User Manual

MDOpen

Eclipse based plug-in for Change and Distribution Management from Midrange Dynamics

Version 8.2.5
Published November 18, 2020

MDOpen - Table of Contents

1 OVERVIEW OF THE MDOPEN MANUAL 6
2 INSTALLING MDOPEN 6
2.1 General MDOpen Client Information 6
2.2 General MDOpen Server Information 6
2.4 Prerequisites 7
2.5 Installing a Specific Version of MDOpen for the First Time 8
2.5.1 Installing from the Cloud 8
2.5.2 Installing from a Zip File 8
2.6 Updating an Installed Version of MDOpen 9
2.7 Uninstall MDOpen 9
2.8 The MDOpen Perspective 9
3 MDCMS REPOSITORY CONNECTIONS 10
3.1 Adding an MDCMS Repository 10
3.2 Change User or Password for Repository Connection 11
3.3 Export/Import Repository Definitions 12
3.4 Using Remote Systems Explorer Toolset in MDCMS perspective 12
4 STARTING MDOPEN 13
4.1 Common Handling Features in MDOpen 13
4.2 MDOpen Preferences 16
5 MDCMS SETTINGS WITHIN MDOPEN 18
5.1 Common Settings Options 18
5.2 App Hierarchy 18
5.3 Applications 18
5.4 Levels 18
5.4.1 Process Flow Report 19
5.5 Deployment Flow Diagram 21
5.6 Attributes 21
5.7 Attribute Commands 22
5.8 Attribute Scripts 23
5.9 Object Commands 25
5.10 Object Scripts 25
5.11 System i Settings 26
5.12 OS/400 Locations 27
5.13 OS/400 Location Groups 27
5.14 Distribution Levels 27
5.15 Remote Server Locations 28
5.16 Services 30
5.17 Git Repositories 31
5.18 SVN Repositories 33
5.19 External Reference Levels 34
5.20 External Reference File Extensions 35
5.21 Continuous Integrations 36
5.22 eMail Bodies 38
5.23 Script Templates 38
5.24 Project Phases 38
5.25 Project Types 39
5.26 Task Types 39
5.27 User Group Types 40
5.28 User Groups 40
6 MDWORKFLOW SETTINGS WITHIN MDOPEN 41
6.1 Acceptance Group Types per Level 41
6.2 Project/Task Status Codes 41
6.3 Project/Task Status Boundaries 42
6.4 Project/Task Status Triggers 42

6.5 *Custom Fields* 43
6.5.1 DDL Entries 44
6.6 *WebApp Config xml* 44
7 REPOSITORY SESSION FUNCTIONS 45
7.1 *Change password* 45
7.2 *Show locked objects* 45
7.3 *Preferences* 45
7.4 *Close editors* 45
7.5 *Report output* 45
7.6 *Spooled output* 46
8 MDXREF 47
8.1 *MDXREF Objects* 47
8.1.1 MDXREF Object Details 48
8.1.2 Objects using File 49
8.1.3 Objects Using File (Inline/Full) 49
8.1.4 Files Used 50
8.1.5 Files Used (Inline/Full) 50
8.1.6 Fields in File 51
8.1.7 Keys in File 51
8.1.8 Join Fields in File 52
8.1.9 CPYF/OVRDBF Information 52
8.1.10 Journal Analysis 53
8.1.11 Up 54
8.1.12 Up Inline / Full Call Stack Up 54
8.1.13 Down 55
8.1.14 Down Inline / Full Call Stack Down 55
8.1.15 Copybooks Used 56
8.1.16 Exported Procedures 56
8.1.17 Imported Procedures 56
8.1.18 Modules Used 57
8.1.19 Objects using Module 57
8.1.20 Process Flow Report 58
8.1.21 Print Relational Dependencies 58
8.1.22 View Source 59
8.1.23 XREF Refresh for Objects 59
8.1.24 Debug – set service entry point 59
8.1.25 X-Analysis Options 59
8.2 *MDXREF Source* 60
8.2.1 Source containing Field 61
8.2.2 Source Compare 61
8.2.3 Programs using Source 61
8.2.4 View Source 61
8.3 *MDXREF Procedures* 62
8.4 *MDXREF IFS* 63
8.5 *MDXREF External References* 64
8.6 *MDXREF Search for Fields* 65
9 OBJECT REQUESTS 66
9.1 *Overview* 66
9.2 *Object Request View* 66
9.3 *Object View Detail* 70
9.4 *Object View Options* 74
10 REQUEST OBJECTS DIRECTLY ON IBMI 79
10.1 *Modify Existing Object* 80
10.2 *Add New Object* 82
10.3 *Recompile Object* 82
10.4 *Delete Object* 82
10.5 *Update Object* 82

10.6 Request Related Objects 83
10.7 Request Linked Objects for Attribute 85
10.8 Bound ILE Objects 86
10.9 Version Conflict Resolution 87
10.10 Import Source 88
10.10.1 Source Import Definition 88
10.10.2 Source Import Result List 89
10.11 Import Library 90
10.12 Request Objects from MDXREF 91
10.13 Request Source Members from MDXREF 92
10.14 Request IFS Objects from MDXREF 93
10.15 Request IFS Source from MDXREF 94
10.16 Import Objects or Source from IFS Folder 95
11 REQUEST OBJECTS FROM OTHER LOCATIONS 96
11.1 Request Objects to/from Workspace 97
11.1.1 Create an Ignore List for Requests and Commits from Local Workspace 98
11.2 Import Objects from Local Folders 99
11.3 Import Objects from Remote Server 99
11.4 Request Objects from the Git Perspective 100
11.5 Request Objects from the SVN Repository Exploring Perspective 101
11.6 Request – Initial Dialog 102
11.7 Request - Selection Confirmation for Objects 104
11.8 Request - Selection Confirmation for Source 105
12 COMPARE OBJECT SOURCE 106
13 DATA TRANSFORMATION 108
14 OBJECT HISTORY 110
15 PROJECTS 111
15.1 Project Listing 111
15.2 Involved Groups for Project 112
15.3 Time Entry Listing 112
15.4 Time Entry Details 113
16 TASKS 114
16.1 Task Listing 114
16.2 Subtask Listing 115
17 RFP 116
17.1 RFP Listing 116
17.2 RFP Details 118
17.3 Copy RFP 119
17.4 Merge Multiple RFPs into One RFP 119
17.5 Reset Status of RFP 119
17.6 Close RFP 119
17.7 Create RFP Objects into Developer Library 119
17.8 List Spooled Files for an RFP 120
17.9 View the RFP Deployment Log 120
17.10 List Projects for an RFP 120
17.11 List Tasks for an RFP 120
17.12 List Object Requests for an RFP 120
17.13 Adding a New Object Request to RFP 120
17.14 RFP Commands 121
17.15 RFP Scripts 122
17.16 Compare RFP with Branch 123
17.17 Promoting an RFP 125
17.18 The Batch Installation Process 126
17.18.1 The Source/Object Preparation Steps 126
17.18.2 Source and Object Signing 126
17.19 The Installation Steps 126
17.20 The Archiving and Cleanup Steps 127

17.21 The Set for Next Level Steps 127
17.22 Approving a Promotion 128
17.23 Launching MDRapid for a Promotion 128
17.24 Installing a Promotion 130
18 RFP INSTALLATION HISTORY 131
18.1 Copy RFP 131
18.2 Rollback RFP 132
18.3 Acceptance Test Status 133
18.4 Objects in Installed RFP 135
19 RFP SEND LISTING 136
19.1 Send RFP to Target Systems 137
19.2 Add RFP to Send Listing 139
19.3 Add Object to Send RFP 141
19.4 Merge Multiple RFPs into One RFP 141
19.5 Send RFP Commands 142
19.6 Send RFP Scripts 142
19.7 Objects in Send RFP 142
20 RFP RECEIVE LISTING 143
20.1 RFP Receive Listing Options 144
21 RFP RECEIVE HISTORY 146
21.1 RFP Receive History Detail 147
21.2 RFP Receive History Options 148

1 Overview of the MDOpen Manual

This manual is a guide for using the MDOpen Eclipse based plug-in for multiplatform development for the Midrange Dynamics Change Management System (MDCMS).

This manual will refer to the core system as MDCMS. MDCMS is a tool that helps manage source and objects throughout the development, migration, modification, and distribution life cycle.

Additional configuration steps pertaining to the core MDCMS product are outlined in the ***MDCMS User Manual*** and will not be covered within this manual.

2 Installing MDOpen

2.1 General MDOpen Client Information

MDOpen consists of a java plug-in for use within an Eclipse-based IDE such as Eclipse, MyEclipse, Rational Developer for i or Zend Studio.

The plug-in provides an Eclipse Perspective named MDCMS. Within the MDCMS Perspective, MDCMS functions are provided within specific views for each function.

Additionally, the MDOpen plug-in provides Context Menu options from within other Eclipse Perspectives for the seamless management of objects by MDCMS from the programmer's preferred development perspective, Git perspective, SVN perspective or X-Analysis.

MDOpen provides nearly every developer function that is available within the MDCMS emulator client. Additionally, MDOpen is required when managing non-native objects and is much more convenient when managing IFS objects.

2.2 General MDOpen Server Information

MDOpen connects to MDCMS database libraries that reside on 1 or more IBMi partitions. All MDCMS information is retrieved from, and stored to, these libraries, so that all MDCMS interfaces (MDOpen, MDWorkflow or emulator) use the same database.

When a connection is established with an IBM i partition, a job is submitted under the user profile in the connection definition to the job queue defined within the MD Job Settings (option 12 from the MDCMS Setup Menu) for property MDOpen/MDWorkflow Job Queue.

The default CCSID of the job is defined in the system settings (option 11 from the MDCMS Setup Menu). This can be overridden for a specific user in the MDSEC user definition.
If issues are encountered with source editing or compiling due to incorrect characters, it is likely because of the CCSID value. In which case, change it to the value used for the interactive jobs of your developers.

The name of the job will be MDRPnnnnnn. The name and CCSID of a specific connection is visible in the Repository Location parameters from within the MDCMS eclipse perspective.

2.4 Prerequisites

- Java JDK 7 or newer (with Java EE)
- Eclipse based IDE
- 1 or more IBMi systems with MDCMS installed
- A valid MDOpen License Key for each IBMi system. If you are unsure of your License Key status you will need to check with your systems administrator or the person who installed and licensed the MDCMS product.
- MDCMS must be on at least version 7.2 to use MDOpen. It is recommended to be on at least version 8.0.
- Unless routing through a Proxy server, the firewall must allow access to the following ports on the IBM i: 446, 449, 8470-8476. If already successfully using Rational Developer for i, these ports will already be available.

2.5 Installing a Specific Version of MDOpen for the First Time

The MDOpen version release must match the version release of MDCMS, for example MDOpen 8.2 must equal MDCMS 8.2.

Check the downloads page for compatibility in case a point release of MDCMS requires a different release of MDOpen. For example, MDCMS 8.2.5-8.2.6 requires MDOpen 8.2.5 instead of MDOpen 8.2.
Each version of MDOpen that has different compatibility rules for MDCMS will have its own update site to avoid a mismatch.

