types of Diagrams

ARCAD Observer can generate different types of diagrams. The types of diagram you can generate depends on the type of element chosen (database relationships, components, application, etc.)

Input / output diagrams

The input / output diagram is intended to graphically represent interactions between a program and the files it uses, or between a file and the programs that use it.

Flow diagrams

Flowcharts are a representation of the interactions between different types of components. They can include use and calling links.

Database diagrams

Database relationship diagrams are graphical representations of the existing relationships between database tables. These links are extracted by a *reverse engineering* engine that functions on the IBM i server.

Macroscopic views

For this specific type of diagrams, refer to Macroscopic views on page 85.

11.2 Accessing diagrams

You can access the existing diagrams for an application or see all the diagrams created for all your applications from the navigator under 🖼 **Document** > 🕸 **Diagrams**.

11.3 Generating diagrams

Diagram can be generated from any view displaying components, database relationship, etc.

To generate a diagram, right-click on an element and select **Diagrams**, then select the type of diagram you wish to generate.

The type of diagrams available in the contextual menu depends on the type of element selected (components, database relationships, etc.)

11.4 Editing diagrams

Once they are generated, you can still edit the diagrams.

In the Diagram list dialog (🖊 *Navigator* > 🖼 Documents > 🕸 Diagrams), you can rename the diagram. Right-click on a diagram and select **Rename** in the contextual menu.

To add a text note to the diagram, right-click anywhere you wish in a diagram (on an element, a connector, in the background...) and select **Add Note** in the contextual menu.

To export the diagram to image formats, right-click in it and select **File** > **Save as image file**. Define the image parameters in the dialog. Click **OK**.
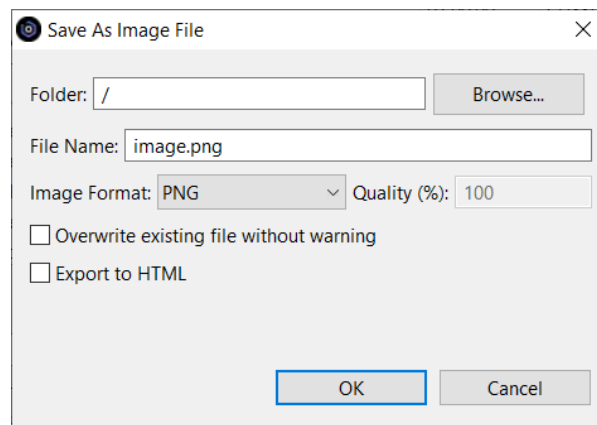
Figure 40: Save diagram as image files

11.5 Using diagrams in documentation

Diagrams can be added to the documentation of your application. The documentation generates automatically diagrams for the components if the corresponding documentation section are included in the plan. You can add other diagrams to the documentation, such as the ones that you generated manually, and add them to the documentation by using customized sections in the plan.

> 📑 *Reference*
> Refer to .

11.6 Deleting diagrams

> ⓘ *Warning!*
> Deleted diagrams cannot be accessed or recovered

To delete a diagram, right-click on the item in the diagram list dialog and select ⊗ **Delete**. Click **OK** to confirm or click **Cancel** to keep the diagram in the dialog.

12 Macroscopic views

Perspective	🖊 ARCAD Observer
Access	🖥 Server > 🌐 Applications > *Application Name* > 🖼 Documents > 🔬 Diagrams 🖥 Server > 🖼 Documents > 🔬 Diagrams

Macroscopic views are diagrams for the high level analysis of application architecture and dependency.

To generate a macroscopic view, right-click on the 🌐 application and select 🔬 **Macroscopic view** in the contextual menu.
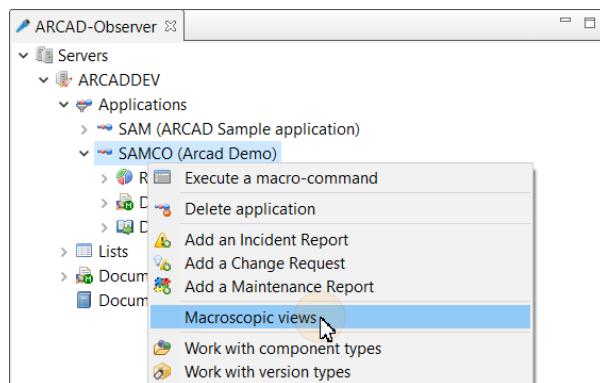
Figure 41: Generate macroscopic views of an application

The macroscopic view diagram is empty when it is created.

To display the content of your choice in the application, right-click in the diagram and select **Load content** in the contextual menu. Tick to select the items you wish to display in the diagram in the dialog. Click **OK**.
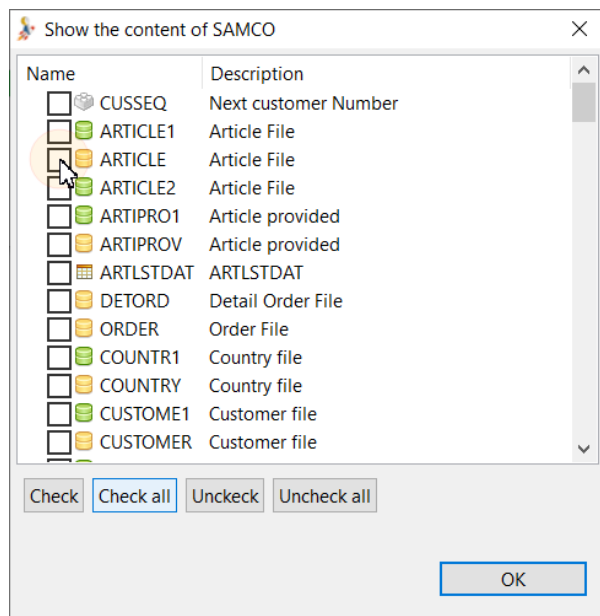
Figure 42: Select the content of the application

The macroscopic view diagram displays the items selected.
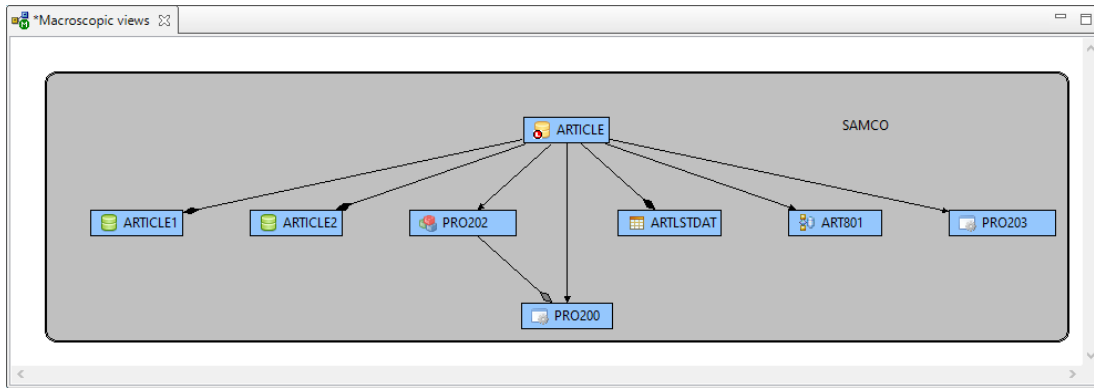
Figure 43: The macroscopic view of an application

You can broaden the macroscopic view diagram by displaying the impact of the items selected.

To display the impact of an item in the diagram, right-click on it and select the **Show impact** in the contextual menu. Select the elements you wish to display in the dialog. Click **OK**.

Alternatively, if you wish to display all the elements related to an item in the diagram, right-click on it and select **Load the entire impact of this element** in the contextual menu.

To export the diagram to image formats, right-click in it and select **File** > **Save as image file**. Define the image parameters in the dialog. Click **OK**.
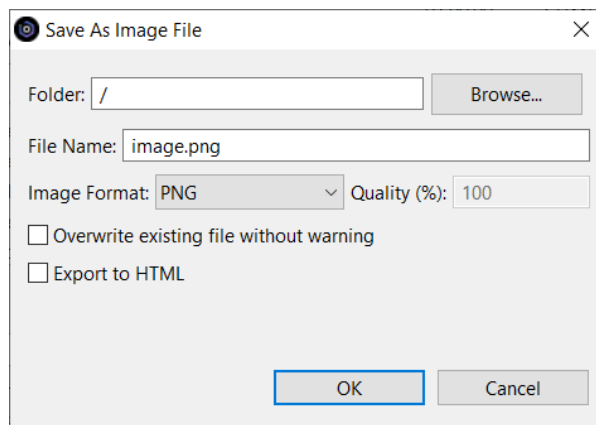
Figure 44: Save diagram as image files

To delete an item from the macroscopic view, right-click on it and select ⊖ **Remove from diagram** in the contextual menu.

DOCUMENTATION

Introduction to documenter

This section will guide you through the documentation features of ARCAD Observer called Documenter.

13 Documentation models

Perspective	🖉 ARCAD Observer
Access	🖳 Server > 🔳 Models

Chapter Summary

Documentation models define all the elements (chapter, sections) needed to create a documentation. The documentation model is the outline of the documentation contents, a set of chapters and sections, each presenting specific information that you choose. In documentation models, you also define the format and presentation for every chapter, section, body text and tables for the generated documents. The documentation model is then used as a template when creating documentations.

13.1 Accessing documentation models

Step 1   In the **ARCAD Observer** navigator, open **Servers** > the desired server > 🔳 **Documentation models**.
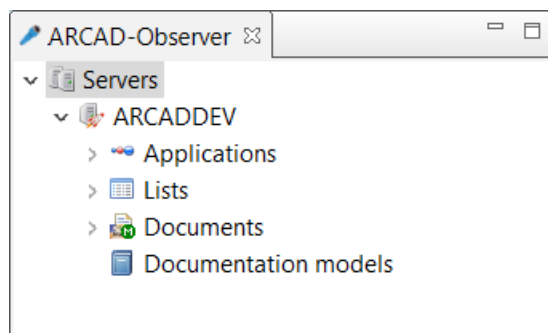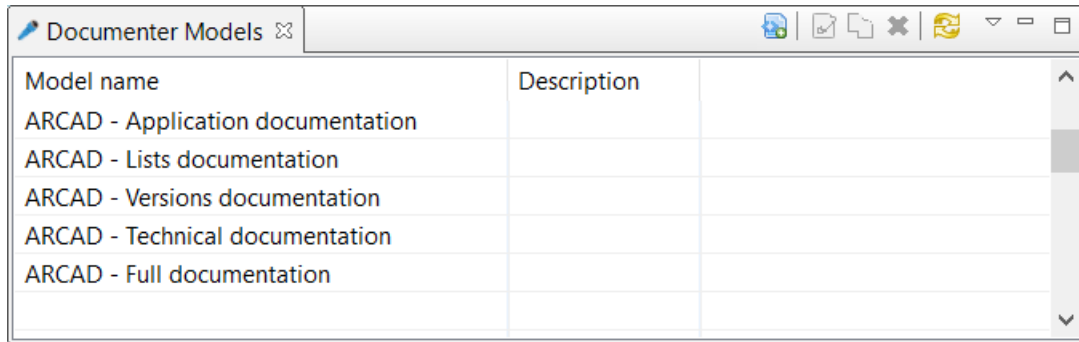
Figure 45: Accessing documentation projects

Result   The list of the documentation models is displayed in the 🖉 **Documenter Models** view.