2.5.1 Installing from the Cloud

Take the following steps within Eclipse:

1) Select Menu option Help/Install New Software…
2) Click Add…
3) Enter the text MDOpen v.r into the Name field
   where v.r is the MDCMS version release such as 8.2
4) Enter the URL for the MDCMS version release into the Location field. The URL format is
   http://mdopen.midrangedynamics.com/update/v.r on the US server and
   http://europe.mdcms.ch/update/v.r on the Europe server
   where v.r is the MDCMS version release such as 8.2
5) Click OK
6) If the URL is not already displayed in the Work with drop down list, select it from the list
7) Select MdOpen and click Next
8) Click Next from the Review list
9) Accept the terms of the license agreement and click Finish.
   Eclipse then downloads the newest package
10) Probably, a Security Warning will be displayed indicating that the software contains unsigned content. Click OK to continue the update.
11) Restart Eclipse

2.5.2 Installing from a Zip File

Take the following steps within Eclipse:

1) Select Menu option Help/Install New Software…
2) Click Add…
3) Enter the text MDOpen v.r into the Name field where v.r is the MDCMS version release such as 8.1
4) Click the Archive… button and browse to the location of the zip file containing the update site. If you don’t yet have the update site downloaded, it can be downloaded from the Downloads section of the MD Service Desk (https://support.mdcms.ch).
5) Click OK
6) If the File is not already displayed in the Work with drop down list, select it from the list
7) Select MdOpen and click Next
8) Click Next from the Review list
9) Accept the terms of the license agreement and click Finish.
   Eclipse then downloads the newest package
10) Probably, a Security Warning will be displayed indicating that the software contains unsigned content. Click OK to continue the update.
11) Restart Eclipse

2.6 Updating an Installed Version of MDOpen

MDOpen may be updated for the same MDCMS version in order to correct problems or add additional minor features. To check for and update MDOpen to a newer build of the same MDCMS version/release, take the following steps within Eclipse:

1) If the MDOpen software site was installed from a zip file, a newer version of the zip file must replace the existing zip file, or the location of the zip file must be changed to point to the new zip file.
2) Select Menu option Help/Check for Updates
3) If MdOpen is listed, ensure it is selected and click Next
4) Click Next from the Review list
5) Accept the terms of the license agreement and click Finish.
   Eclipse then downloads the newest package
6) Probably, a Security Warning will be displayed indicating that the software contains unsigned content. Click OK to continue the update.
7) Restart Eclipse
8) If not already open, open the MDCMS Perspective (Window => Open Perspective => Other => MDCMS)
9) Reset the MDCMS Perspective by Right-Clicking on the MDCMS Perspective button in the upper-right corner of the IDE and selecting option Reset

2.7 Uninstall MDOpen

If MDOpen is no longer necessary, it can be uninstalled within Eclipse using the following steps:

1) Close the MDCMS Perspective by Right-Clicking on the MDCMS Perspective button in the upper-right corner of the IDE and selecting option Close
2) Select Menu option Help/About Eclipse
3) Click Installation Details
4) Left-Click MdOpen. If MdOpen is not in the list, then it may be a version older than 7.2. In this case, manually delete the jar file from the plugins or dropins folders.
5) Click Uninstall…
6) Click Finish

2.8 The MDOpen Perspective

1. Start your Eclipse or Rational IDE
2. If not already open, open the MDCMS Perspective (Window => Open Perspective => Other => MDCMS)
3. If not already shown, display the ***MdRepositoryView*** (Window => Show View => Other => Mdcms Main Views => MdRepositoryView)

3 MDCMS Repository Connections

A Repository definition specifies the parameters required to connect to an instance of MDCMS on an IBMi partition. The MDCMS libraries, including MDREP, must be installed on that system and a valid MDOpen license key must exist on that system.

3.1 Adding an MDCMS Repository

To add an MDCMS Repository definition, position your cursor within the ***MdRepositoryView*** and right-click to select ⊞ ***New Repository Location.*** That action will display the ***Add MDCMS Repository*** dialog.

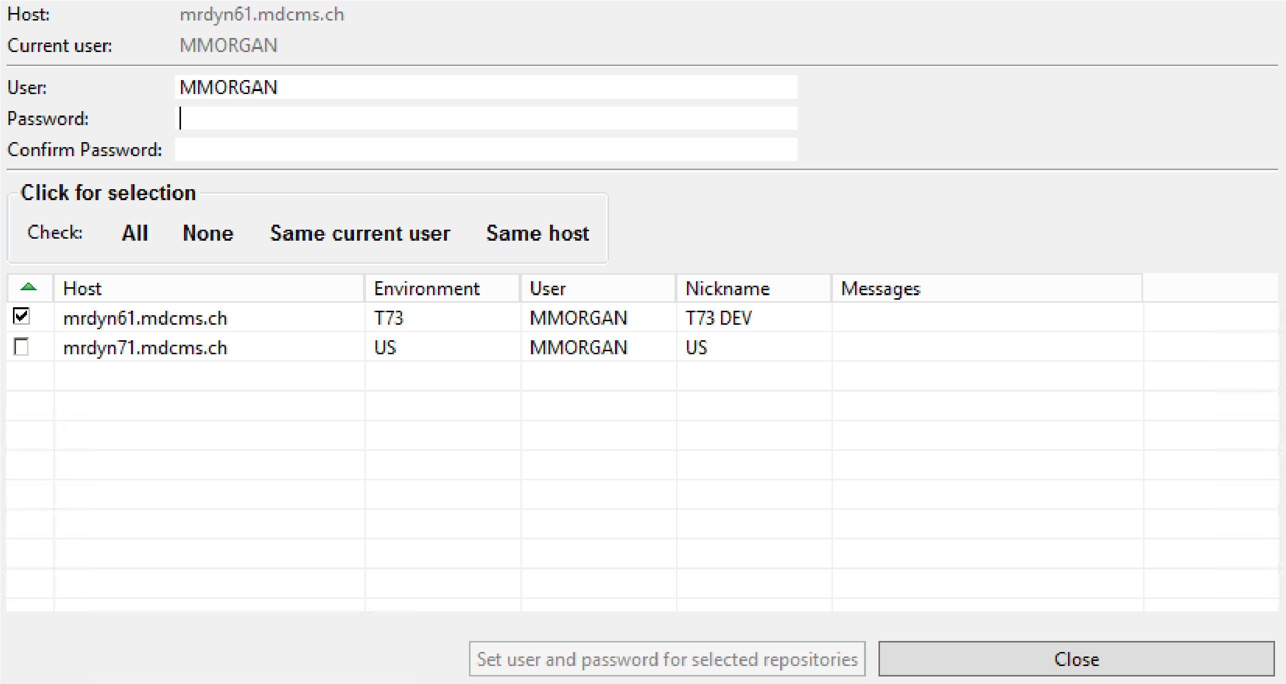
Fill in all required parameters and any optional parameters that are necessary.

The parameters are as follows:

Host	the IBM i host name that is used to connect to that system. If using MDOpen from within Rational Developer for i, it is highly recommended that the Host value matches the connection value in RSE for the partition, so that seamless use of LPEX editors, etc. function correct
Username (required)	a valid user profile on the IBM i
Password (required)	the password for the profile
Port	the JTOpen Signon Port – defaults to 8476
Nickname	a name to help you identify this repository connection when multiple connections are present
Environment	the MDCMS instance ID. Blank indicates the default instance without a suffix for the libraries
Proxy server	the host name of the proxy server, if required to connect to the IBM i
Proxy server port	the port on the proxy server
ASP Group	The ASP device name that MDCMS is installed in. Leave blank if installed to the base system.
Default repository	1 Repository may be designated as the default connection when checking out remote objects from other perspectives/views
Connect on startup	When true, MDOpen will attempt to automatically connect to the repository when the perspective is started.
Connect to X-Analysis	When true, MDOpen will provide the X-Analysis options within the MDXREF view for objects to directly view XA lists and dialogs from the MDCMS perspective. This parameter is only visible if the X-A plugin is installed in the same instance of Eclipse
IBM i Job Name	when a connection is successful, the corresponding job name on the IBM i will be displayed here

3.2 Change User or Password for Repository Connection

If the currently defined user or password for an existing repository definition should be changed, Right-click and select option Set repository user/password. If the option doesn't appear, then the repository is currently connected. In this case, first select option Disconnect and then try again.



Host:	mrdyn61.mdcms.ch
Current user:	MMORGAN
User:	MMORGAN
Password:	
Confirm Password:	

Click for selection

Check: All None Same current user Same host

	Host	Environment	User	Nickname	Messages
☑	mrdyn61.mdcms.ch	T73	MMORGAN	T73 DEV	
□	mrdyn71.mdcms.ch	US	MMORGAN	US	

Enter the user, if it should change, and the password for the user in the single fields at the top of the window. Then, select each repository that the change should be applied to and click button Set user and password for selected repositories.

The changes are only made to the repository definitions. If the password for the user on the target system itself should be made, then use the Session option Change password.

3.3 Export/Import Repository Definitions

Some or all repository definitions can be exported to a file. This file can then be imported into a different workspace.

Export:

1) From the Eclipse Menu, select File/Export…
2) Select MDOpen/Repository connections and click Next
3) Select one or more Repository definitions to be exported. Click Next
4) Save to a file in a folder of your choice and click Finish
   For each selected definition, everything except the password is included.

Import:

1) From the Eclipse Menu, select File/Import…
2) Select MDOpen/Repository connections and click Next
3) Enter or browse to existing connection file. Click Next
4) Enter user id and password to be applied to imported definitions
5) Select one or more Repository definitions to be imported. Click Finish

3.4 Using Remote Systems Explorer Toolset in MDCMS perspective

If using MDOpen as a plug-in within Rational Developer for i, the Remote Systems Explorer LPEX editors can be used to edit source code.
To enable this, a connection in RSE must be defined for the exact same IBM i host name as the name used for the MDOpen repository connection. Then, when MDOpen finds this connection, it will automatically use the LPEX editors. One optional exception to using the LPEX editors is for SQL file types - these can be based on the File Association preferences if the MDOpen preferences parameter "Use File Associations Preferences for SQL editing" is enabled.

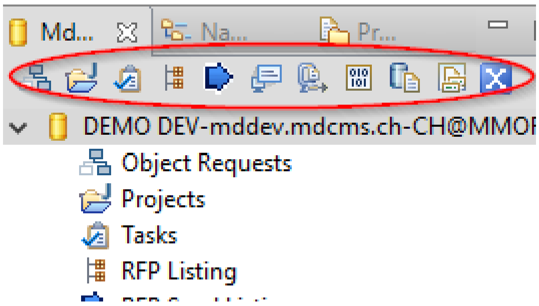
If MDOpen is used within an eclipse build not based on Rational Developer for i, then the File Association preferences will be used for all source types.

4 Starting MDOpen

After completing the creation of one or more Repository locations within your MDOpen installation you can now access the different MDOpen views and functions as follows:

Within the ***MdRepositoryView*** you can expand your connection node using the arrow to the left of the Repository connection to carry out options for that partition. If an expand arrow isn't visible, then right-click on the node and select option Connect.

Some of the most common options for the default repository can also be invoked by clicking the icons above the list of repositories.



4.1 Common Handling Features in MDOpen

Prompting for Valid Field Values

MDOpen uses the **Content Assist *Ctrl + Space*** for the prompting of fields within most editable text fields where a value can then be selected from a list. For certain fields with large lists of information, such as object names or project ids, an initial string of characters can be entered before pressing Ctrl + Space to limit the list to values starting with that string.

Selecting Multiple Rows in a List
To individually select several rows before selecting and option, hold down the Ctrl key and Left-Click each row to select. Left-Click again to unselect one of the rows.
To select a range of rows, hold down the Shift key and Left-Click the first and last row in the range.
To select all rows, hover over the list and press Ctrl + a

Prompting for Valid Options for a List Row

Right-Click on a row (or one of the multi-selected rows) within a list view to select and perform an action for that row or rows.

All defined options for the list are displayed, but only options that are valid for the specific row or rows will be enabled.

Right-Click within the white space of a list view to select to add a new row to that list.

Display/Edit all fields for a Row
Left-Click on a row within a list view to display and, when allowed, edit all fields for that row in a separate editor view. The editor will either replace any prior MDCMS editor or it will open in a new tab, depending on the preference set in Window->Preferences->MDOpen. If a specific editor should be opened differently to the preference, then right-click and select the choice of editor.

Loading Next Frame of Data for a List View

For performance reasons, MDOpen limits the number of list rows collected from MDCMS based on the frame size. The frame size default is 100 rows and can be changed in the MDOpen preferences. A list view indicates if additional rows are available by displaying the icon at the top right of the view. Click on the icon to load the next frame.

Loading Next Frame of Data or all Data for an Object Child List View

For performance reasons, MDOpen limits the number of object request rows for an RFP, Project, Task or Subtask collected from MDCMS based on the frame size. The frame size default is 100 rows and can be changed in the MDOpen preferences. If more object requests exist for the parent, the buttons "Load more rows" and "Load all rows" appear at the bottom of the object request list and can be clicked to load more or all remaining rows.

Changing the Row Sorting of List Views

To reorder any view within MDOpen you can click on any of the column headings and the view will be sorted by that column. A down arrow ▼ in the column heading will indicate that the column is sorted by descending order and an up arrow ▲ in the column heading will indicate that the column is sorted in ascending order. In order to return the row sorting to the default, click the clear sort icon at the top right of the view.

Changing the Column Sorting of List Views

For the most important list views, the preferred ordering of the columns can be modified and retained for future use. To do so, click the icon at the top right of the view. This brings up a dialog listing all columns in the view in the current sort sequence. Drag and Drop a column within the list to move that column to a different position and then click Save. In order to return the column sorting to the default, click the Reset Column order icon at the top right of the view.

Setting List of favourite options for a List View

Each List view that provides the icon allows you to select your most commonly used options for that view. Each favourite option, when enabled for a row, will appear as an icon to the right of the first column for the row.

For Object Request options, set the favourites in the Object Requests view. They will then be made available for request rows within other views, such as for Tasks or RFPs.

Exporting Data in a List View to Excel

If the current list of rows, based on the filter criteria, should be export to an excel file, click the icon at the top right of the view. Then, select the location to save the file and click the Save button.

If the view has parent elements that can be expanded to list object requests, such as Project, Tasks or RFPs, the Export to XLSX option can be used to export the parent row and all children to an excel file.

Refreshing a List View

To refresh the list of rows for a view:

Click the Refresh icon at the top right of the view
Click the Refresh button in the filter section of the view, if applicable
Select option Refresh for a row
Press F5

Expand/Collapse Tree Elements

When the › arrow is displayed at the beginning of a row, it can be clicked to expand the view to display child elements for that row. It can clicked again to collapse the child element information.

To collapse all expanded rows in a list view, click the ⊟ icon at the top right of the view.

Filter Rows in a List

For many list views, an expandable filter settings section is available above the list. When expanded, a set of fields will be displayed allowing for the entry of filter values for that list.

Copy Editor Values to Clipboard

Within the metadata editor views, the 📋 icon is available at the top right of the editor. Click this to copy all of the property label/value pairs to the clipboard for pasting into another window.

Setting Default Values for fields in Add Dialogs

When selecting to add a new element (object request, task, RFP, attribute, etc…), a dialog window opens for the entry of data. For any editable field in the dialog, you can right-click on the field and select option “Set as default value”. The current value in the field with then be saved in the Eclipse workspace. When the same add dialog is then opened in the future, the default value will be loaded into the field for any otherwise blank field.

The list of set default values can be managed, exported and imported from the MDOpen preferences.

4.2 MDOpen Preferences

Certain perspective-wide preferences can be set for MDOpen and stored per instance of the Eclipse IDE.

To review/change the preferences for MDOpen, either navigate to Window->Preferences→MDOpen from the eclipse menu or click on Session→Preferences within a connected repository location.

Compare and Framesize Handling

Number of compare history to keep	Each time a two-way or three-way source comparison is performed, MDOpen saves the input values for simple reuse later. This number is the total number of distinct input values that should be saved (FIFO)
Initial framesize	The initial number of rows to load when a list view is opened or refreshed
Subsequent framesize	The size of any subsequent frames that are loaded when the user clicks the next frame icon at the top-right of a list view

Default Values
The list of any add dialog default values are listed here. They are added by right-clicking on an editable field value in an add dialog for an element in MDOpen.
Within the preferences:

- individual default values can be deleted
- default values for a specific dialog can be exported
- default values for all dialogs can be exported
- the default values in an exported file can be imported

Ignore List
Click the Ignore list button in this preference panel in order to add file patterns that should be ignored when importing objects into MDCMS from Local Objects, IFS, Remote Server, Git or SVN.

Logging

Log level	Debug – the most verbose amount of logging – may effect performance Info – also very verbose logging – may effect performance Warn – Warnings and Errors are logged Error – Errors are logged The logs themselves are stored in folder /Apps/mdopen_logs off the root storage device on the PC (in windows, this would typically be the C drive).

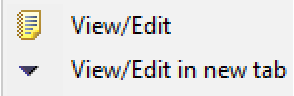
Perspective and View Management

Show outline view when opening LPEX editor	When true, a separate view opens at the bottom left of the MDOpen perspective that displays the source outline when an LPEX editor is opened for source editing.
Notify when new MDOpen version available	When true, MDOpen occasionally checks the defined update site to see if a new version of MDOpen is available. When this is the case, a message is displayed in the task tray.
Replace each editor on element left click for session	When true, any MDCMS element, except source, that is opened using the action left-click will replace the prior element that was shown in the MDCMS editor tab in the bottom portion of the perspective. When false, the left-click action will open the editor in a new tab.
Replace each view for same function on element right click for session	When true, when a child view is requested by right-clicking on a parent row, the view will replace any tab that had the same child view open, even if the child element is different. When false, the child view will only be replaced if for the same child element.
Optimize for Dark Theme	When true, the color scheme within the MDOpen perspective will be optimized for use with the Eclipse Dark Theme When false, the color scheme will be optimized for use with the Eclipse Classic Theme
Hide column specific icons in list views	When true, icons will not be shown within columns to indicate that the column can be clicked on to navigate to specific functions, such as level detail, defined commands, etc. When false, the icons will be shown. It is recommended to show the icons unless the display settings cause the icons to block the column text.
Use File Associations Preferences for SQL editing	This preference is only applicable when using MDOpen within Rational Developer for i. When true, then the editor used for sql database source types (such as tables, functions, views but not programs) will be based on the File Associations preference for the file type in Eclipse. When false, then the editor will explicitly be the Remote Systems LPEX Editor
Ignore Whitespace in Compare Function	When true, then when using the 2-way or 3-way compare function for source members or IFS files, differences in the amount of whitespace (blanks) won't be marked as different. When false, then differences in the amount of whitespace will be marked as different.

5 MDCMS Settings within MDOpen

The various MDCMS configuration settings are located within the expandable option named Settings.

5.1 Common Settings Options



View or change the detail of the setting element in a shared tab or new tab.

Add a new element for the specified setting without pre-filled information.

Add a new element for the specified setting with all fields pre-filled with the values from the selected existing element.

Delete the setting element

5.2 App Hierarchy

App Hierarchy displays the defined Application Attributes in tree form with Application as primary branch followed by level, type and then attribute. The App Hierarchy view opens by default in a view under the repository view.

Left-Click on a tree node to view details for that node.

Right-Click on a tree node to select a maintenance option. The options are the same as those discussed in the following sections for the specific node type.

5.3 Applications

An Application defines your business application software into manageable groups within MDCMS.

Complete descriptions of the Application Fields can be found in the MDCMS user manual.

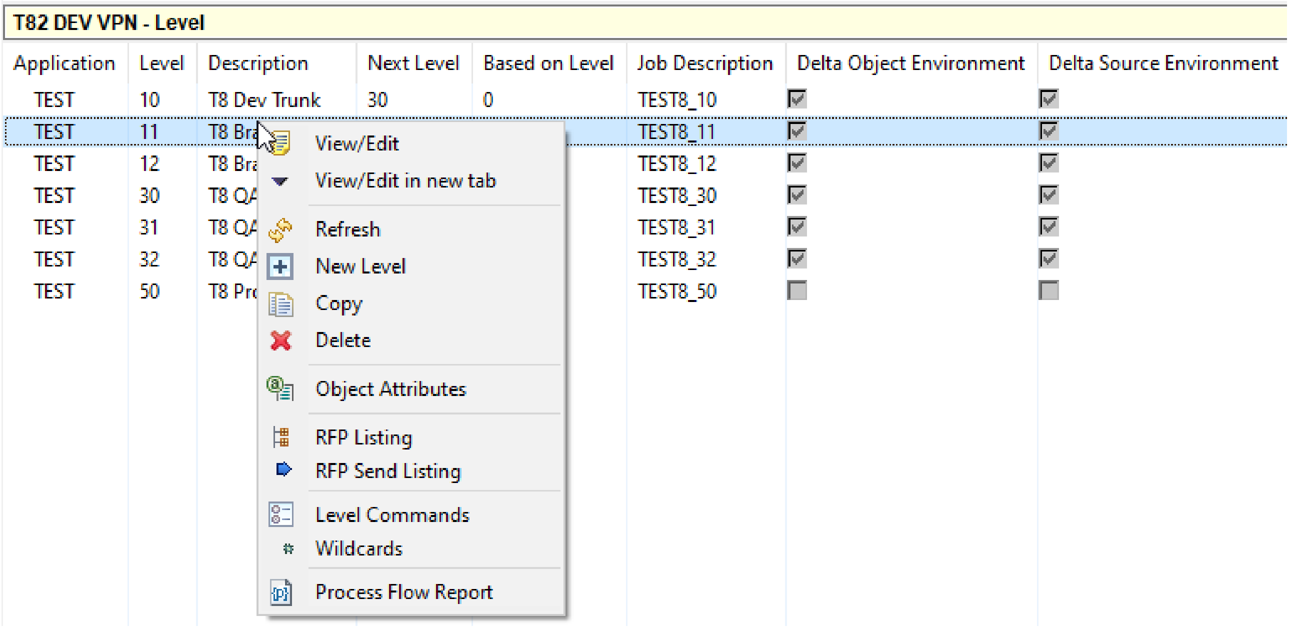
Options

Levels	Open Level View with rows filtered to selected Application
Deployment Flow Diagram	Generate the Deployment Flow Diagram for the selected Application
RFP Listing	Open RFP View with rows filtered to selected Application
RFP Send Listing	Open RFP Send View with rows filtered to selected Application

5.4 Levels

A Level defines a specific runtime environment (dev, test, prod, etc.) for an application.

Complete descriptions of the Level Fields can be found in the MDCMS user manual.



T82 DEV VPN - Level

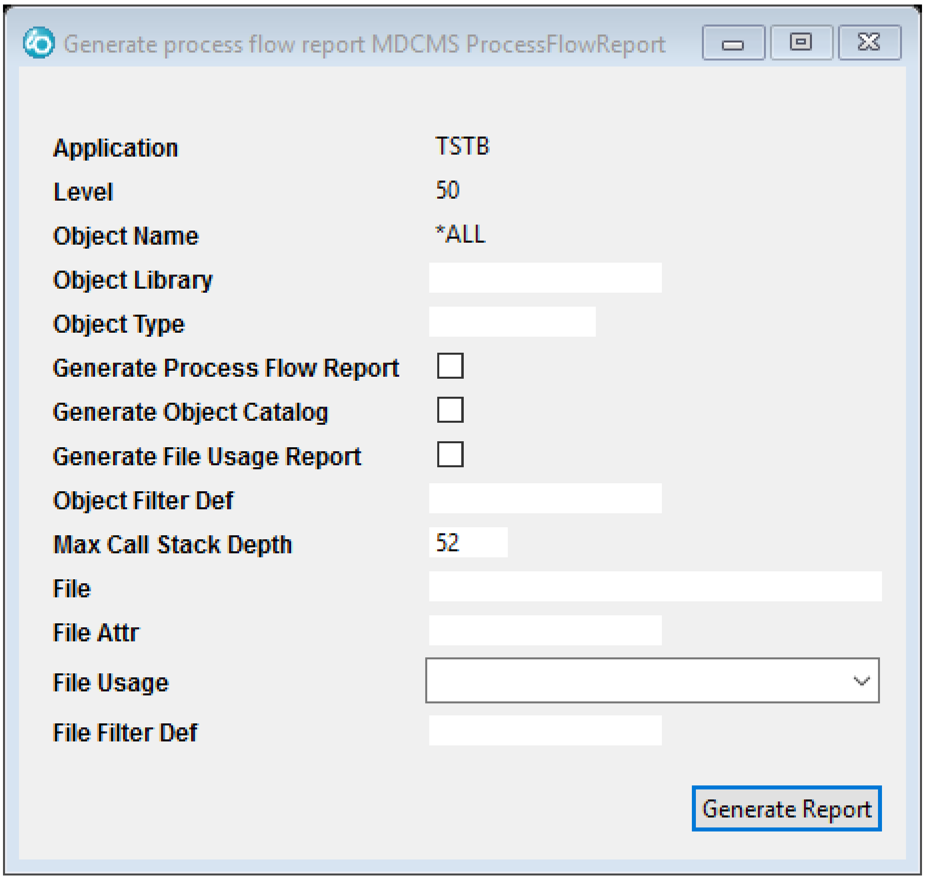
Application	Level	Description	Next Level	Based on Level	Job Description	Delta Object Environment	Delta Source Environment
TEST	10	T8 Dev Trunk	30	0	TEST8_10	☑	☑
TEST	11	T8 Bra			TEST8_11	☑	☑
TEST	12	T8 Bra			TEST8_12	☑	☑
TEST	30	T8 QA			TEST8_30	☑	☑
TEST	31	T8 QA			TEST8_31	☑	☑
TEST	32	T8 QA			TEST8_32	☑	☑
TEST	50	T8 Pro			TEST8_50	□	□

Options

Object Attributes	Open Attribute View with rows filtered to selected Application Level
RFP Listing	Open RFP View with rows filtered to selected Application Level
RFP Send Listing	Open RFP Send View with rows filtered to selected Application Level
Level Commands	Open Command View with rows filtered to *RFP commands (commands that run once per RFP regardless of contents) for selected Application Level
Process Flow Report	Open the ***ProcessFlowReport*** dialog for the selected Application Level

5.4.1 Process Flow Report

Function: to report on the flow of all execution objects for an application level.



Generate process flow report MDCMS ProcessFlowReport

Application	TSTB
Level	50
Object Name	*ALL
Object Library	
Object Type	
Generate Process Flow Report	□
Generate Object Catalog	□
Generate File Usage Report	□
Object Filter Def	
Max Call Stack Depth	52
File	
File Attr	
File Usage	
File Filter Def	

Complete descriptions of the Generate Process Flow Report for Level fields dialog can be found in the MDXREF user manual.

5.5 Deployment Flow Diagram

The Deployment Flow Diagram illustrates the migration flow for a specific application.