13.2 Standard documentation models types

ARCAD provides standard documentation models corresponding to a specific documentation issue. These standard documentation models can be duplicated and / or edited.

Figure 46: Standard documentation models

Application documentation

This type of documentation provides an overall documentation of applications.

Version documentation

This type of documentation provides an overall documentation of ARCAD versions.

List documentation

List documentation consolidates into a single document the contents of different lists created on the server. If you wish to analyze existing lists and use customization mechanisms, you can generate a genuine audit report.

Technical documentation

This type of documentation provides a general, non-detailed documentation. It is intended for those who wish to have an overall view of an application and the interactions between the different internal components.

13.3 Creating documentation models

There are two ways to create documentation models:

13.3.1 Creating a documentation model

Follow the subsequent steps to create a documentation model.

Step 1   In the **Documenter Model** menu, click the 🖫 **Create Model** icon.

Step 2   Enter a name and a description for the new model in the model creation wizard. Click **Finish**.

Result   The documentation model is created and ready to be populated with chapters and sections.

13.3.2 Duplicating a documentation model

Duplicating a documentation model enables you to quickly reuse the structure, without having to redefine the same documentation elements multiple times. When a documentation model is duplicated, the new model contains the same information as the original and can be edited to only change specific fields.

Typically, standard documentation models can be duplicated to customize a documentation type.

Follow the subsequent steps to duplicate a documentation project.

Step 1 In the **Documenter Model** menu, select a documentation model and click the Duplicate Model icon.

Step 2 Enter a name and a description for the duplicated model in the model duplication wizard. Click **Finish**.

Result The documentation model is duplicated and ready to be populated with the desired chapters and section.

13.4 Editing documentation model plans

Model plans are presented in an outline. The root contains the documentation type. First-level nodes represent chapters, and second-level nodes represent sections.

Each documentation type has a finite, pre-defined number of chapters and sections. For this reason, content management means choosing the elements that will appear in your documentation from the available elements. You can also define the order in which they will appear in your documentation.

When a model is created, all the chapters are included in the documentation model plan. You can then choose to delete the chapters that don't match the objectives you have set and add the sections to the chapters you wish to keep.

13.4.1 Adding a chapter

Follow the subsequent steps to add a chapter to the documentation model.

Step 1 Right-click on the root node and select **Insert a Chapter** in the contextual menu.

Step 2 select the chapter you wish to add in the **Insert a Chapter** options.

> **ⓘ Note**
> The new chapter is inserted at the bottom of the list. Reorder the list of chapters using the **Move the chapter up** and **Move the chapter down** actions in the contextual menu.

Step 3 Save the changes (Click the **Save the model** icon, Ctrl+S or **File** > **Save**).

To remove a chapter, right-click on the chapter it in the model and select **Remove the chapter** in the contextual menu.

> **⚠ Warning!**
> Removing the chapter will remove the sections defined in the chapter for the documentation model.

13.4.2 Adding a section

Follow the subsequent steps to add a section to the documentation model.

Step 1 Right-click on the chapter to which you wish to add a section and select 🖳 **Insert a section** in the contextual menu.

> ℹ️ **Note**
> If the chapter is empty, you can add all the sections available for that chapter type at once by selecting **Insert a section** > **All sections**.

Step 2 Select the section you wish to add in the **Insert a section** options.

> ℹ️ **Note**
> The section is inserted at the bottom of the list. Reorder the list of sections using the **Move the section up** and **Move the section down** actions in the contextual menu.

Step 3 Save the changes (Click the 🖫 **Save the model** icon, Ctrl+S or **File** > **Save**).

To remove a section, right-click on the section it in the model and select 🖳 **Remove the section** in the contextual menu

13.4.3 Reordering the model's chapters and sections

Follow the subsequent steps to reorder the chapters and sections within chapters.

> ℹ️ **Note**
> If you move a chapter, all associated sections will move with it.

Step 1 Right-click on a chapter or a section in the model and select the ⬆️ **Move Up** and ⬇️ **Move Down** actions in the contextual menus.

Step 2 Save the changes (Click the 🖫 **Save the model** icon, Ctrl+S or **File** > **Save**).

13.4.4 Available sections by chapter types

> ℹ️ **Note**
> Custom chapters can also be added to the documentation model. They can be populated with custom sections. All the custom chapters' and sections' title, content and format can be modified.

Application description
- General description
- Operational attributes
- Application naming convention
- Linked applications Sub-systems
- Functions
- Sub-functions
- Field dictionary

- Versions
- User functions
- Application variables
- Procedure repository
- Literals repository
- Libraries
- IFS directories
- Customized section

Program description

- Detailed description
- Functional description
- Function tree
- Homonyms
- Versions
- Input/output diagram
- File cross references
- Modified fields (by file)
- Call chain
- Called programs
- Calling programs
- Procedure
- Maintenance reports
- Copy clauses
- Message identifiers
- ILE links
- Development folder
- Customized section

File description

- Detailed description
- Functional description
- Function tree
- Homonyms
- Versions
- Fields
- Primary keys
- File utilization diagram
- Cross references on programs

- Maintenance reports
- Database relationships diagram
- Information on database links
- Dependent files
- Message identifier
- Trigger
- Customized section

Other components description
- Detailed description
- Functional description
- Function tree
- Homonyms
- Versions
- Maintenance reports
- Message identifiers
- Customized section

Version description
- Detailed description
- Functional description
- Child versions
- Integration log
- Development team
- Test team
- Maintenance reports
- Modified components
- Customized section

List description
- Detailed description
- Element list
- Customized section

13.5 Editing document model's layout and format

Now that you have defined the elements that will appear, and their order, you will set up your document's format.

Formatting depends on the display type used by the element to present its information. There are three major display groups: Text display, table display, and diagram display. Each of these groups corresponds to specific formatting settings.

Even if all of the elements in the documentation use one of these three display groups, some may still offer additional information inherent to the information they represent.

13.5.1 Defining titles

Define here the document title, a document description as well as the title of each chapter or section.

> ⓘ **Note**
>
> The document **Title** is different from the documentation model identifying name defined when creating the documentation model.
>
> The document title is by default the same as the name of the model. Changing the documentation title has no impact on the documentation model identifier set in the definition section.
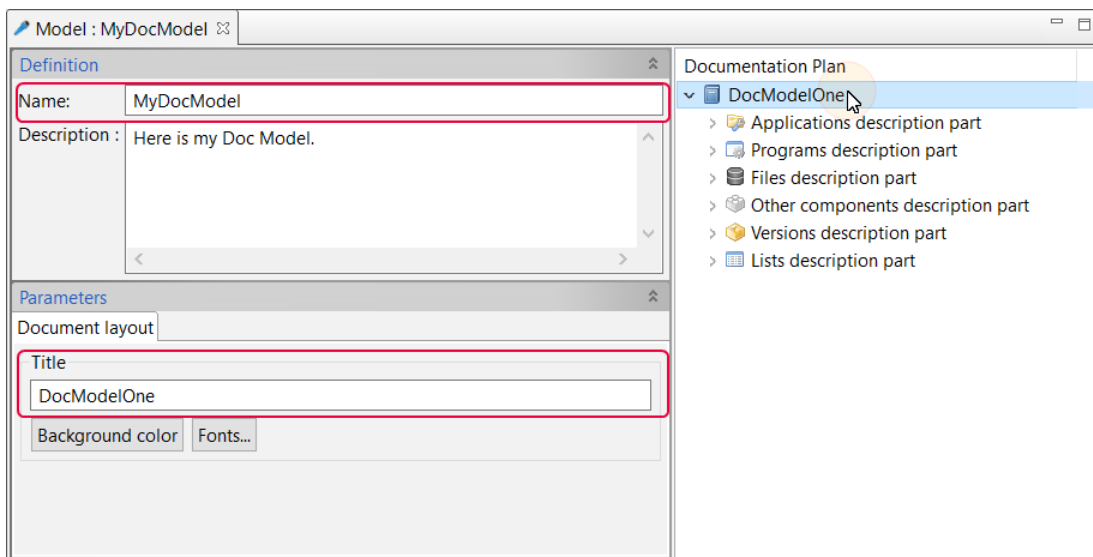
Figure 47: Documentation models titles

Follow the subsequent steps to to define a document title, ad chapter title or a section title:

Step 1   Click on the node in the documentation.

Step 2   Enter a title appropriate to the description of the selected element.

Step 3   Save the changes (Click the 🖫 **Save the model** icon, Ctrl+S or **File** > **Save**).

13.5.2 Defining titles and texts' formats

Define how the information will be displayed.
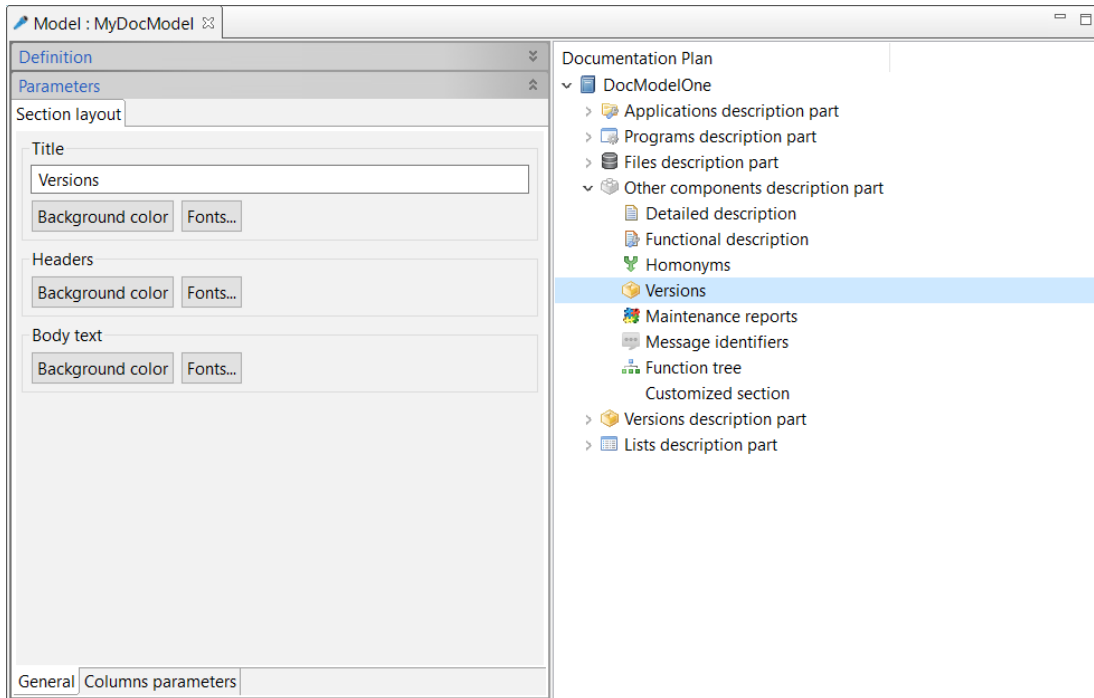
Figure 48: Documentation models layout

Select the color and font of the titles, labels and text body for the documentation. You can also choose the background color for titles and text body.