Parameters

Application	The application code for which the diagram should be drawn
Limit to Starting Level	The application which should serve as the starting point for the diagram. Leave blank to start at the lowest level.
Limit to Attribute	Specify a value for an MDCMS Attribute here if only levels where the attribute is defined should be included
Include Distribution Levels	None - exclude any levels on remote systems - only show level directly migrated to on this partition Default - include any levels that are distributed to by default, based on the Distribution Level settings All - include all levels that are distributed to
Retrieve Info from Remote Systems	True - detailed information about the level on a remote partition is included in the diagram False - only the level number for a remote level is included in the diagram

5.6 Attributes

An Attribute defines the target location and behaviour for a specific type of object in MDCMS.

Complete descriptions of the Attribute Fields can be found in the MDCMS user manual.

Options

Attribute Commands	Open Command View with rows filtered to selected Attribute
Attribute Scripts	Open Remote Server Script View with rows filtered to selected Attribute. This option is only applicable for *REMOTE attribute types.
Linked Attributes	Define the Attributes that are linked to the selected Attribute. When a checkout occurs for this attribute, the developer will be prompted to check out objects for the linked attributes.

5.7 Attribute Commands

A Command defines an executable IBM i command that runs during the promotion process at the time indicated by the command type.
Commands defined for an attribute are executed when an object is assigned to the attribute.
Commands defined with the attribute value = *RFP run once per RFP (installation package) regardless of the contents of the RFP.

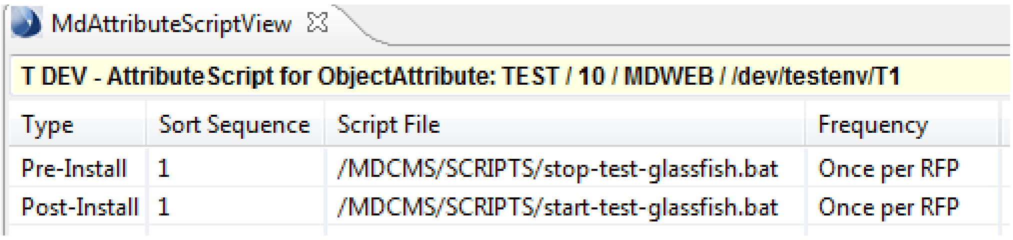
Complete descriptions of the Command Fields can be found in the MDCMS user manual.

MDOpen provides a Command Prompter for command keywords and descriptions. To use the prompter, enter at least the name of the command and then press the Command Prompter button. The entries into the prompter will be pasted into the command string.

MDOpen also provides for the automatic insertion of MDCMS wildcards into the command string. Position the cursor to the position in the string where a wildcard is necessary and then press Ctrl-Space for the list of Wildcard values.

5.8 Attribute Scripts

Scripts may be allocated to a specific *IFS or *REMOTE attribute to be executed during the installation of an RFP containing that attribute. To access the scripts for an Attribute, right click on the MDCMS Attribute and select ***Attribute Scripts***. The ***MDAttributeScriptView*** is displayed. The following is an example of the ***MDAttributeSpriptView*** that contains several of the possible script types:

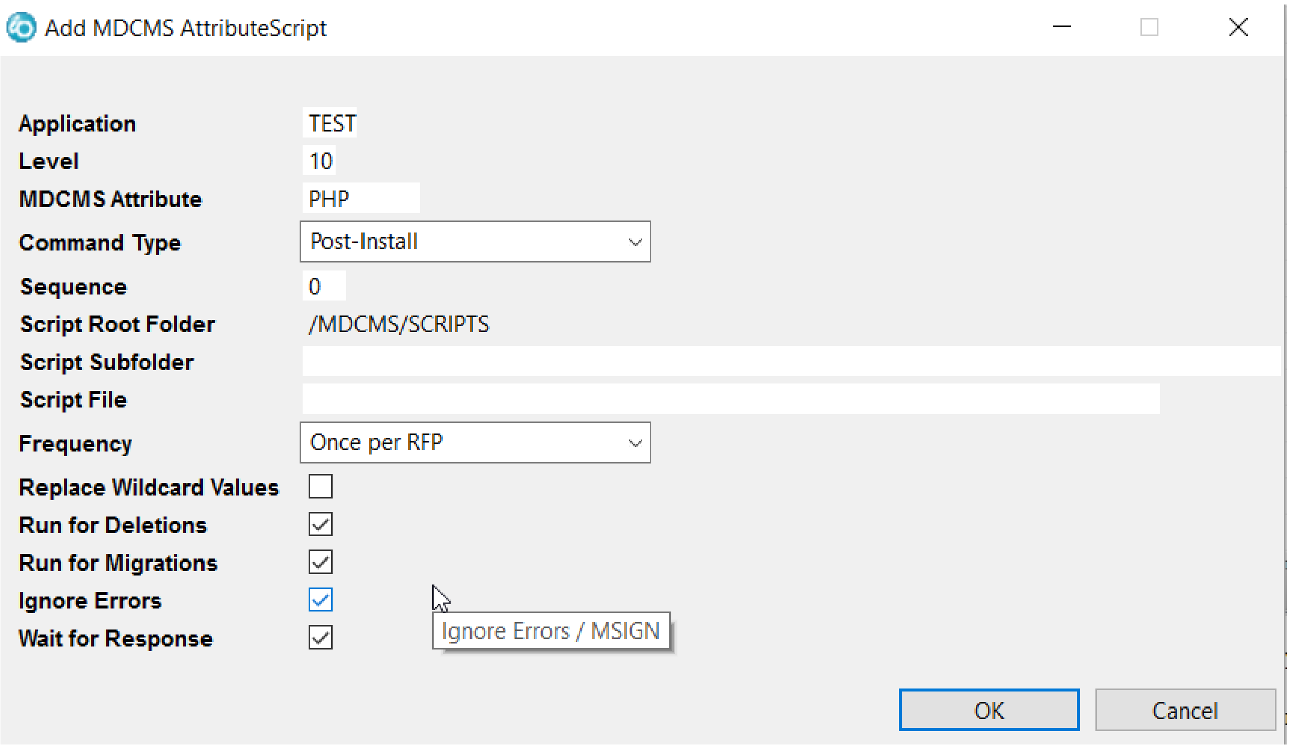


MdAttributeScriptView

T DEV - AttributeScript for ObjectAttribute: TEST / 10 / MDWEB / /dev/testenv/T1

Type	Sort Sequence	Script File	Frequency
Pre-Install	1	/MDCMS/SCRIPTS/stop-test-glassfish.bat	Once per RFP
Post-Install	1	/MDCMS/SCRIPTS/start-test-glassfish.bat	Once per RFP

To add a new script, right click within the ***MDAttributeScriptView*** and select ⊞ ***Add.*** That action brings up the ***Add MDCMS AttributeScript*** dialog.



Add MDCMS AttributeScript

Application	TEST
Level	10
MDCMS Attribute	PHP
Command Type	Post-Install
Sequence	0
Script Root Folder	/MDCMS/SCRIPTS
Script Subfolder	
Script File	
Frequency	Once per RFP
Replace Wildcard Values	□
Run for Deletions	☑
Run for Migrations	☑
Ignore Errors	☑
Wait for Response	☑

Complete the following parameters and click OK to add the Script to the Attribute.

Application
The Application Group that the Attribute resides in.

Level
The Application Group Level that the Attribute resides in.

MDCMS Attribute

This field is tied to the attribute defined in Source/Object Attribute Maintenance (see previous section).

Type

The Type designates when the script should run for this Attribute.

- Pre-Install – the script should be run prior to the installation of objects using the attribute
- Post-Install – the script should be run after the installation of objects using the attribute

Sort Sequence

The sort sequence of the script at run time, in case multiple scripts for the same attribute and type are defined.

Script Root Folder

The constant value /MDCMS/SCRIPTS which is the IFS location for script folders and files.

Script Subfolder

The relative path of the script within the root folder. Content assist is available to browse for the value.

Script File

The name of the script file within the folder path. Content assist is available to browse for the value.

Frequency

- Once per Object – the script should be run for each object in the RFP using this attribute
- Once per RFP – the script should be run once per RFP containing at least one object using this attribute

Replace Wildcard Values
A checked value (Y) indicates that the script contains wildcard values that should be replaced by the actual values at run-time.

Run for Deletions

A checked value (Y) indicates this script should run for deleted objects.

Run for Modifications

A checked value (Y) indicates this script should run for new or changed objects.

Ignore Errors

A checked value (Y) indicating if the RFP Installation should continue running if the script fails. **NOTE:** This flag is only considered for Pre-Install scripts.

***REMOTE Specific Parameters**

Wait for Response

When true, the MDCMS installation process will wait until a response is received from the script execution service on the remote server before continuing to the next step.

***IFS Specific Parameters**

Run as User Profile

The user profile under which the QSHELL script will run

Submit Job

When true, the IFS Script execution will be submitted to a separate job. MDCMS will not wait for a response in this case, but instead continue with RFP processing.

When false, the ifs script execution runs within the RFP job and the RFP won't continue to the next step until the execution is complete. When run inline with the RFP, any output from the QSHELL interpreter will be included in the RFP Deployment Log to assist with troubleshooting.

Job Name

The name of the submitted job that will process the IFS script

Job Queue

The name and library of the Job Queue to receive the submitted job

5.9 Object Commands

Object Commands are defined in the same way as for attribute commands, except that they are defined for a specific object and will be automatically applied to the object when it is checked out for the target level. If defining a C (compilation) command for a specific object, then it will be used instead of the compile command defined for the object's attribute.

Complete descriptions of the Command Fields can be found in the MDCMS user manual.

MDOpen provides a Command Prompter for command keywords and descriptions. To use the prompter, enter at least the name of the command and then press the Command Prompter button. The entries into the prompter will be pasted into the command string.

MDOpen also provides for the automatic insertion of MDCMS wildcards into the command string. Position the cursor to the position in the string where a wildcard is necessary and then press Ctrl-Space for the list of Wildcard values.

5.10 Object Scripts

Object Scripts are defined in the same way as for attribute scripts, except that they are defined for a specific object and will be automatically applied to the object when it is checked out for the target level.

See the above instructions for Attribute Scripts for details about the individual script parameters.

5.11 System i Settings

The global MDCMS settings for the partition.

Complete descriptions of the common System i Fields can be found in the MDCMS user manual.

Task Reference URL
This is an additional field in the System i Settings in MDOpen to provide the URL string template to use so that a developer can navigate to an external URL for a specific task or subtask.

When this string is defined, then a icon will be displayed in the Internal Reference column within the Tasks or Subtasks view for each row where an internal reference value is defined.
The column heading for the internal reference column is set in the System i Settings field Task Reference Label.

The following Wildcards can be used in the wildcard string and will be replaced with the runtime values when the view is displayed:

++PROJID++ - the project ID of the MDCMS project
++TSKREF++ - the internal task reference value for the task

An example Task Reference URL value:
https://gitlab.com/mycompany/++PROJID++/-/issues/++TSKREF++

5.12 OS/400 Locations

An OS/400 Location is another partition where MDCMS is installed and information or deployments should be shared between that partition and this partition.

Complete descriptions of the Location Fields can be found in the MDCMS user manual.

Left-clicking on a location allows for the definition of the basic information about the location as well as the DDM connection and the distribution method. Additional distribution configuration, based on the distribution method, is accessible by using option Distribution Settings.

Options

Export Data to Location	Provide project information from the local system to the target system. If the target system contains the MDWorkflow repository, then additional object and promotion information is provided. If the local system contains the MDWorkflow repository, it should also be synced initially to load the local information into the MDWorkflow tables.
Distribution Settings	Define the connection information to the target location for the distribution of RFPs based on the distribution method.
Distribution Levels	Define the list of target Application Levels for the distribution of RFPs

5.13 OS/400 Location Groups

OS/400 Location Groups provides the ability to group locations based on arbitrary requirements. A group can contain any number of OS/400 locations and an OS/400 location can belong to any number of groups.
Click on the OS/400 Location Groups option in the repository list to view and maintain the groups.

A group value can then be used to filter the list of locations for OS/400 Locations settings, Distribution Levels settings and the RFP Send listing.

5.14 Distribution Levels

A Distribution Level defines a target application level on an IBMI partition that objects for a given application and level should be deployed to.

Complete descriptions of the Distribution Level Fields can be found in the MDCMS user manual.

Options

Target Attributes	Open List of all Attributes for selected Target Level to view/change what should be sent for each attribute. Left-Click the MDCMS Attribute Option to select a different value from the drop-down-list.

5.15 Remote Server Locations

Remote Server Locations are the connection definitions used by the MDFTP client for one of the following:

- non-IBM i servers that you will be deploying *REMOTE objects to during an RFP deployment. In order for MDCMS to deploy to a server, that server must have an FTP service running on it. The Remote Server Location definition describes how MDCMS as the FTP client is to connect to the FTP service on the remote server.
  A Remote Server Location is then defined for each *REMOTE attribute. In this way, MDCMS knows the destination of each remote object for each level within the migration path.