- Titles (for the document title, chapters and sections titles)
- Layout for the aggregate title (at chapter level)
- Headers (for sections with tables)
- Body text

Text formatting options

Click the **Background color** button to open the standard color selection dialog. Select a color or define a custom one. Click **OK**.
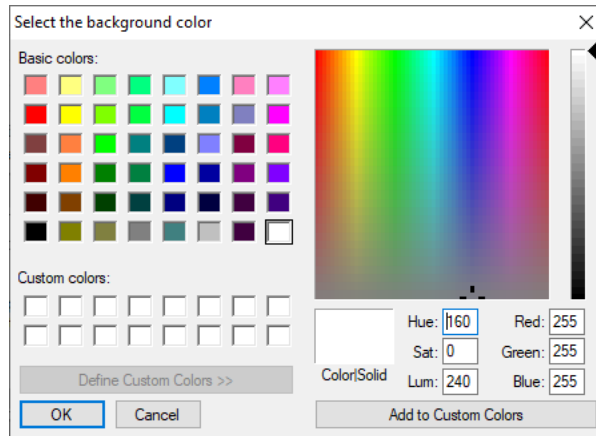
Figure 49: Formatting options - background colors

Click the **Fonts…** button to open the standard font selection dialog. Define the format (text color, font, size, etc.) of the text. Click **OK**.
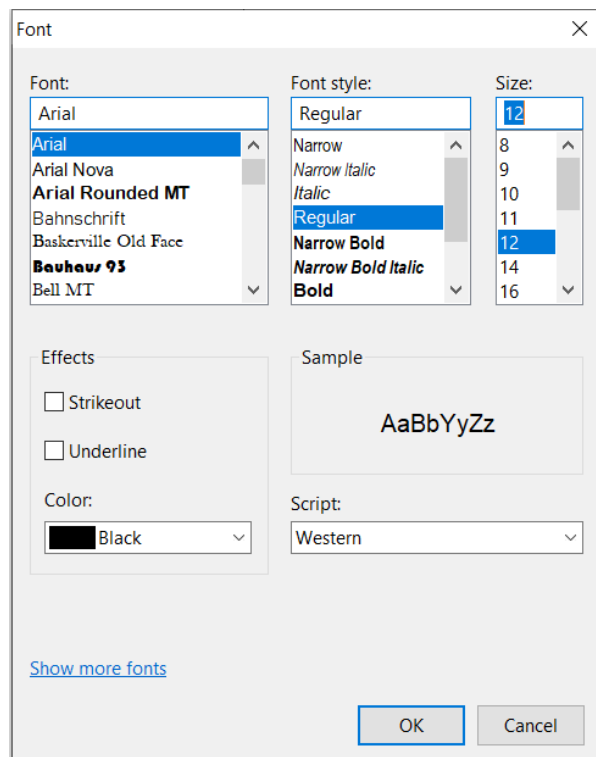
Figure 50: Formatting options - Fonts

Save the changes (Click the 🖫 **Save the model** icon, Ctrl+S or **File** > **Save**).

13.5.3 Defining tables' layout and format

For each table, ARCAD Observer provides the option of choosing the columns displayed, as well as their position, size, and the header text to show.
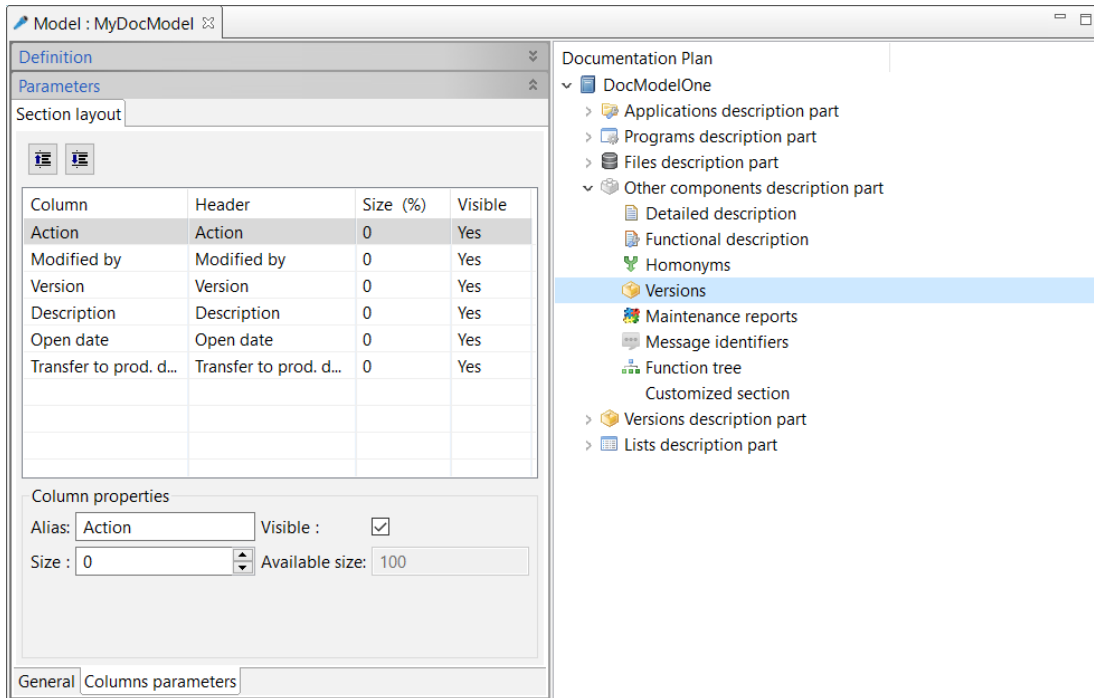
Figure 51: Documentation models tables layout

For the sections of the documentation model presented as a table, the table's layout properties are found in the **Column Properties** tab in the contextual **Parameters** of the section.

- To show or hide the column in the final table, click to select the column in the list and check or uncheck the **Visible** box.
- To modify the header text for your column, click to select the column in the list and enter a new value in the **Alias** entry field.
- To modify column size, click to select the column in the list and enter a value in the size entry field or use the field's increment arrows. The **Available Size** field shows you the remaining size available as a function of the modifications you have made.

> **ⓘ Note**
> Column size is specified as a percentage of overall table size. This is why column value must be between 1 and 100.

- To modify the order of your column's appearance in the table, click to select the column you wish to move and click the ⊕ or ⊕ icons to move the column up or down.

Save the changes (Click the 🖫 **Save the model** icon, Ctrl+S or **File** > **Save**).

13.5.4 Other documentation model options

In addition to the documentation plan and documentation formatting, some optional parameters are available for the documentation model.

These options are available for some of the sections of the model. They can be found in the **Options** tab in the contextual **Parameters** of a section.
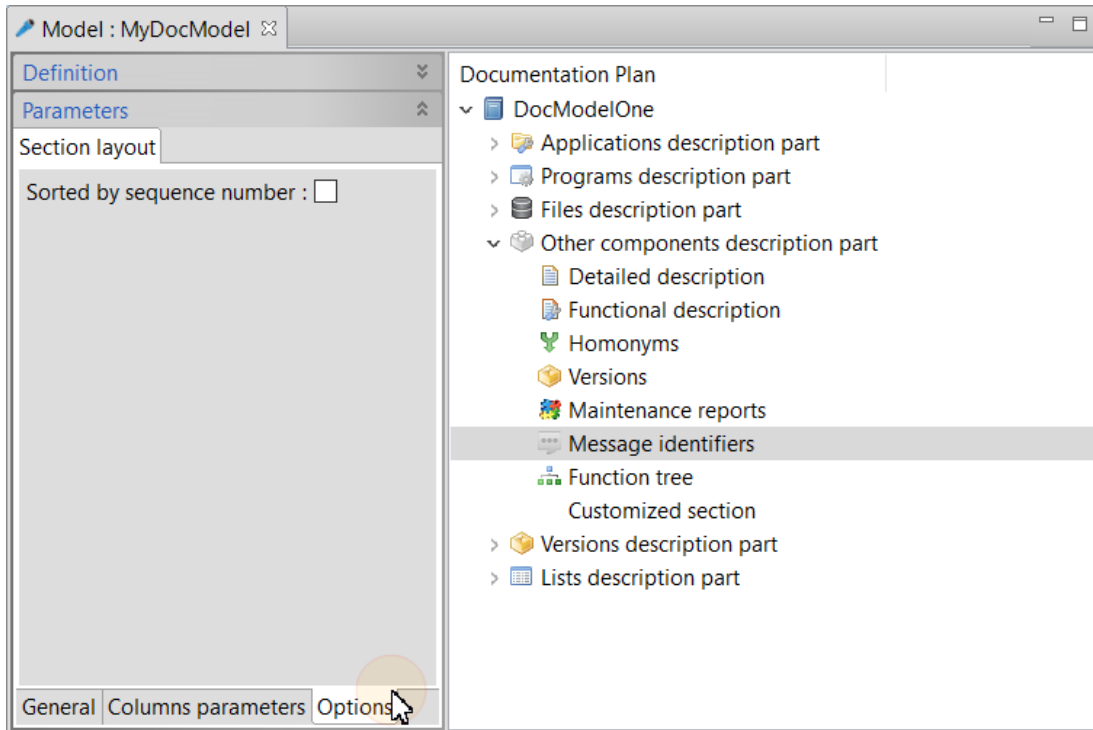
Figure 52: Documentation models options

Fields

ARCAD Observer offers the option of including functional field descriptions. This option is available for:

- **Field dictionary** in the Application description chapter;
- **Modified fields (by file)** in the Program description chapter;
- **Fields** in the File description chapter.

Check the **Include functional description** box in the **Options** tab.

> ⚠ **Important!**
> Printing the functional description has a significant impact on the field documentation. Use this option sparingly.

Call Tree (Program description chapter)

Defining a call tree involves defining the different calculation levels.

Select a calling chain type from the drop-down menu:

- Ascending calling chain
- Descending calling chain
- Ascending and descending calling chain

Called/Calling Programs (Program description chapter)

ARCAD Observer offers you the option to display in the documentation the call parameters headings between two programs.

Check the **Display call parameters heading** box in the Options tab.

Cross-references keys

ARCAD Observer offers you the option to display in the documentation the used keys for the cross-references. This option is available for:

- **File cross-references** in the Program description chapter;
- **Cross-references on programs** in the File description chapter.

Check the **Display used keys** box in the Options tab.