- an IBM i partition or a secure FTP server on another platform that MDCMS settings and RFPs should be sent to when they should be sent via SFTP instead of connecting to the native IBM FTP server.
  The OS/400 location distribution method would then be set to MDF (MDFTP client) and the Distribution Address would be the description of the Remote Server Location definition.

Fields

Description	A description of the location to make it easy to identify from a list. The description is then applied to each Attribute that will connect to the server using the RemoteServerLocation definition, or applied to the Distribution Address on a OS/400 location using the MDF method.
Server Address	The address of the server that is known to the IBMi partition
FTP Method	• FTP – standard File Transfer Protocol • FTPS – FTP over SSL • SFTP – FTP over SSH
Port	The port number for the FTP service
Transfer Mode	• Active • Passive
Authentication	• Anonymous – no credentials are required to connect to the FTP server • User/Password - MDFTP will pass a User and Password when connecting to the FTP server • User/SSH Key - MDFTP will pass a User and SSH key when connecting to the SFTP server
User	A user id with registered credentials on the FTP server Value required when Authentication not anonymous
Set New Password	The password for the user Value Required when Authentication = User/Password and Password Registered = false
Proxy Address	The address of a proxy server to route the FTP connection through, if necessary
Proxy Port	The port number of the proxy server to route the FTP connection through
Proxy Type	• HTTP • SOCKS5

Proxy User	The user id for the connection to the Proxy Server, if necessary
Set New Proxy Password	The password for the Proxy User, if necessary
FTP Timeout in Seconds	The maximum amount of time to wait for a response from the FTP service. If the timeout is reached, MDCMS assumes that the FTP deployment step has failed
Update Permissions	• False – the folder/file permissions will not be updated after deployment to the server • True – the folder/file permissions are set based on the attribute object authority settings
Client Certificate Keystore	The path in IFS to the Keystore for FTPS connections
Set New Keystore Password	The password of the Client Certificate Keystore
Client Certificate Keystore Type	• JKS – java keystore • PKCS12 – Public Key Standards 12
FTPS Encryption Method	• Implicit – connection without negotiation • Auth SSL – request security and step up to SSL encryption • Auth TLS – request security and step up to TLS encryption
SSH Private Key in IFS including path	when using SFTP and User/SSH Key authentication, MDFTP authenticates by passing a private key file to the SFTP server. This key must be stored in the IFS. The path to the key, including the key name, should be entered here. For example: /home/myuser/.ssh/okey4096
Remove SSH Private Key Passphrase	If the private key was registered in MDOpen to be protected by a passphrase, but no longer is, then select this checkbox
Set New SSH Private Key Passphrase	Enter the value of the passphrase that protects the contents of the SSH private key file
Script Runtime Folder on Server	The path on the remote server where Pre-Install or Post-Install scripts should be placed. If scripts are used on this server, a service must be installed on the server to monitor for script files in the defined folder. This is only applicable for usage by *REMOTE attributes.
Script Timeout in Seconds	The maximum amount of time to wait for a response file to be placed in the scriptFolder/OK or NOK subfolder. If the timeout is reached, MDCMS assumes that the script run has failed.
Script Command Folder Symbol	/ is the standard folder symbol for non-windows systems \ is the folder symbol for windows systems

5.16 Services

The Services list provides an overview of all possible persistent batch jobs that run in the background to carry out MDCMS processes. The job name for a service is the same as the service name + the environment ID.

A service can be started or ended from the Service editor. To do so, left-click on the service in the list and then click the Start Jobs or End Jobs button in the editor.

Complete descriptions of the Service Fields can be found in the MDCMS user manual.

Options

Logs	Some of the services write output to log files in the IFS. Select this option to list and view the log files for a specific service

5.17 Git Repositories

If Git is used in your organization for source management, MDCMS can connect with Git Repositories via the MDGIT service for the following purposes:

- Cross-reference usage of IBMi system or database objects by open-source code stored in Git.
- Automatically checkout and deliver project components directly from Git to MDCMS for managed deployment to the IFS or remote servers.
- Manually request objects from the Git tree in the eGit eclipse perspective.

When MDCMS connects to a Git repository on a server for the first time, it will clone the defined branch to the IFS within folder /MDCMS/EXTREF/<instance>/GIT/<Repository ID>. Whenever a refresh is requested or triggered, MDCMS will pull any changes from the server repository to the clone.
All cross-referencing or object request actions then occur against the clone.

Note: If you wish to only manually request project components from Git, without making use of cross-referencing or Continuous Integration (CI), you can do this directly from the eGit perspective and Git Repositories don't need to be defined in MDOpen. See chapter Request Objects from Git for more information on this feature.

Fields

Repository ID	A unique, user-defined 10-character ID for the Git Repository branch
URL	The http(s) or git address of the Git repository. An example https format would be https://bitbucket.org/mycompany/myrepo An example git format would be git@github.com:mycompany/myrepo.git When using SSH authentication, the git address format should be used
Description	A description of the repository branch
Authentication	• Anonymous - no credentials are required to connect to the git server • User/Password - a user and password are used to authenticate • SSH - an SSH private key is used to authenticate
Repo User	A user id authorized to read and clone the Repository on the Server when authentication = User/Password
Set New Password	The password for the user when authentication = User/Password
Register new Private/Public Key Pair	enter the value of the path and key file on your pc for either the private or public key file. Or, click the folder icon to browse and select a key file. The public key and private key must be in the same folder and have the same name (except the public key has a .pub suffix). This public key must be the same as a key registered on the git server. Once registered in MDOpen, you can click the View public key button in the dialog to view the string if the key isn't yet registered on the git server.
Set New Passphrase	enter the value of the passphrase that protects the private key

Branch	The repository branch to use for this Repository ID. If left blank, the master branch will be used. If multiple branches should be considered for a Repository, then create additional Repository definitions for each additional branch.

Options

New Branch for Repository	Clone a different branch using the same credentials for the repository. MDOpen will then use the MDGIT service to connect to the git server and list any branches that aren't already defined in the GITRepositoryView
External References	View/manage the references between code in the Git branch and objects on the IBMi
External Reference Levels	View/manage the mapping between MDCMS levels and the paths within the repository that should be cross-referenced.
Continuous Integrations	View/manage the mapping between MDCMS IFS/Remote attributes and the paths with the repository that should be automically requested from whenever changes are pushed to the Git server.
User Mapping	If the user ids on the git server are different than the user profile of the developers in MDCMS, the ids can mapped using this function. Then, when a push is performed by a user to the git server, that user's mapped profile will be used for the object request and RFP ownership.
Test Connection	Verify that a connection can be established with the Git server based on the entered field values.
Refresh External References	Refresh the cross-reference information for all defined xref paths for this Git Repository branch

To have the cross-referencing for all Git repositories regularly refreshed, add *GIT to the list of libraries in MDXREF for an Application Level.

5.18 SVN Repositories

If SVN (Subversion) is used in your organization for source management, MDCMS can connect with SVN Repositories via the MDSVN service for the following purposes:

- Cross-reference usage of IBMi system or database objects by open-source code stored in SVN.
- Automatically checkout and deliver project components directly from SVN to MDCMS for managed deployment to the IFS or remote servers.
- Manually request objects from the SVN tree in the Subclipse or Subversive eclipse perspectives.

Any code or components that MDCMS requires in order to perform cross-referencing or object requests will be copied to the IFS within folder /MDCMS/EXTREF/<instance>/SVN/<Repository ID>.
All other paths within the SVN repository will be ignored to minimize disk space and processing time.

Fields

Repository ID	A unique, user-defined 10-character ID for the SVN Repository branch
URL	The http(s) address of the SVN repository
Description	A description of the repository
Authentication	• Anonymous - no credentials are required to connect to the SVN server • User/Password - a user and password are used to authenticate
Repo User	A user id authorized to read and fetch from the Repository on the Server when authentication = User/Password
Password	The password for the user when authentication = User/Password
Include Branches in MDXREF	Specifies if the /branches path should be included when cross-referencing code
Include Tags in MDXREF	Specifies if the /tags path should be included when cross-referencing code

Options

External References	View/manage the references between code in the SVN repository and objects on the IBMi
External Reference Levels	View/manage the mapping between MDCMS levels and the paths within the repository that should be cross-referenced.
Continuous Integrations	View/manage the mapping between MDCMS IFS/Remote attributes and the paths with the repository that should be automically requested from whenever changes are committed to the SVN server.
Test Connection	Verify that a connection can be established with the SVN server based on the entered field values.
Refresh External References	Refresh the cross-reference information for all defined xref paths for this SVN Repository

To have the cross-referencing for all SVN repositories regularly refreshed, add *SVN to the list of libraries in MDXREF for an Application Level.

5.19 External Reference Levels

If non-native source code is stored in Git, IFS or SVN, and that code possibly references native IBMi objects, MDCMS can cross-reference that code so that developers are aware of impacts to that code when a change is made to one of the objects.

The Level definitions map specific repository/IFS paths where the non-native source resides with specific Application Levels where the IBMi objects reside.

Fields

Type	*GIT – the code is stored in a Git repository. The repository must be defined in Settings->Git Repositories *IFS – the code is stored within an IFS (Integrated File System) path *SVN – the code is stored in a SVN repository. The repository must be defined in Settings->SVN Repositories
Repo ID	The unique ID of the Git or SVN repository. Leave blank for IFS
Path	The relative path within the repository or IFS. Content-assist can be used to navigate downwards through the tree. If the path is left blank for a repository, then the entire repository will be cross-referenced. The path must be defined for IFS
Application	The MDCMS application to reference against.
Level	The MDCMS application level to reference against. Only objects belonging to libraries defined in MDXREF as belonging to the level will be considered. If multiple application levels are referenced by a given repository or IFS path, then create row for each combination.

Options

Refresh External References	Refresh the cross-reference information for the defined Path/Level combination

NOTE: See the separate MDCMS Rest API documentation for instructions on setting up the Repository webhooks to automatically update MDXREF and request objects for deployment when a commit is pushed to the Git or SVN repository.

5.20 External Reference File Extensions

If non-native source code is stored in Git, IFS or SVN, and that code possibly references native IBMi objects, MDCMS can cross-reference that code so that developers are aware of impacts to that code when a change is made to one of the objects.

File extensions define which type of non-native source code is managed by your organization.
Only non-native code with a matching file extension will be checked for object references.

When the checking occurs, MDXREF inspects the non-commented, executable portion of the code for strings that match the object names of types to be considered for referencing.
A string match isn't case-sensitive.
A string must be delimited on both sides with a non-alphanumeric character (such as space, ", :, etc.) in order to be considered a match.

Using this logic, a reference will be found in any file, so that MDCMS doesn't need to understand the nearly infinite number of ways that an object can be referenced based on syntax. However, false-positives can occur. The false-positives can be permanently ignored within the External References view or when an RFP missing dependency warning is displayed.

MDCMS is shipped with the extensions for C#, java, PHP and Python. Other file extensions can be added from this view.

Fields

File Extension	The type of file, which is the part of the file name after the final period
Line Comment Delimiter 1	A comment delimiter for the language that indicates that the remaining text is a comment until the end of the line.
Line Comment Delimiter 2	A 2nd comment delimiter for the language that indicates that the remaining text is a comment until the end of the line.
Comment Block Begin	The comment delimiter for the language that indicates that a comment has commenced and continues until the Comment Block End delimiter is encountered
Comment Block End	The comment end delimiter for a comment block
Exec Block Begin	A delimiter to indicate that executable code has begun. If left blank, then MDXREF will consider that code can be executed from the very beginning of the file
Exec Block End	A delimiter to indicate that executable code has ended. If left blank, then MDXREF will consider that code can be executed until the very end of the file. If an end delimiter is encountered, MDXREF will ignore everything until the next Begin delimiter is found.

Options

System Types to Reference	Define the list of Object or SQL types/attributes that can be referenced by the given file extension. Limit the list to only those types/attributes that might be referenced to keep the referencing time and false positives to a minimum.

5.21 Continuous Integrations

Continuous Integration (CI) is the concept of seamless flow from component bundling to deployment. MDCMS utilizes this concept for components that are committed to Git or SVN repositories.

Whenever changes are pushed to a Git Server or committed to an SVN server, MDCMS can generate object requests for those components for eventual deployment to target IFS folders or remote servers.