File utilization diagram (File description section)

ARCAD Observer offers you the option to define the maximum size of the file utilization diagram in the documentation. Enter a value for the diagram size **Max. length** and **Max. height** in the section's layout tab.

Messages identifiers

For message identifiers, you can request the messages to be sorted by sequence numbers. This option is available for the **Message identifiers** section in the Program, Files and Other components description chapters.

Check the **Sort by Sequence Number** box in the Options tab.

Maintenance reports (Version description chapter)

> ⚠ **Important!**
> These options are only available for **Version description** chapter's maintenance reports. Other maintenance reports for Program, Files and Other components description chapters follow a standard layout and have no options.

Check the corresponding boxes to include or exclude items in the **Maintenance Report's Information**.

Workload

Estimated load and real load.

Schedule

Opening and closing dates.

Qualification

Maintenance type and action type.

Managers

Change manager and project manager.

To include the incident reports associated with a version, check the **Display incident reports** box in the **Incident report** tab of the section's parameters.

To include a table of components impacted by a version, check the **Display the components table** option in the Components tab.

13.6 Deleting documentation models

> ⚠️ *Warning!*
> Deleted documentation models cannot be accessed or recovered

To delete a documentation model, either right-click on the item in the Documenter Models view and select 📇 Delete, or select the item and click the 📇 Delete icon in the toolbar. Click **OK** to confirm or click **Cancel** to keep the documentation model in the dialog.

14 Documentation projects

Perspective	✎ ARCAD Observer
Access	🗄 Server > 🖧 Applications > *Application Name* > 📖 Documentation

Chapter Summary

A documentation project is linked with an application. You can create several documentation projects for an application, depending on what you want to document. The documentation project contains all the versions of your documentation.

14.1 Accessing documentation projects

Step 1   In the **ARCAD Observer** navigator, open **Servers** > the desired server > **Applications**.

Step 2   In the list of applications, open the desired application.

Step 3   Open 📖 **Documentation**.
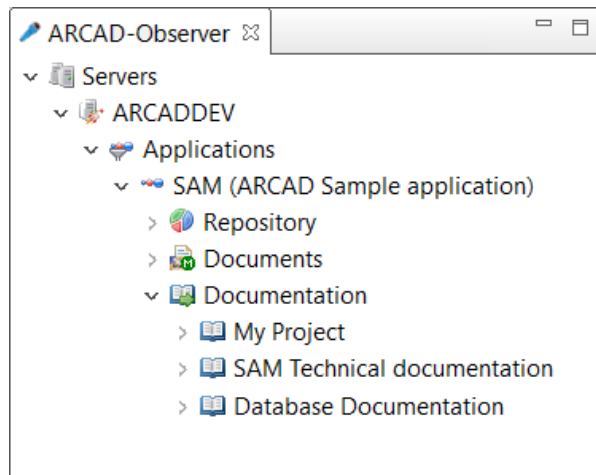
Figure 53: Accessing documentation projects

Result   The list of the documentation projects that already exist for the application is displayed.

14.2 Creating documentation projects

The documentation project is the framework within which all documentation is defined.

Follow the subsequent steps to create a documentation project.

Step 1   In the **Documenter Navigator** view, open the desired application.

Step 2   Right-click the **Documentation** node and select ▣ **Add a documentation project**.

Step 3   Fill in the fields in the **Creating a documentation project** dialog.

Name

Define the name of your project. This is the value that will be your project's identifier. This field is required.

Description

Enter a short description of your project, limited to 255 characters.

Comment

The comments are a longer description, not limited in number of characters, which provides a full description of the contents and objectives of the documentation to be performed.

Default Project

If you check this option, you declare that the project is the default project for the application to be documented.

Project Manager

This drop-down list references all of the ARCAD Observer system's users.

Step 4   Click **Finish**.

Result   The new documentation project is displayed under the **Documentation** node of the application.

14.3 Editing documentation projects

The documentation project properties can be modified.

> ⚠ **Important!**
> Projects are associated to the parent application and you cannot change the application to which the documentation project relates.

Follow the subsequent steps to rename or edit the properties of a documentation project.

Step 1   In the **Documenter Navigator** view, right-click on the desired project and select ⚙ **Properties**.

Step 2   Edit the properties in the **Properties** dialog.
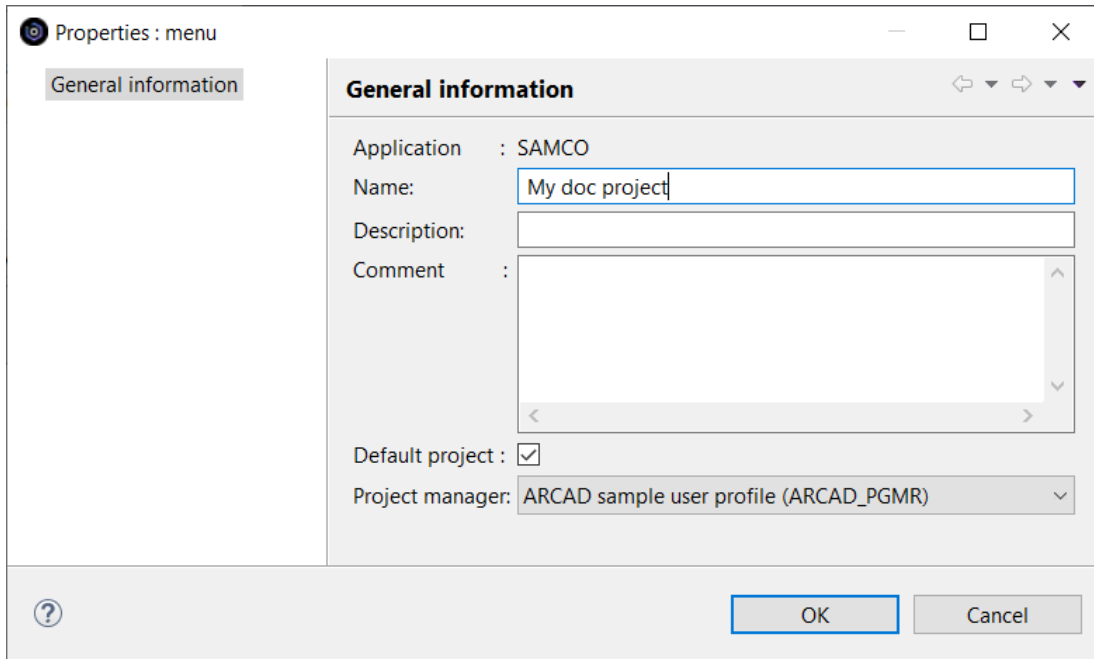
Figure 54: Documentation project properties

Step 3 Click **OK**.

Result The properties are edited. You may need to refresh the ARCAD Observer navigator to see the modifications to the name. Right-click the *[documentation project name]* node and select 🔁 Refresh.

14.4 Deleting documentation projects

> ⚠️ *Warning!*
> Deleted documentation projects cannot be accessed or recovered.

> ℹ️ **Note**
> Documentation projects containing documentation versions cannot be deleted.

To delete a documentation project, right-click on the item in the application navigator and select 📕 Delete. Click **OK** to confirm or click **Cancel** to keep the documentation project in the dialog.

15 Documentation versions

Perspective	🖊 ARCAD Observer
Access	🖥 Server > 🐝 Applications > *Application Name* > 📖 Documentation > 📑 Versions

Chapter Summary

Once a documentation project is set up for you application, you create a first version of the documentation. Your documentation can then be updated, by creating new version of the same documentation.

15.1 Accessing documentation versions

Step 1   In the **ARCAD Observer** navigator, open **Servers** > the desired server > **Applications**.

Step 2   In the list of applications, open the desired application.

Step 3   Open **Documentation** > the desired documentation project > **Versions**.
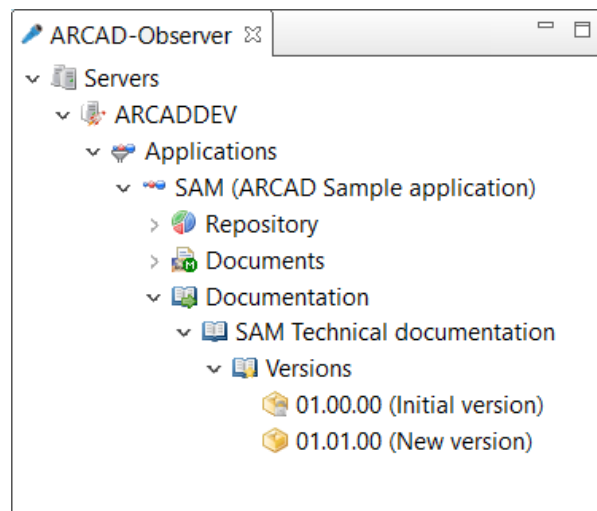
Figure 55: Accessing documentation versions

Step 4   The list of the versions that already exist for the documentation project is displayed.

15.2 Creating documentation versions

Each documentation project comes in versions. To get started with a documentation project, you need to create the first version of it. There are two ways to create a version, either create a new documentation version or duplicate and edit and existing one.

15.2.1 Creating a documentation version

Follow the subsequent steps to create a documentation version.

Step 1   In the **ARCAD Observer** navigator, open the desired documentation project.

Step 2   Right-click the **Versions** node and select 🖼 **Add a documentation version**.

> **Result**   The **Creating a documentation version** wizard opens.

Step 3   Fill in the fields in the dialog:

Start date

Specify the version opening date. By default, this is the current date. To modify, click the calendar icon to open the date selection dialog. Select a date by clicking a day on the calendar. The date highlighted in red is the current date.

End date

Specify the version closing date. To modify, click the calendar icon to open the date selection dialog. Select a date by clicking on a day on the calendar. The date highlighted in red is the current date.

Version number (Mandatory)

The documentation version number uses a three-parts version number: the major number, the minor number, and the sub-version number. Enter a numerical value for each of these parts, or use the increment arrows. It is not possible to create a version if the version number already exists for the documentation project.

> ℹ **Note**
>
> This version number must be unique within each project.
>
> ARCAD Observer verifies that a version associated with a project underway does not have the number you specified.

Query level

Select a query level in the drop-down list:

- ***LASTPRD**: the level used is the one of the application's last transfer to production.
- ***LAST**: the level used is the one of the last update of the cross-references for each component.
- ***CURENV**: the components version level is the one on which you were initialized.
- *Version number*: the specified level is used.

Description

> Enter a short description of your version. This description field is limited to 255 characters. The description is displayed between parenthesis in the ARCAD Observer navigator.

Comment

> Enter a long description, not limited in number of characters, which provides a full description of the contents and objectives of the new version.

Project manager

> Select one of the ARCAD Observer system user referenced in the drop-down list.

Model (Mandatory)

> Select the reference model to be used for your version. Use the down arrow to access different models defined with the Model view.

Step 4   Click **Finish**.

Result   The new documentation version appears under the **Versions** node of the documentation project.

15.2.2 Duplicating a documentation version

Creating a document version by duplicating an existing one is a specific creation operation. It allows the creation of a new version by using a copy of an existing version model. It is also possible to recover the component list and documentation contents.

> ⓘ **Note**
> Besides its practical aspects, this operation is very useful for updating the version, as you will see further in this document.

Follow the subsequent steps to duplicate a documentation version.

Step 1   Select the version you wish to duplicate from the project management tree.

Step 1   In the **ARCAD Observer** navigator, open the desired documentation project.

Step 2   Right-click the documentation version and select 🖼 **Duplicate** in the contextual menu.

> **Result**   The **Duplicate a documentation version** wizard opens.

Step 3   Fill in the fields in the dialog:

Start date

> Specify the version opening date. By default, this is the current date. To modify, click the calendar icon to open the date selection dialog. Select a date by clicking a day on the calendar. The date highlighted in red is the current date.

End date

> Specify the version closing date. To modify, click the calendar icon to open the date selection dialog. Select a date by clicking on a day on the calendar. The date highlighted in red is the current date.

Version number (Mandatory)

The documentation version number uses a three-parts version number: the major number, the minor number, and the sub-version number. Enter a numerical value for each of these parts, or use the increment arrows. It is not possible to create a version if the version number already exists for the documentation project.

> ⓘ **Note**
>
> This version number must be unique within each project.
>
> ARCAD Observer verifies that a version associated with a project underway does not have the number you specified.

Query level

Select a query level in the drop-down list:

- ***LASTPRD**: the level used is the one of the application's last transfer to production.
- ***LAST**: the level used is the one of the last update of the cross-references for each component.
- ***CURENV**: the components version level is the one on which you were initialized.
- *Version number*: the specified level is used.

Description

Enter a short description of your version. This description field is limited to 255 characters. The description is displayed between parenthesis in the ARCAD Observer navigator.

Comment

Enter a long description, not limited in number of characters, which provides a full description of the contents and objectives of the new version.

Project manager

Select one of the ARCAD Observer system user referenced in the drop-down list.

Model (Mandatory)

Select the reference model to be used for your version. Use the down arrow to access different models defined with the Model view.

Step 4   Click **Next >** to continue.

Step 5   Check the **Duplicate the component list**, the **Duplicate the version list** and/or the **Duplicate the list of lists** box to recover the lists.

Step 6   Check the **Duplicate the documentation** box to recover the documentation's contents.

Step 7   Click **Finish**.

Result   The new documentation version appears under the **Versions** node of the documentation project.

15.3 Editing the properties of a documentation version

Once you have created a documentation version, you can modify its general properties. However, it is impossible to edit the properties if the version is closed.

Follow the subsequent steps to edit the properties of a documentation version.

Step 1 In the **Documentation** node of the **Navigator**, right-click on the desired version and select ⚙
 Properties.

> **Result** The **Properties** window of the documentation version opens.

Step 2 Edit the fields in the dialog:

Start date

Specify the version opening date. By default, this is the current date. To modify, click the calendar icon to open the date selection dialog. Select a date by clicking a day on the calendar. The date highlighted in red is the current date.

End date

Specify the version closing date. To modify, click the calendar icon to open the date selection dialog. Select a date by clicking on a day on the calendar. The date highlighted in red is the current date.

Version number (Mandatory)

The documentation version number uses a three-parts version number: the major number, the minor number, and the sub-version number. Enter a numerical value for each of these parts, or use the increment arrows. It is not possible to create a version if the version number already exists for the documentation project.

> ⓘ **Note**
>
> This version number must be unique within each project.
>
> ARCAD Observer verifies that a version associated with a project underway does not have the number you specified.

Query level

Select a query level in the drop-down list:

- ***LASTPRD**: the level used is the one of the application's last transfer to production.
- ***LAST**: the level used is the one of the last update of the cross-references for each component.
- ***CURENV**: the components version level is the one on which you were initialized.
- *Version number*: the specified level is used.

Description

Enter a short description of your version. This description field is limited to 255 characters. The description is displayed between parenthesis in the ARCAD Observer navigator.

Comment

Enter a long description, not limited in number of characters, which provides a full description of the contents and objectives of the new version.

Project manager

Select one of the ARCAD Observer system user referenced in the drop-down list.

Model (Mandatory)

Select the reference model to be used for your version. Use the down arrow to access different models defined with the Model view.

> ⚠️ **Important!**
> You cannot edit the **Version number** and the **Model**. If those fields are not correct for the version created, you have to delete it and create a new one with the right Version number or Model.

Step 3 Click **OK**.

Result The properties are edited. You may need to refresh the ARCAD Observer navigator to see the modifications to the description. Right-click the **Versions** node and select 🔄 Refresh.

15.4 Editing the documentation version

15.4.1 Editing the document structure

When the version is created, the version editor opens with the version structure documentation. This structure comes from the model selected before. At this stage, it is still possible to change the structure (Add, Delete, Move items). However, it is not possible to select a new model for the plan. Modifications to the documentation plan does not affect the documentation plan of the model.

> 📥 **Reference**
> Refer to Editing documentation model plans on page 91.
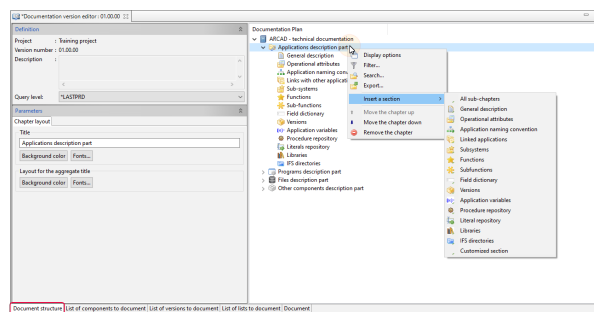
Figure 56: The documentation version plan editor

15.4.2 Editing the document format and layout

When the version is created, the format and layout from the model selected before. At this stage, it is still possible to change the format and layout for each chapter, sections or tables. The format and layout editor is the same as the one used to create the model. Modifications to the documentation layout and format does not affect the layout and format of the model.

> **Reference**
> Refer to Editing document model's layout and format on page 94.

15.4.3 Selecting the elements to be documented

First, it is necessary to select in the version the elements to be documented. These elements can be **Components**, **ARCAD Versions**, and/or **ARCAD Lists**. The process is the same for components, versions or lists, you retrieve the lists of items you wish to document.

> **ⓘ Note**
> It is possible to use items from Components tab, version or list in the same documentation.

15.4.3.1 List of components to document
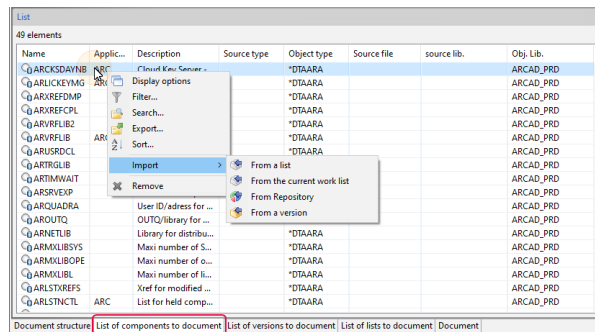
Figure 57: The list of components to document

Components can be imported in four different ways:

Import from a list

In this case, a window allows to choose an ARCAD list of components to proceed.

Step 1   In the **List of components to document** tab of the documentation editor, right-click in the list section and select 🔲 **Import from a list** in the contextual menu.

Step 2   Select the list to document. Navigate in the **Lists** node of the connected server in the dialog and click the desired list. You can document several ARCAD lists of the application but you have to select one after the other.

Click **Next >** to continue.

Step 3   Select the merge options.

Tick to **Merge this imported list with the existing list**.

Tick to **Sort the entire list after the objects are added**.

Click **Finish**.

Result   The list of components is displayed in the editor.

Import from the current work list

The components stored is the work list will be added as components to document.

To import from the current worklist, right-click in the list section of the **List of components to document** tab of the documentation editor, and select ⊞ **Import from the current work list** in the contextual menu.

Import from the repository

In this case, a window displays a selection filter to choose the components.

Step 1   In the **List of components to document** tab of the documentation editor, right-click in the list section and select ⊞ **Import from a repository** in the contextual menu.

Step 2   Use the filters to refine the list of components.

Click **Next >** to continue.

Step 3   [*Optional*] Select one or several functions to restrict to specific functions of the application. Click on the functional module and the click the **[> >]** button.

> ⚠ **Important!**
> Only components that belong to one of the specified branches appear in the list of impacted components.

Click **Next >** to continue.

Step 4   Select the merge options.

Tick to **Merge this imported list with the existing list**.

Tick to **Sort the entire list after the objects are added**.

Click **Finish**.

Result   The list of components is displayed in the editor.

Import from a version

A window allows you to choose an ARCAD version with the components to proceed.

During the merge, ARCAD Observer compares the list of components from the ARCAD version and the list of components from the base ARCAD version.

- New components are added to the corresponding chapters, depending on their types.
- Content entries associated with the deleted components are deleted.
- Information regarding the components marked **Checked Out** or **Modified** are deleted but the contents entries are kept.

Step 1   In the **List of components to document** tab of the documentation editor, right-click in the list section and select 🖼 **Import from a version** in the contextual menu.

Step 2   Select the version in the drop-down list. Click **Next >** to continue.

Step 3   Use the filters to refine the list of components.

Click **Next >** to continue.

Step 4   [*Optional*] Select one or several functions to restrict to specific functions of the application. Click on the functional module and the click the **[> >]** button.

> ⚠️ **Important!**
> Only components that belong to one of the specified branches appear in the list of impacted components.

Click **Next >** to continue.

Step 5   Select the merge options.

Tick to **Merge this imported list with the existing list**.

Tick to **Sort the entire list after the objects are added**.

Click **Finish**.

Result   The list of components is displayed in the editor.

To remove a component from the **List of components to document**, right-click the component in the list section of the editor and select ⛔ **Remove** in the contextual menu.

15.4.3.2 List of versions to document
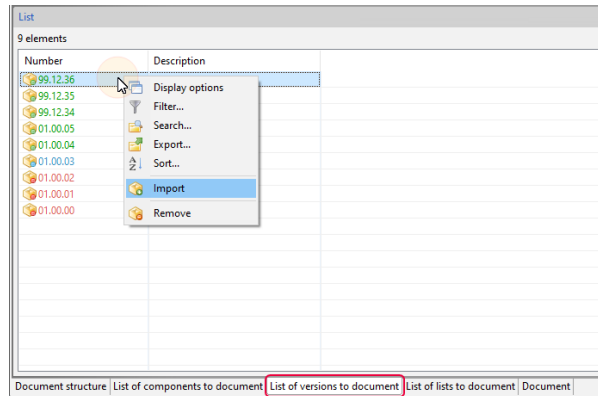
Figure 58: The list of versions to document

It is possible to import **All** the ARCAD versions, or to **Select a range of versions**.

Step 1   In the **List of versions to document** tab of the documentation editor, right-click in the list section and select 🔷 **Import** in the contextual menu.

Step 2   Select the version to document. You can document several ARCAD versions of the application.

Tick **All** to document all the application versions.