Fields

Type	*GIT – the components are stored in a Git repository. The repository must be defined in Settings->Git Repositories *SVN – the components are stored in an SVN repository. The repository must be defined in Settings->SVN Repositories
Repo ID	The unique ID of the Git or SVN repository
Path	The relative path within the repository. Content-assist can be used to navigate downwards through the tree. If the path is left blank for a repository, then the entire repository will be included.
Application	The MDCMS application to contain the object requests
Level	The MDCMS application level to contain the object requests. The level must allow Checkout to be valid.
Checkout	What to request when changes are committed to the server. Diffs – only differences (adds, changes, deletes in comparison with the contents prior to the commit) Contents – all folders and files within the path will be requested for deployment
Request IFS/Remote Objects	when selected, the files to be requested will be deployed to the IFS or a remote server as objects, such as java, php, etc.
Request Source for IBM i Objects	when selected, the files to be requested will be deployed as source for IBM i objects to be compiled from.
MDCMS Attribute	The *IFS or *REMOTE attribute to use for the requested objects. Required when Request IFS/Remote Objects is used. If a value is entered when requests are for source for IBM i objects, then the attribute will be fixed for those requests. If left blank, then MDCMS will attempt to dynamically assign the attribute based on history, name and suffix.
Default User	The IBM i user profile to be the owner of the object requests when the repository user ID isn't mapped to a profile in MDSEC. This profile must already be registered in MDSEC with authority to request objects in MDOpen.
Relative Path	From Repo Root – the Relative Path of the objects when copied to the target folder will be from the root of the Git or SVN repository. This means that the folder structure on the target will exactly match the folder structure of the repository.

	From Repo Path – the Relative Path of the objects when copied to the target folder will begin from the end of the value in the Path parameter.
Project	The MDCMS Project to assign the object requests to. If blank, the requests can be assigned to a Project later. The following special values can be used to dynamically assign a project, task and subtask by parsing the beginning of the commit message: *JIRA - the beginning of the commit message will contain the value in this format: PROJECTID-ISSUENUMBER: for example: TESTPROJ-307: MDCMS will then map the Jira issue to the MDCMS task or subtask. *MD - the beginning of the commit message will contain the value in this format: PROJECT-TASKNUMBER.SUBTASKNUMBER: for example: TESTPROJ: if just for a project. TESTPROJ-307: if for a project and task. TESTPROJ-307.5: if for a project, task and subtask
Task	The MDCMS Project task to assign the object requests to, if applicable and already known. Leave blank when using special values for the project.
Subtask	The MDCMS Project Subtask to assign the object requests to, if applicable and already known. Leave blank when using special values for the project.
Generate RFP	True – when a commit occurs, a new RFP will be created and the object requests will be assigned to it. False – the object requests will be created in unassigned status
RFP Description	The description to apply to the newly created RFP *COMMITMSG – the RFP description will be the comments provided with the commit to the Git or SVN server.

NOTE: See the separate MDCMS Rest API documentation for instructions on setting up the Repository webhooks to automatically update MDXREF and request objects for deployment when a commit is pushed to the Git or SVN repository.

5.22 eMail Bodies

The eMail Bodies function enables a user to define the contents used for the body of an email for the MDCMS Workflow process or for use within the MDMAILF command.

A set of email templates are predefined and are placed in IFS folder /MDCMS/MAIL. When upgrading to a new version of MDCMS, the existing templates are left in place and the new templates are placed in a subfolder named the same as the version id (for example 7.3).

View/Edit an existing email body by left-clicking on the file name. MDOpen provides its own editor for email bodies that allow for the Content-Assist selection of wildcards to be inserted at the current cursor location within the body.

Navigate into a subfolder by left-clicking on the folder name. Navigate up to a parent folder by left-clicking on folder 📂.. .

Options

New Folder	Create a subfolder
Rename	Rename an existing file or folder

When option Copy is used, the file can be copied to the same folder or to a different folder within the /MDCMS/MAIL path. Content-Assist is available for the New Folder parameter.

5.23 Script Templates

The Script Templates function enables a user to define the scripts to be run on Remote Servers before or after *REMOTE objects are installed, or to define the scripts to be run on the partition before or after *IFS objects are installed.

If for *REMOTE, the script should be written in the language that the remote server understands as the script will be placed on the server at runtime after any wildcard values have been replaced by runtime values.

If for *IFS, the script should be written to be understood in a QSH session.

View/Edit an existing script by left-clicking on the file name. MDOpen provides its own editor for scripts that allow for the Content-Assist selection of wildcards to be inserted at the current cursor location within the body.

Navigate into a subfolder by left-clicking on the folder name. Navigate up to a parent folder by left-clicking on folder 📂.. .

Options

New Folder	Create a subfolder
Rename	Rename an existing file or folder

When option Copy is used, the file can be copied to the same folder or to a different folder within the /MDCMS/SCRIPTS path. Content-Assist is available for the New Folder parameter.

5.24 Project Phases

The Project Phase is used as a category for time entered for work performed for a Project or Task. Project Phases can be created and managed from this function.

5.25 Project Types

The Project Type is used as a category for projects as well as to set some general rules for Projects of the given type.

Complete descriptions of the Project Type Fields can be found in the MDCMS user manual.

Options

Task Types	The list of task types that are permitted to be used for tasks within projects of a given Project Type. One default task type can also be selected per Project Type to reduce keying by the user.
Custom Fields	Select/Unselect the defined custom fields that should be available for projects of a given Project Type
Status Transitions	Define the transitions between the defined status codes to use for projects of a given Project Type. This gives you the ability to define a separate process workflow for each type of Project. To change which Status Codes can transition to a specific code, left-click on the Transition From Status column for the code's row and then select/de-select from the list of defined codes. To change which Status Codes can get transitioned to from a specific code, left-click on the Transition To Status column for the code's row and then select/de-select from the list of defined codes. To get a graphical overview of the transitions for a Project Type, click the icon at the top-right of the Status Transitions view

5.26 Task Types

The Task Type is used as a category for tasks and subtasks as well as to set some general rules for Tasks of the given type.

Complete descriptions of the Task Type Fields can be found in the MDCMS user manual.

Options

Custom Fields	Select/Unselect the defined custom fields that should be available for projects of a given Task Type
Status Transitions	Define the transitions between the defined status codes to use for projects of a given Task Type. This gives you the ability to define a separate process workflow for each type of Task. To change which Status Codes can transition to a specific code, left-click on the Transition From Status column for the code's row and then select/de-select from the list of defined codes. To change which Status Codes can get transitioned to from a specific code, left-click on the Transition To Status column for the code's row and then select/de-select from the list of defined codes. To get a graphical overview of the transitions for a Task Type, click the icon at the top-right of the Status Transitions view

5.27 User Group Types

The User Group Type is used as a category for user groups.

5.28 User Groups

A User Group is a collection of users defined in MDSEC that can then be used for Project assignments, Workflow management and Reporting.

Left-Click on the Users column to view/manage the users belonging to a User Group.

6 MDWorkflow Settings within MDOpen

The MDWorkflow section contains settings functions that can be managed from MDOpen, if a valid MDWorkflow license is applied to the partition that MDOpen is connected to.

6.1 Acceptance Group Types per Level

For each MDCMS Promotion Level, between 0 and n types of groups can be designated to be responsible for the acceptance of an installed RFP into that level. This means that each time an RFP, which is a package of objects, is installed into a specific level, 1 person from a group for each required group type must accept the results of the installation before that object package will be allowed to move on to the next step in the Workflow process.

Example:
MDCMS Promotion Level 50 is defined as the User Acceptance Environment level. When objects are installed into this level, business users and IT management must test the environment before those objects are allowed to be installed into the Production Environment level. To ensure that this occurs, group types BUS_USER and IT_MGMT are specified as Required Acceptance Group Types for level 50. Then, whenever an RFP is installed into level 50, a user within a group of type BUS_USER and a user within a group of type IT_MGMT must accept the RFP before it can be sent to Production.

Fields

Application	MDCMS Application
Level	The MDCMS Application Promotion Level
Group Type	The type of group that must accept the results of an installation into the given Level
Description	Description of the Group Type

6.2 Project/Task Status Codes

MDCMS comes packaged with several fixed Status Codes to be used by Projects and Tasks. Additional Custom Status Codes can be created to fulfil the personal requirements for the Organization.

Fields

Status	A 1 Character Code for the Status of a Project or Task
Description	A description of the Status
Sort Sequence	The order in which the Status Code appears in the list. It is recommended to sort the status codes based on when a Project or Task would reach that status. Status Codes that indicate the end of a Project or Task may not be sorted lower than Status Codes indicating an active Project or Task
Ending Status	The Code indicates that the Project or Task has ended
Use in Projects	If true, the Status Code may be used by Projects
Use in Tasks	If true, the Status Code may be used by Tasks
Allow Auto-Update	If true, a project or task can be set to this status automatically via the use of an MDCMS API.
Allow Manual-Update	If true, a project or task can be set to this status from within the MDCMS, MDOpen or MDWorkflow client screen.
Group Type Authorized to Update	Limit the persons who can manually update a project or task to this status to involved groups of the entered Group Type.

6.3 Project/Task Status Boundaries

An RFP in MDCMS can be blocked from performing a certain step for specific application levels until Projects or Tasks that are impacted by the RFP are within a defined range of status codes.

If the RFP's objects are assigned to a subtask, then the status of the subtask is considered, else, if the RFP's objects are assigned to a task, then the status of the task is considered, otherwise the status of the project is considered.

If at least one of the impacted projects or tasks have a status outside the allowed range, then the requested action will be denied.

Fields

Application	The MDCMS application
Level	The MDCMS application level
RFP Action	The action that should have the boundary check performed Submit – Begin the validation/bundle phase of the RFP Approve – Approve the RFP for Installation Install – Install the objects on the RFP to the target application level Send – Send the RFP to other partitions/branches
Min Proj/Task Status	The minimum status, based on the sort sequence of the status codes, that the impacted project, task or subtask can have to allow the action to occur. If the minimum isn't set, then any code that is <= the maximum is ok.
Max Proj/Task Status	The maximum status, based on the sort sequence of the status codes, that the impacted project, task or subtask can have to allow the action to occur. If the maximum isn't set, then any code that is >= the maximum is ok.

6.4 Project/Task Status Triggers

Triggers can be put in place to automatically initiate the submit, approve, install or send of RFPs for a Project or Task when the status of the Project or Task changes.

Triggers are checked whenever a status changes for a Project, Task or Subtask.

Fields

Application	The target MDCMS application of the RFP
Level	The target MDCMS application level of the RFP
RFP Action	The action that should automatically execute: Submit – Begin the validation/bundle phase of the RFP Approve – Approve the RFP for Installation Install – Install the objects on the RFP to the target application level Send – Send the RFP to other partitions/branches
New Status	The new status value that has just been transitioned to
From Status	The prior value of the status. Leave blank to trigger the action when transitioning from any status to the new status. Otherwise, enter a value to limit the transitions causing the trigger. A Trigger record can be created for each transition if multiple from values, but not all from values, are necessary.
Project Trigger	True = trigger the action when the status was changed on a project
Task Trigger	True = trigger the action when the status was changed on a task
Subtask Trigger	True = trigger the action when the status was changed on a subtask

6.5 Custom Fields

Additional Fields can be added to the Project or Task Detail screens. These fields are then available for list filtering, managing and reporting within MDOpen and MDWorkflow.

Fields

Field Name	A 10 Character Code for the ID of the custom field
Description	A description of the field – this is then displayed as the label for the field
Sort Sequence	The order in which the field appears in the list
Type	String – alphanumeric Number – numeric Date Time Checkbox DropDownList Text Box – a large string, with rich text editing when used in the MDWorkflow UI URL – web link User – a registered User ID in MDSEC
Length	The total length of a field, including decimal positions for a numeric field The maximum length for a string is 160 and the maximum length for a number is 24
Decimal Positions	The number of decimals for a number field. The maximum is 9.
DropDownList View	What should be displayed to the user for a DDL field DDL Code DDL Value Both
Visible from Stastus	The minimum status that allows the field to be seen. If blank, then the field is always visible.
Required from Status	The minimum status that requires that a value is entered for the field. If blank, then the field is always optional.
Editable from Status	The minimum status that allows a value to be entered or changed. If blank, then the field is editable as soon as it's visible.
Editable until Status	The maximum status that allows a value to be entered or changed. If blank, then the field is editable until the project or task is closed.
Allow Auto-Update	If true, the field can be updated via the MDUPDCFLD API.
Allow Manual-Update	If true, the filed can be updated from within the MDOpen or MDWorkflow client screen.
Group Type Authorized to Update	Limit the persons who can manually update the field to involved groups of the entered Group Type.
Wildcard ID	A 6-character ID to use in commands or email templates. When the command or email is prepared, the value ##<wildcard id>## will be replaced by the value of the Custom Field for the applicable Project or Task.

Options

Project Types Using Field	The list of all project types is opened so that the user can see which Project Types allow for the use of the field as well as to change the selections.
Task Types Using Field	The list of all task types is opened so that the user can see which Task Types allow for the use of the field as well as to change the selections.
DDL Entries	Manage the list of DropDownList elements for the DDL field

6.5.1 DDL Entries

This view allows for the maintenance of the list of Code Elements for a Custom DropDownList field. Use option DDL Entries for a DDL field to open this view.

Fields

Code	The unique code value to identify the element. Maximum Length of code is set by the parent field
Description	The description of the code, which can be set to be displayed to the user
Sort Sequence	The order in which the code appears in the list
Selectable from Status	The minimum status that allows this code to be selected. If blank, then the code is selectable as soon as it's visible and editable.
Selectable until Status	The maximum status that allows this code to be selected. If blank, then the code is selectable until the field is no longer editable.