Tick **Select a range of versions** to document only a sub-set of versions. In this case, you should define the upper and lower limits of this set by selecting their values from the drop-down lists provided (**From Version** and **To Version**).

> ℹ️ **Note**
> The limits are included in the set to be documented.

Click **Next >** to continue.

Step 3   [*Optional*] Select one or several functions to restrict to specific functions of the application. Click on the functional module and the click the **[> >]** button.

> ⚠️ **Important!**
> Only components that belong to one of the specified branches appear in the list of impacted components.

Click **Next >** to continue.

Step 4   Select the merge options.

Tick to **Merge this imported list with the existing list**.

Tick to **Sort the entire list after the objects are added**.

Click **Finish**.

Result   The list of versions is displayed in the editor.

To remove a version from the **List of versions to document**, right-click a version in the list section of the editor and select 🔶 **Remove** in the contextual menu.

15.4.3.3 List of lists to document
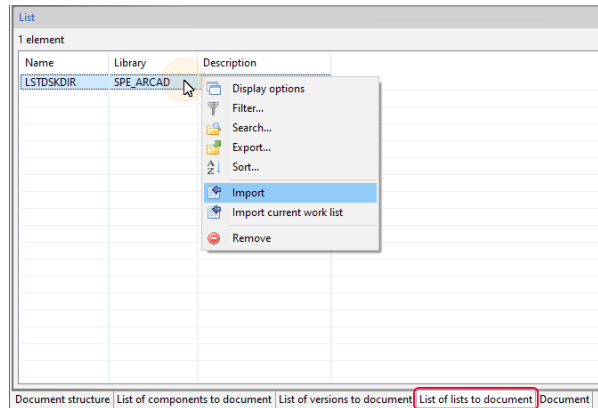
Figure 59: The list of lists to document

Lists can be imported in two different ways:

> ⓘ **Note**
> It is possible to import more than one list.

Import from the current work list

The components stored is the work list will be added as components to documents.

To import from the current worklist, right-click in the list section of the **List of lists to document** tab of the documentation editor, and select 📊 **Import from the current work list** in the contextual menu.

Import from a list

In this case, a window allows to choose an ARCAD list of components to proceed.

Follow the subsequent steps to import from an ARCAD list:

Step 1   In the **List of lists to document** tab of the documentation editor, right-click in the list section and select 📊 **Import from a list** in the contextual menu.

Step 2   Select the list to document. Navigate in the **Lists** node of the connected server in the dialog and click the desired list. You can document several ARCAD lists of the application but you have to select one after the other.

Click **Next >** to continue.

Step 3   Select the merge options.

Tick to **Merge this imported list with the existing list**.

> Tick to **Sort the entire list after the objects are added**.

> Click **Finish**.

Result   The list of lists is displayed in the editor.

To remove a list from the **List of lists to document**, right-click a list in the editor and select ⊖ **Remove** in the contextual menu.

15.4.4 Customizing the documentation

For the documentation to be complete, you must be able to integrate elements that do not directly belong to the information system. ARCAD Observer enables you to manage user-defined information.

It is possible to add notes or comments at any level of your documentation. While comments are saved with the documentation project in ARCAD Observer only, notes will appear as parts of the final documentation.
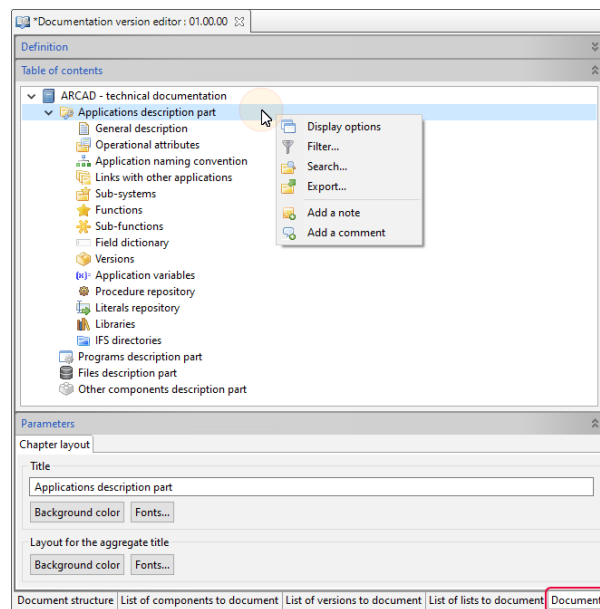
Figure 60: The documentation's customization options

15.4.4.1 Adding a note

Notes are information that are part of the completed documentation. Notes can be user-defined elements or diagrams calculated in the navigator section of ARCAD Observer.

> ⚠️ **Important!**
> Notes can be **text**, **diagrams** or **ARCAD lists**.
>
> The diagrams generated and saved during the analysis can be added to the final documentation through notes.

Follow the subsequent steps to add a note to the documentation.

Step 1 Right-click on a chapter or a section of the document in the table of contents, and select 🔖 **Add a note** in the contextual menu.

The note creation wizard opens.

Step 2 Tick the **Has a title** option to show the note title in the final documentation. Enter a title for the note in the title field.

Step 3 Select the **Note position**. Tick the **Header** button to put the note in the header of the chapter or section, or tick the **Footer** button.

Step 4 Tick the corresponding button to select a **Note type**:

- **Text**: add text and links.
- **Graph**: insert a graph.
- **List**: add an ARCAD list.

Click **Next >** to continue.

Step 5 Define the note according to its type.

Text type

- Enter the text in the text input field.
- Click the 🖼 add image icon and or the 🔗 add hyperlink icon to add external resources to the note.
- In the file selection dialog, select the file you wish to reference. Click **Open**.

Graph type

> ℹ **Note**
>
> The diagrams use in a documentation are duplicates of existing diagrams, which means that modification of the original diagram will not affect the diagram in the note.

- Click the **Diagram** button.
- Select your diagram in the diagram dialog.
- [*Optional*] Select the display **Height** and **Width** of the diagram. Click **OK**.

ARCAD list type

- Select a list in the list manager.

Step 6 Click Finish.

Save the changes (💾, Ctrl+S or **File** > **Save**).

To edit a note, click on the note in the table of contents and make the necessary modifications in the **Parameters** section of the documentation editor.

You can add as many notes as you wish. If you have added several notes to a section or a chapter, you can reorder the notes in the section or chapter. In the table of contents ,right-click on the note and select ⊛ **Move up** or ⊛ **Move down** in the contextual menu.

> ℹ️ **Note**
> Reordering a note is done within the header or the footer.

To delete a note, right-click on the note in the table of contents and select 🗑️ Delete.

15.4.4.2 Adding a comment

Comments are intended to create a dialog between the different operators in documentation generation. They are presented in text form and will not be included in the final documentation.

Follow the subsequent steps to add a comment.

Step 1   Right-click on a chapter or a section of the document in the table of contents, and select 🗨️ **Add a comment** in the contextual menu.

A comment is added in the section or chapter node.

Step 2   Enter a comment in the **Comment** section in the documentation view.

Result   You can consult the comments by clicking the comment entries of the table of contents.

Save the changes (💾, Ctrl+S or **File** > **Save**).

To edit a comment, click on the comment in the table of contents and make the necessary modifications in the **Parameters** section of the documentation editor.

To delete a comment, right-click on the comment in the table of contents and select 🗑️ Delete.

15.4.5 Reviewing the documentation

You can preview your documentation in the documentation version editor.

> ⚠️ **Important!**
> By default, the documentation preview is displayed in the editor, unless you have changed the default viewer in the **Preferences**.

> 📖 *Reference*
> Refer to .

In the Document tab, click on a chapter or a section of the table of contents to display a preview of the documentation.

> ℹ️ **Note**
> This action is similar to generating a part of the documentation and will validate the content and the layout. This can be useful to perform a validation on a sample element rather than on the entire documentation.
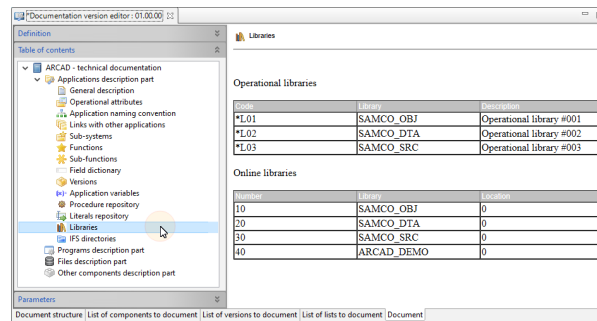
Figure 61: Documentation preview

15.5 Generating documentations

The generation of the documentation does not produce any documents. This is only an update of the ARCAD Database. Generating the documentation is required to export the documentation.

To generate the complete document, right-click on the version in the navigator and select 📘 **Generate all** in the contextual menu.

It is possible generate the complete document from scratch. In this case, the documentation stored in the ARCAD database will be calculated and saved again. To generate the complete document form scratch, right-click on the version in the navigator and select 📕 **Generate all from scratch** in the contextual menu.

A progress dialog is display when the documentation generation job is calculating and when the job is finished.

15.6 Exporting documentations

With ARCAD Observer, you can export your documentation for external consultation or publication in different formats. To do this, the export engine will generate a directory whose structure and data can be transferred to an HTTP server.

> ⚠ **Important!**
> ARCAD Observer does not offer site deployment features. This task will belong to the company's web site management department.

When the documentation is generated, it can be exported. There is three ways to export documentation:

- export to site,
- export to file, or
- export to Confluence.

Export in multiple web-pages is useful to upload files on a remote site.

> ⓘ **Note**
> If you have defined external references in your documentation, ARCAD Observer is able to identify them and make a copy in the export directory.

15.6.1 Export to site

You can export the documentation to multiple files HTML format.

Step 1 To export the documentation version, right-click on the version in the navigator and select
 Export to site in the contextual menu.

Step 2 Click the Browse icon to select the **Export location**.

Define the directory in which the site structure will be saved. In the export location dialog, you can navigate to the desired directory, or create a new one if needed.

Step 3 Tick the **Include section** option to include all the sections to the side navigation of the exported documentation. Otherwise, only chapter entries are included in the navigation.

Step 4 Click **Next >** to continue.

Step 5 Define the layout of the final documentation.

Header setup	**Height**: select the height, in pixels, of the header section of the final documentation.
	Background Color and **Font**: select the background color of the header and the format (the font, color and size) of the header's text.
Background image	**Set an image in the background**: tick this option if you want to use an image as background instead.
	Click the Browse icon to select a background image from your file system.
	Transparent: tick this option to make the background image transparent.
	Image style: tick the **Mosaic** option to repeat the background image across the header or tick the **Logo** option to use the image as a logo, situated at the top left corner of the header.
Footer setup	**Title**: enter the title displayed in the footer.
	Height: select the height, in pixels, of the footer section of the final documentation.
	Background Color and **Font**: select the background color of the footer and the format (the font, color and size) of the footer text.

Step 6 Click **Finish**.

Result The documentation is exported to multiple web pages. You can open it in any web browser.