6.6 WebApp Config xml

The MDWorkflow web app configuration settings, which are stored in file conf-faces-config.xml within the MDWorkflow web app within the WEB-INF folder, can be saved in MDCMS and then published within MDOpen.

By storing the values in MDCMS, it is very simple to set the properties and then republish them when a new version of MDWorkflow is installed.

The initial field, Config File, should be the full path to the conf-faces-config.xml that should be updated when the Publish to Config File button is clicked. Content-Assist can be used to navigate the IFS directories to get to the file.

Complete explanations of the property fields can be found in the MDWorkflow installation instructions.

Buttons

Save	Save the settings to MDCMS for reuse at a later time
Publish to Config File	Update the contents of the Config File with the current values

7 Repository Session Functions

The various MDCMS session settings are located within the expandable option named Session.

7.1 Change password

Use this option to change the password of the user profile defined for the Repository connection. This will change the password in the connection definition and for the user profile on the IBM i partition.

7.2 Show locked objects

If source located on the IBM i is opened in an editor outside the control of Eclipse (such as MS Word or Notepad), MDOpen isn’t informed when the editor is closed again to automatically release the lock on the source member.

Use this option to view all source members that were locked by MDOpen for editing in an external editor.

7.3 Preferences

Set MDOpen preferences. See section MDOpen Preferences for more information.

7.4 Close editors

Use this option to close any open non-source editors in the MDOpen perspective that belong to the repository connection. The ☒ icon at the top of the repository view can also be used to close the editors for the default repository connection.

7.5 Report output

Use this option for reports (MD Output) generated to be copied to a formatted table, deleted, viewed, printed, exported or emailed. The exported and emailed formats supported are CSV, PDF, TXT and XLS.

Left clicking on a report displays the report as a TXT file. NOTE: If the file does not open, you probably lost your connection to the server.

Options

View/Edit	Display the report as a TXT file.
Copy	Copy to PF – Copy the detail contents of the report into a formatted table (DDS Physical File). This provides a simple means to extract information out of the MD database for use in SQL, Queries or programs.
Delete	Permanently delete the report.
Print	Print the report contents to a spooled file.
Export	Export the report to a CSV, PDF, TXT or XLSX formatted file. The file can be placed in the IFS or emailed to one or more recipients. See the MDXREF or MDCMS User Manual for the parameters of command MDRUNRPT for more information.

7.6 Spooled output

Use this option for spool files generated to be deleted, viewed, exported or emailed. The exported and emailed formats supported are PDF and TXT.

Left clicking on a spool file displays the spool file as a TXT file. NOTE: If the file does not open, you probably lost your connection to the server.

Options

View/Edit	Display the spool file as a TXT file.
Delete	Permanently delete the spool file.
Export	Export the spool file to a PDF or TXT formatted file. The file can be placed in the IFS or emailed to one or more recipients. See the MDXREF or MDCMS User Manual for the parameters of command MDEXPSPLF for more information.

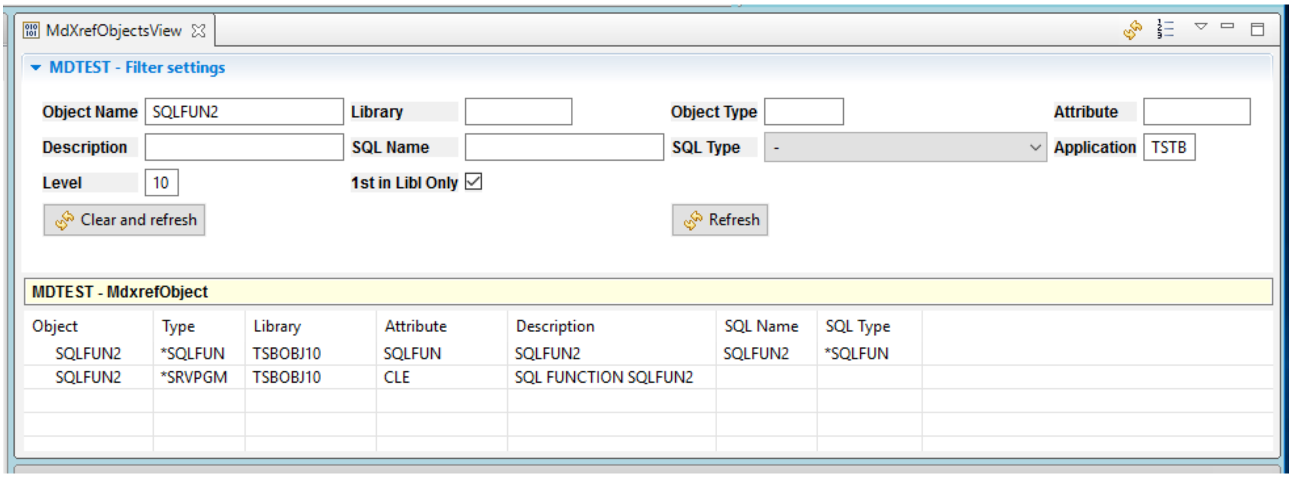
8 MDXREF

The MDXREF section of MDOpen provides the cross-reference functionality for referential analysis of objects, source, procedures and fields, as well as listing and searching abilities for IFS, Git and SVN.

Prerequisite:
The Cross-Referencing information for the Application Levels must already have been built on the partition. This is done within the green-screen MDXREF product. See the MDXREF manual for more information.

8.1 MDXREF Objects

The MDXREF Objects View ***MdXrefObjectsView*** provides the user with a filtered list of IBMi objects in all libraries that are built in the MDXREF cross reference product.



Object Name	SQLFUN2	Library		Object Type		Attribute	
Description		SQL Name		SQL Type	-	Application	TSTB
Level	10	1st in Libl Only	☑				

Object	Type	Library	Attribute	Description	SQL Name	SQL Type
SQLFUN2	*SQLFUN	TSBOBJ10	SQLFUN	SQLFUN2	SQLFUN2	*SQLFUN
SQLFUN2	*SRVPGM	TSBOBJ10	CLE	SQL FUNCTION SQLFUN2		

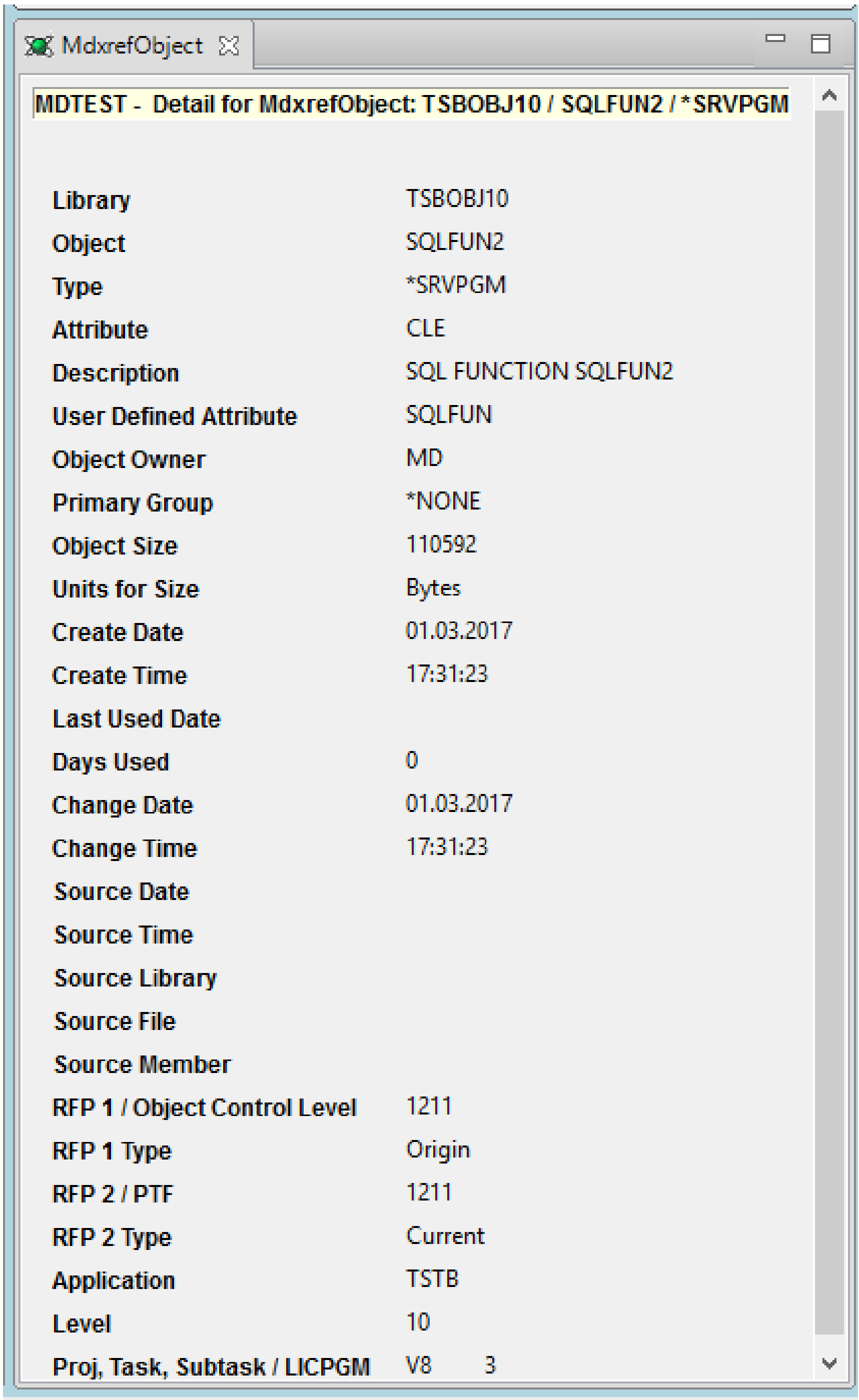
Right-Click on a row to see the enabled options available for that specific object attribute. The complete list of all object attribute menu options and explanations follows.

If the selected object had the library entry of '*LIBL' and the object exists in more than one library within the cross-reference database, the user is prompted to select the desired library from a list before the result is displayed.

In order to always show and use the first library in the library list where an object is found, then filter the Application and Level to specific values and set 1st in Libl Only to true.

8.1.1 MDXREF Object Details

Left-Click on a row to bring up the ***MdxrefObject*** view to see a large amount of system and SQL details about the object. Or select a row and right-click to select the option View or View in new tab to also bring up the ***MdxrefObject*** view.



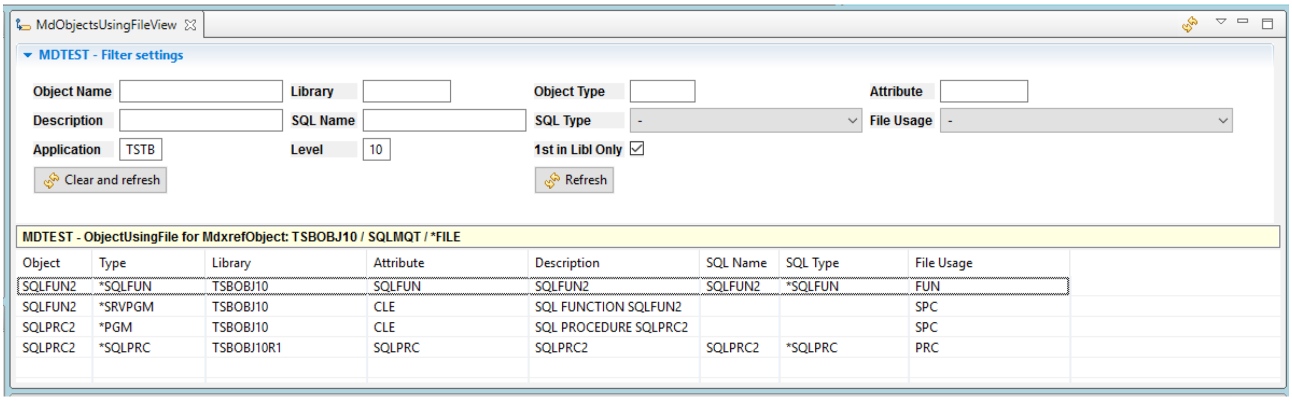
MDTEST - Detail for MdxrefObject: TSBOBJ10 / SQLFUN2 / *SRVPGM

Library	TSBOBJ10
Object	SQLFUN2
Type	*SRVPGM
Attribute	CLE
Description	SQL FUNCTION SQLFUN2
User Defined Attribute	SQLFUN
Object Owner	MD
Primary Group	*NONE
Object Size	110592
Units for Size	Bytes
Create Date	01.03.2017
Create Time	17:31:23
Last Used Date	
Days Used	0
Change Date	01.03.2017
Change Time	17:31:23
Source Date	
Source Time	
Source Library	
Source File	
Source Member	
RFP 1 / Object Control Level	1211
RFP 1 Type	Origin
RFP 2 / PTF	1211
RFP 2 Type	Current
Application	TSTB
Level	10
Proj, Task, Subtask / LICPGM	V8 3

If stamping is not being used to stamp all object fields available to be stamped in the application, there could be values in those fields that are not there from MDCMS/MDOpen object stamping.