15.6.2 Export to file

You can export the documentation to a single file HTML format, that contains all of your documentation.

> ⚠ **Important!**
> As pictures cannot be stored in an HTML file, they are stored in a directory.
> They will be referenced using hypertext links in the export file.

Step 1 To export the document version, right-click on the version in the navigator and select 📄 **Export file** in the contextual menu.

Step 2 Click the ⊡ Browse icon to select the **Export location**.

Define the directory in which the file will be saved. In the export location dialog, you can navigate to the desired directory, or create a new one if needed.

Step 3 Tick the **Include section** option to include all the sections to the table of contents of the exported documentation. Otherwise, only chapter entries will be included in the table of contents.

Step 4 Enter a **File name**. Indicate the name of the HTML file that will contain your documentation.

Step 5 Click **Finish**.

Result The documentation is exported. You can open it in any web browser.

15.6.3 Export to Confluence

You can export the documentation to a defined Confluence space, that contains all of your documentation.

> 📑 *Reference*
> To set the connection to your Confluence space, set the preferences described in the .

Follow the subsequent steps to export your documentation to Confluence.

Step 1 In the **ARCAD Observer** navigator, open the desired documentation project.

Step 2 Right-click on the version of the documentation that needs to be exported and click the 📄 **Export to Confluence** button.

The Export to Confluence wizard opens.

Step 3 In the Export to Confluence wizard, define the space where to export the documentation in the **Confluence space key** field.

> 📑 *Reference*
> To retrieve the Confluence space key, go to a designed Confluence space, click on the **Space tools** button and then on **Overview**. The key field contains the Space key.

Step 4 In the Export to Confluence wizard, fill the **Documentation home page title** field.

> ⚠️ **Important!**
> This documentation title must be unique in Confluence. Click the **Check title in Confluence** to make sure the title does not exist already in the Confluence space.
> If you export to Confluence with a document title that already exists in the space, an error message is displayed.

> **ⓘ Note**
>
> If you test the uniqueness of the document title and it already exists, the wizard then displays a **Delete Confluence document** button, that allows you to delete the existing documentation in Confluence. The deleting process can take a few minutes, depending on the number of pages.
>
> If you wish to keep the existing documentation in Confluence, you can modify the title in the **Documentation home page title** and test it again.

Result   The documentation is successfully exported and you can access the pages in your Confluence space.

15.7 Closing documentation versions

Closing a documentation version consists in declaring all phases of documentation generation to be complete. Once the version is closed, no modification of any kind can be made.

Follow the subsequent steps to close your version.

Step 1   In the **ARCAD Observer** navigator, open the documentation project.

Step 2   Right-click on the documentation version and select 🖼 **Close** in the contextual menu.

Step 3   Click **OK** to validate closure or **Cancel** to cancel the operation.

Result   The documentation version is closed. The closed documentation version is displayed with a lock.

15.8 Updating a documentation

Updating a documentation means recalculating obsolete information. Information can be considered obsolete when:

- Changes are made on components (cross-reference, calling chain ...).
- New components are added in the repository or a version.
- Components are deleted in the repository, or a version.

Within an ARCAD version of an application, this information is made available and is used to complete the update operation.

To update a documentation, duplicate the documentation version using the options to recover the **component list**, the **version list** and the **list of lists**; and also the option to recover the documentation contents. Then, generate the documentation.

> **▶ Reference**
>
> Refer to.
>
> Refer to.

15.9 Deleting documentation versions

> ⓘ *Warning!*
> Deleted documentation versions cannot be accessed or recovered

To delete a documentation version, right-click on the item in the application navigator and select 📖 Delete.
Click **OK** to confirm or click **Cancel** to keep the documentation version in the dialog.

Appendices

The AUPDPFKEY command

The AUPDPFKEY command allows you to generate the primary keys in the ARCAD Repository automatically. You can generate keys for one file, for all application files, or for all files in a list.

Primary keys come from the files with these characteristics:

- A unique key on the physical file.
- A unique key on one of the non-joined logical files built over this physical file.
- A non-unique key specified on the physical file.
- A non-unique key specified on one of the non-joined logical files built over this physical file.

> ℹ **Note**
>
> The logical files are analyzed in alphabetical order, and numbers are processed before letters (files named XXXXL1 and XXXXL2 processed before XXXXLA and XXXXLB).

The following table contains:

- A link to a complete description of each parameter available in the AUPDPFKEY command.
- The values allowed for each parameters. If a parameter has a default value, it is indicated in **bold**.

Options	Values
File(s) (FILE)	*ALL,*LST, *VERS
Application ID (APPID)	***CURENV**
Version level (VERSLVL)	***LASTPRD**, *CURENV
List (LIST)	*CURENV, *LIBL, *CURLIB, Library name

Table 7: Summary of the AUPDPFKEY options

File(s) (FILE)

Identify the file(s) for which you want to calculate the primary keys.

Alpha value	Enter the name of a file to process. This file must belong to the application and version you have already specified.
*ALL	The command will process all the files in the application.
*LST	The files to process are in the list specified in List (LIST) parameter.
*VERS	The command processes all the files that belong to the version.

Application ID (APPID)

Identify the application to process.

*CURENV	The current environment's application.
Alpha value	Enter an application code.

Version level (VERSLVL)

Identify the version level to process.

*LASTPRD	The last version transferred to product
*CURENV	Use the current version's identifier.
Alpha value	Enter the specific version to process.

List (LIST)

Enter the name of an existing list.

*CURENV	The object is searched for in the current environment. Run the ADSPCURENV command to identify this environment.
*LIBL	The object is searched for in the list of libraries.
*CURLIB	The object is searched for in the current library for the job. If no *curlib was specified, QGPL is used by default.
Library name	Enter the name of the library containing the object.

The AUPDFLDDBR command

The AUPDFLDDBR command allows you to generate database links at the field level. Once generated, the links can be displayed and used via the ARCAD Observer perspective in the ARCAD Client RCP or the plug-in in RDi.

The generation of the database links using the AUPDFLDDBR command proceeds as follows:

1. The command links the fields of the different physical files using:
   - the references to the fields of the common reference file, or
   - the codification of the field name.

2. The command searches for and memorizes the primary keys (or principal identifiers) in each file. These primary keys are taken from the first file that meets the following conditions:
   a. The physical file has a unique key.
   b. One of the non-joined logical files that depends on the physical file has a unique key.
   c. The physical file has a specified key.
   d. One of the non-joined logical files that depends on the physical file has a specified key.

3. The command creates links between the fields of the two physical files if all the primary keys of one of the files are found in the other file.

The logical files are processed in alphabetical order. However, the numbers are processed before the letters.

> **Example**
> The XXXXL1 and XXXXL2 files are processed before the XXXXLA and XXXXLB files.

> **Note**
> The first time you run this command, the primary keys are memorized (accessible from the field repository) and you can modify them. Then, when you re-run the AUPDFLDDBR command, it will take into account the primary keys that you have modified.

The following table contains:

- A link to a complete description of each parameter available in the AUPDFLDDBR command.
- The values allowed for each parameters. If a parameter has a default value, it is indicated in **bold**.

Options	Values
Application ID (APPID)	***CURENV**
Component display level (VERSLVL)	***LASTPRD**, *LAST, *CURENV
Replace or add record (OPTION)	***ADD**, *REPLACE
With the reference files (USEFLDREF)	***YES**, *NO
Use field naming (USEFLDNAME)	***NO**, *YES

Table 8: Summary of the AUPDFLDDBR options

Options	Values
Use repository constraints (USEREFCST)	***YES**, *NO
Use program analysis (USEPGMPRP)	***NO**, *YES
List of reference files (REFFILLIST)	***APP**, list-name = ***LIBL**, *CURLIB, library-name
List of physical files (PFFILLIST)	***FLDDCT**, list-name = ***LIBL**, *CURLIB, library-name
List of propagation monitoring (DRVFLDLIST)	***CURENV**, *LIBL, *CURLIB, library-name
Substitute the field names (FLDNAMRPL)	up to 50 substitutes
Search words to be known (FLDNAMCOD)	up to 20 words
Fields or codes to omit (FLDOMT)	***NONE**, field/code-name
% Fields without prefix/suffix (PRXSFX)	percentage
With inter application links (USEAPPLNK)	***USE**, *ALL, *NONE

Table 8: Summary of the AUPDFLDDBR options

Application ID (APPID)

This parameter allows you to identify the application to process.

*CURENV	The current environment's application.
Alpha-Value	Enter the ID code of the application.

Component display level (VERSLVL)

This parameter allows you to display the repository at a different level than the current level.

The possible values are as follows:

*LASTPRD	Enter this value to display the repository at the last transfer to production level.
*LAST	Enter this value to display the repository at the last version that includes the literal.
*CURENV	Enter this value to display the repository at the current version level.
Alpha-Value	Enter the version number level for which you want to have the repository display.

Replace or add record (OPTION)

Specifies if this process should add the new links or totally regenerate them.

*ADD	Only the new links are added. The old links, which have already been defined, are neither changed nor deleted.
*REPLACE	All the defined database links of this application are deleted; they are then regenerated.

With the reference files (USEFLDREF)

This parameter allows you to indicate if the method usedto use to generate the database links is based on the references to the common reference file fields.

*YES	All fields defined by external references relating to the same reference file field will be matched for the establishment of database links. Fields that do not have a reference for a reference file field will be linked in relation to their complete name.
*NO	In relation to a reference file, external field references are not taken into account.

> ⓘ **Note**
>
> If the 2 methods (USEFLDREF and USEFLDNAME) for linking fields are chosen, it is the one based on the external references to the common reference file fields which takes precedence over the method based on the codification of field names.

Use field naming (USEFLDNAME)

Specify if the method used to generate the database links is based on the codification of field names.

*YES	The physical file field names are assumed to adhere to the following codification: • a prefix (or suffix) of one or more characters, identical for all fields in the physical file. • the remainder of the field name corresponds to an identifier: all fields with the same type, length, number of decimals, and identifier are matched to define the database links.
*NO	The codification of the field names is ignored.

Use repository constraints (USEREFCST)

Specify if the method used to generate the database links takes Referential Constraints into consideration.

*YES	Referential Constraints are used to determine database links.
*NO	Referential Constraints are ignored.

Use program analysis (USEPGMPRP)

Specify if the method used to generate database links is based on the derived field list created by an ARCAD-Transformer propagation process. See the ACVTPGMFLD command which creates a list of derived fields (specified in the DRVFLDLIST parameter) that can be used by this command.

> ⮕ *Reference*
>
> See the OBSDBRXRF macro command, for an example of the use of this parameter (after the ACVTPGMFLD command).
>
> Refer to the OBSDBRXRF macro-command documentation.

*YES	There are derived fields, processed by the ARCAD-Transformer command ACVTPGMFLD, to take into account. The field list is specified in the DRVFLDLIST parameter.
*NO	ARCAD-Transformer is not used.

List of reference files (REFFILLIST)

Indicate here, the (source or object) list name that corresponds to the application's reference files.

No database link will be established on these reference files, they will be ignored.

*APP	The reference files are those defined on the operational attributes level of the application.
list-name	Enter the name of a list containing the reference files to be omitted.

The possible values for the name of the library are as follows:

*LIBL	The object is searched for in the list of libraries.
*CURLIB	The specified object is searched for in the current library for the job.

	If no *CURLIB was specified, QGPL is used by default.
Library name	Enter the name of the library containing the object.

List of physical files (PFFILLIST)

Indicate here the (source or object) list that corresponds to all the physical files for which you must analyze the database links at the field level.

These fields must belong to the application to be processed and their cross-references must be updated.

*FLDDCT	All the physical files defined in the application and present in the field repository are processed (except for the reference files).
list-name	Enter the name of a list containing the physical files to be processed.

The possible values for the name of the library are as follows:

*LIBL	The object is searched for in the list of libraries.
*CURLIB	The specified object is searched for in the current library for the job. If no *CURLIB was specified, QGPL is used by default.
Library name	Enter the name of the library containing the object.

List of propagation monitoring (DRVFLDLIST)

Indicate the name and library of the list file where the propagation process will list the derived fields. This file will contain one record per field/program combination.

*CURENV	The object is searched for in the current environment. (Run the ADSPCURENV command to identify this environment).
*LIBL	The object is searched for in the list of libraries.
*CURLIB	The object is searched for in the current library for the job. If no *CURLIB was specified, QGPL is used by default.
Library name	Enter the name of the library containing the object.

The *OBJ special value is reserved for the propagation in mode *ALLFLD, or *RNMOBJ in order to give this list the same name as the referenced object.

> ⓘ **Note**
> The name of this file (with its library) identifies the propagation. This is the file name that you will specify in the ACLRDRVFLD command to clear the data that a propagation generates in ARCAD files.

Substitute the field names (FLDNAMRPL)

This parameter is only operational if you use the matching method according to the codification of field names (Use field naming (USEFLDNAME) is set to *YES).

It allows you to indicate from 0 to 50 substitutions on the complete or partial ("identifiers") field names to compensate a codification of field names that is not always normalized.

Search words to be known (FLDNAMCOD)

This parameter is only operational if you use the matching method according to the codification of field names (Use field naming (USEFLDNAME) is set to*YES).

It allows you to indicate from 0 to 20 "words" that would have been used as identifiers for the main key field of the file.

> 🚩 *Example*
>
> For the Customer (prefix CU) and Supplier (SU) files, the customer code field name is CUCODE and the supplier code is SUCODE, while they should have been CUCUS and SUSUP (CUS and SUP being the Customer and Supplier identifiers).
>
> In this case, specify that CODE is a word to be recognized.

> ℹ️ **Note**
>
> This substitution method only works if there are other files where the field names begin with the prefix (or suffix) of the file and have similar text (although not necessarily identical).

Fields or codes to omit (FLDOMT)

This parameter is operational whatever the field matching method you are using.

It allows you to indicate from 0 to 50 fields, with or without prefixes (or suffixes), that should not be taken into account during the matching process.

*NONE	None of the fields or codes are omitted.
Alpha-Value	Enter the text to omit.

% Fields without prefix/suffix (PRXSFX)

This parameter defines the percentage of fields without prefixes (or without suffixes) accepted by the process.

> 🚩 *Example*
>
> 0% means that all fields should have the same prefix (or suffix), in order for it to be taken into account.
>
> 20% means that one field out of five can have a different prefix (or suffix) than the other fields. This tolerance allows you to keep the prefix (or suffix) previously detected.

With inter application links (USEAPPLNK)

Specify whether or not to use inter-application links for cross-references. The list of applications linked to the current application is obtained by examining the links defined in the application's descriptive parameters (found using the ADSPAPPINT command).

*USE	The inter-application links are used to simulate the real use context for components. The link is made towards the applications used or with a reciprocal link. In the case of a component homonym, only the first one found is taken.
*ALL	The inter-application links are used to determine all the applications concerned for a component. The link is made in all directions (applications used by or using the current application or reciprocal link). In the case of a component homonym, all the applications of this component are displayed, in the order defined in the inter-application links.
*NONE	The inter-application links are not used. Only the components present in the interrogating application are known.

Troubleshooting

Compatibility with the ARCAD Server

ARCAD RCP client applications, such as ARCAD Verifier or ARCAD Client, include a consistency version control. This control is based on the file *compliant.xml*.

You may have trouble connecting an RCP to the server after it is upgraded from one version to the next. The blocking message in red below may appear when you attempt to connect to the server.
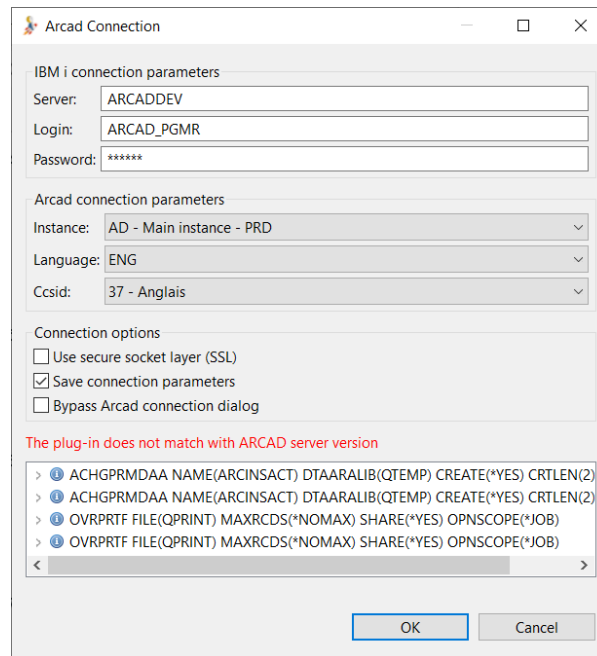

Figure 62: Server compatibility error

If you receive this error, install a new version of the RCP application, if a new version is available.

If no updated version of the RCP application has been released, and after carefully checking the compatibility aspects in the release notes, update the *compliant.xml* file as described below.

Updating the compliant.xml

> ⚠ *Warning!*
> This procedure should only be done in emergency situations. It is not recommended to use this as a permanent solution to server/studio compatibility problems.

The *compliant.xml* file is located in the following directory:

<installation>/plugins/com.arcadsoftware.core_<version>/compliant.xml.

Where **<installation>** is the RCP application installation directory and **<version>** is the highest version level in the plug-in directory.

Follow the subsequent steps to update the *compliant.xml* file.

Step 1 Check the existing version of the server using the following command (if on IBM i):
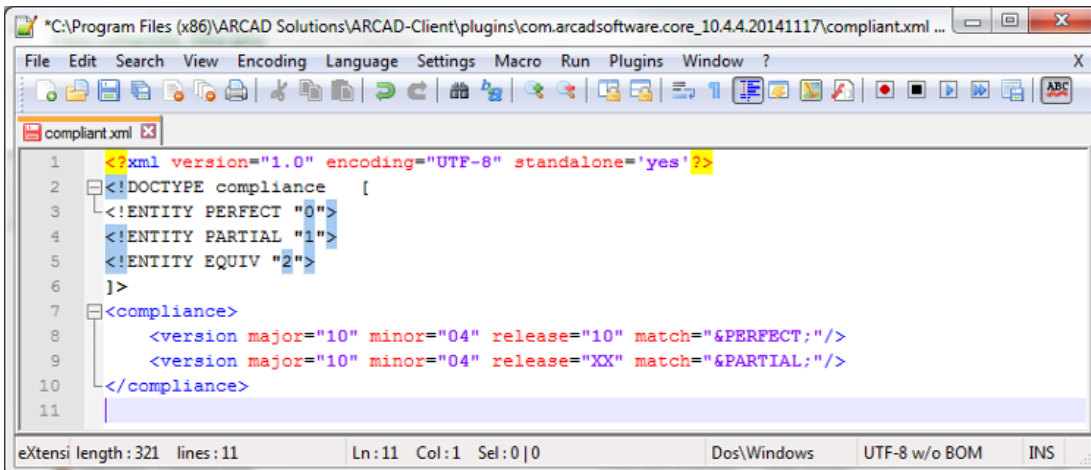
dspdtaara arcad_prd/arcversion

Step 2 Open the *compliant.xml* file for modification.

Step 3 Add a line based on one of the following examples:

- To be able to connect to a server running ARCAD version "aa.bb.cc":
  <version major="aa" minor="bb" release="cc" match="&PERFECT;"/>

- To be able to connect to a server running ARCAD aa.bb.whatever the release number:
  <version major="aa" minor="bb" release="XX" match="&PARTIAL;"/>

In this last case, a warning message will inform you that compatibility with the ARCAD version on the server is only PARTIAL.

Figure 63: Editing the compliant.xml

Glossary

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

A

application

The highest entities ARCAD can work with. On the IBM i server running ARCAD, an application corresponds to a set of libraries, referred as application operational libraries.

ARCAD identifies an application through its 3-character identifier, also called an "application code" in the user interface.

ARCAD Repository

For an application, the repository corresponds to the list of all the components. For each component, it includes the following:

- its physical location (source, object),
- its position in the system function tree, and
- its cross references.

This object repository allows ARCAD to know the detailed content of an application.

B

C

CCSID

Coded Character Set Identifier. Specific to IBM i. A 16-bit number that includes a specific set of encoding scheme identifiers, character set identifiers, code page identifiers, and other information that uniquely identifies the coded graphic-character representation.

component

Most of the components of an ARCAD application will consist of source and a corresponding object. The ARCAD Repository also contains:

- sources without object components (COPY members, etc.)
- objects without source components (*DTAARA, *JOBD, etc.)
- IFS components
- data-type components (a.k.a DTC)

> ⓘ **Note**
> Compiling a source member creates an object of the same name.

component repository

A database list of all the components in an application. It contains some information (topology, functional description, system function tree etc.) used during the development stages for automation and reliability.

cross-reference

Cross-references display links between elements in the repository.

- Components
    - Links between programs (called and calling programs)
    - Programs using files
    - Files used by programs
    - Links between physical and logical files

- Fields
    - Components using fields
    - Fields used by components

- Sources
    - Links between source members (/COPY)
    - Manual links between a component created by copy of an existing component
    - Links between 4GL/3GL components (e.g. ADELIA or SYNON)

- Literals used in components
- Procedures
    - Procedures used in ILE programs
    - Programs using procedures

- ID messages (from MSGF)
    - Messages sent by components
    - Components using messages

D

E

F

Field Repository

A database list of all the fields defined in the physical files (and/or tables) of an application. It is very useful for the impact analysis performed before modifications are made.

G

H

I

J

K

L

Literal Repository

A database list of all the literals defined in an application's components.

M

N

O

P

Procedure Repository

A database list of all the procedures in an application.

Q

qualified name

A fully qualified name for an IBM i object is its absolute path (the library in which it is found and a slash).

> **Example**
> <library>/<name>

R

S

T

U

V

W

X

Y

Z