8.1.2 Objects using File

This option will display a list ***MDObjectsUsingFileView*** of all objects that use the specific file as well as how the file is used by the object.



Object Name		Library		Object Type		Attribute	
Description		SQL Name		SQL Type	-	File Usage	-
Application	TSTB	Level	10	1st in Libl Only	☑		

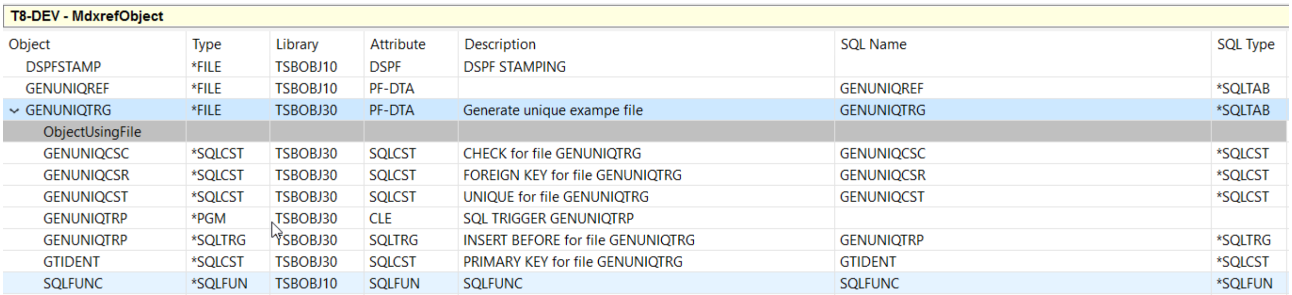
MDTEST - ObjectUsingFile for MdxrefObject: TSBOBJ10 / SQLMQT / *FILE

Object	Type	Library	Attribute	Description	SQL Name	SQL Type	File Usage
SQLFUN2	*SQLFUN	TSBOBJ10	SQLFUN	SQLFUN2	SQLFUN2	*SQLFUN	FUN
SQLFUN2	*SRVPGM	TSBOBJ10	CLE	SQL FUNCTION SQLFUN2			SPC
SQLPRC2	*PGM	TSBOBJ10	CLE	SQL PROCEDURE SQLPRC2			SPC
SQLPRC2	*SQLPRC	TSBOBJ10R1	SQLPRC	SQLPRC2	SQLPRC2	*SQLPRC	PRC

8.1.3 Objects Using File (Inline/Full)

Same as option Objects using File except the usage list remains in this same list

When the Full option is used, expansions will automatically occur for objects using files that use the selected file. For example, if a physical file is selected, it will show all objects directly using the physical file and branch out to all objects using any logical files over the physical file.
If the Full processing is taking too long, click the cancel button to abort the auto-expand process.



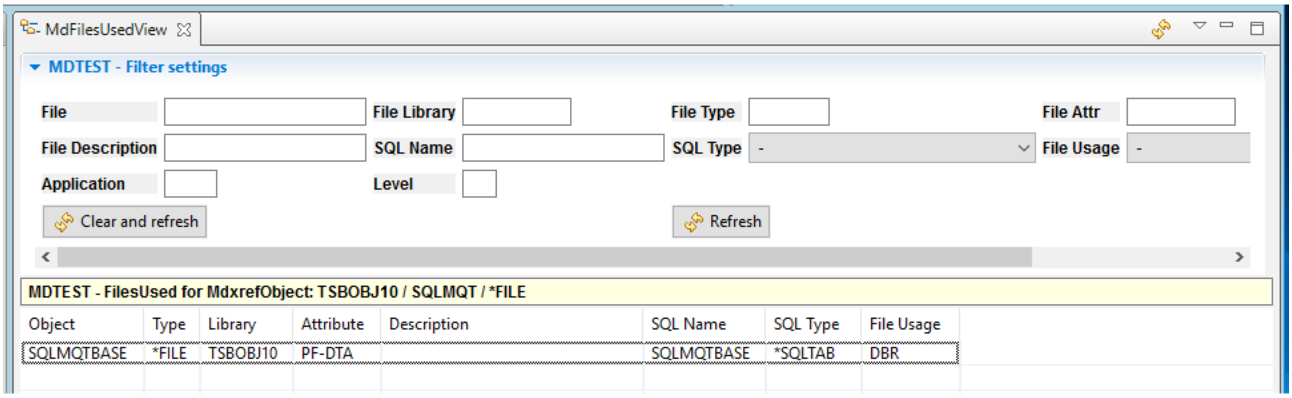
T8-DEV - MdxrefObject

Object	Type	Library	Attribute	Description	SQL Name	SQL Type
DSPFSTAMP	*FILE	TSBOBJ10	DSPF	DSPF STAMPING		
GENUNIQREF	*FILE	TSBOBJ10	PF-DTA		GENUNIQREF	*SQLTAB
GENUNIQTRG	*FILE	TSBOBJ30	PF-DTA	Generate unique exampe file	GENUNIQTRG	*SQLTAB
ObjectUsingFile						
GENUNIQCSC	*SQLCST	TSBOBJ30	SQLCST	CHECK for file GENUNIQTRG	GENUNIQCSC	*SQLCST
GENUNIQCSR	*SQLCST	TSBOBJ30	SQLCST	FOREIGN KEY for file GENUNIQTRG	GENUNIQCSR	*SQLCST
GENUNIQCST	*SQLCST	TSBOBJ30	SQLCST	UNIQUE for file GENUNIQTRG	GENUNIQCST	*SQLCST
GENUNIQTRP	*PGM	TSBOBJ30	CLE	SQL TRIGGER GENUNIQTRP		
GENUNIQTRP	*SQLTRG	TSBOBJ30	SQLTRG	INSERT BEFORE for file GENUNIQTRG	GENUNIQTRP	*SQLTRG
GTIDENT	*SQLCST	TSBOBJ30	SQLCST	PRIMARY KEY for file GENUNIQTRG	GTIDENT	*SQLCST
SQLFUNC	*SQLFUN	TSBOBJ10	SQLFUN	SQLFUNC	SQLFUNC	*SQLFUN

8.1.4 Files Used

Valid for many types of objects.

This option will display a list ***MdFilesUsedView*** of all files that are used by the specific object as well as how the file is used.



SQL Type: -

File Usage: -

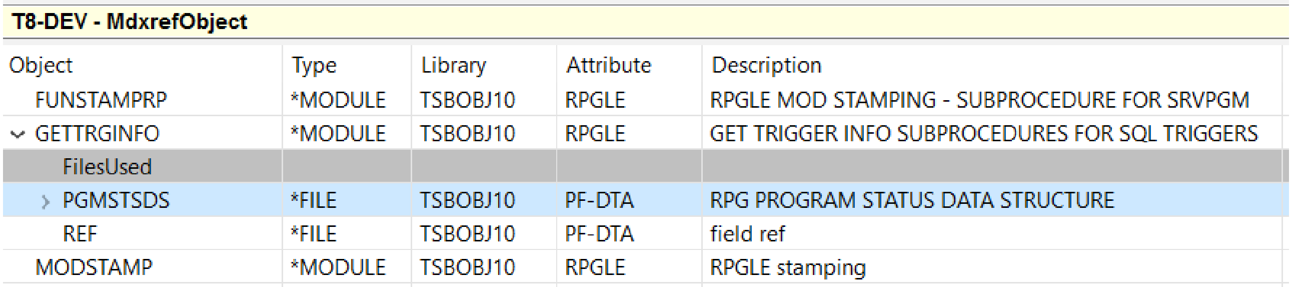
MDTEST - FilesUsed for MdxrefObject: TSBOBJ10 / SQLMQT / *FILE

Object	Type	Library	Attribute	Description	SQL Name	SQL Type	File Usage
SQLMQTBASE	*FILE	TSBOBJ10	PF-DTA		SQLMQTBASE	*SQLTAB	DBR

8.1.5 Files Used (Inline/Full)

Same as option Files Used except the usage list remains in this same list.

When the Full option is used, expansions will automatically occur for files used by files that use the selected file. For example, if a program is selected, it will show all files directly used by the program and branch out to all physical files used by logical files that are used by the program.
If the Full processing is taking too long, click the cancel button to abort the auto-expand process.

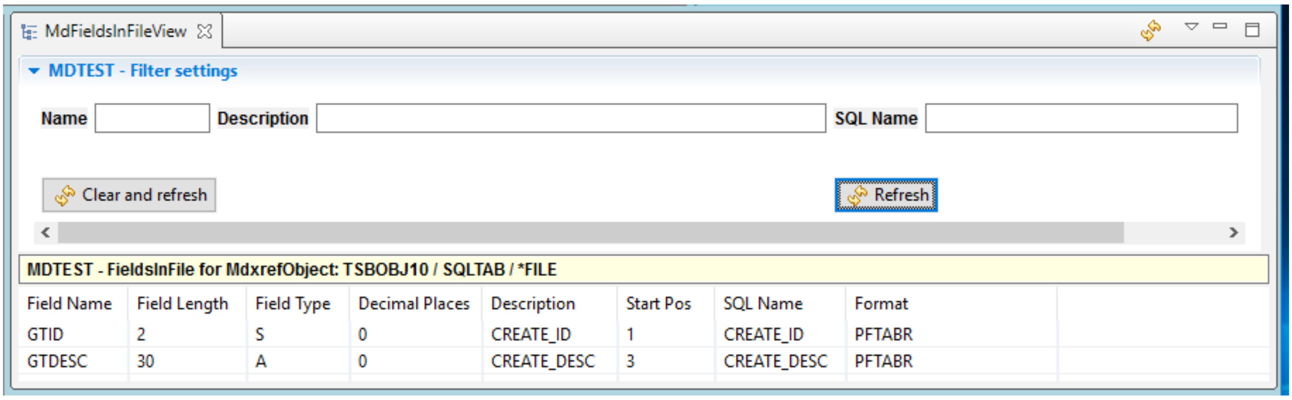


T8-DEV - MdxrefObject

Object	Type	Library	Attribute	Description
FUNSTAMPRP	*MODULE	TSBOBJ10	RPGLE	RPGLE MOD STAMPING - SUBPROCEDURE FOR SRVPGM
⌄ GETTRGINFO	*MODULE	TSBOBJ10	RPGLE	GET TRIGGER INFO SUBPROCEDURES FOR SQL TRIGGERS
FilesUsed				
› PGMSTSDS	*FILE	TSBOBJ10	PF-DTA	RPG PROGRAM STATUS DATA STRUCTURE
REF	*FILE	TSBOBJ10	PF-DTA	field ref
MODSTAMP	*MODULE	TSBOBJ10	RPGLE	RPGLE stamping

8.1.6 Fields in File

Valid option for physical, logical and SQL files only.
This option will display a list ***MdFieldsInFileView*** of all fields and their attributes for a file.



MDTEST - FieldsInFile for MdxrefObject: TSBOBJ10 / SQLTAB / *FILE

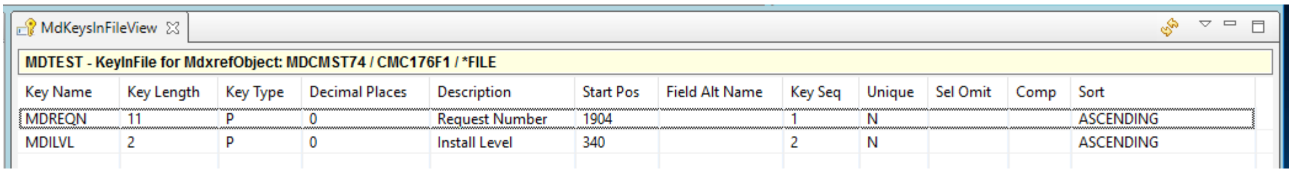
Field Name	Field Length	Field Type	Decimal Places	Description	Start Pos	SQL Name	Format
GTID	2	S	0	CREATE_ID	1	CREATE_ID	PFTABR
GTDESC	30	A	0	CREATE_DESC	3	CREATE_DESC	PFTABR

The Source containing Field context menu option is available for a row in the list. This option causes MDXREF to collect the source for all objects either directly using the file or a logical over the file, along with all copybooks used by dependent programs. It then inspects the code for any non-commented lines containing the field and then presents the results.

8.1.7 Keys in File

Valid option for physical, logical and SQL index files only.

This option will display the list ***MdKeysInFileView*** of field keys and their attributes for a file. Also, if for a logical file, all select/omit rules will also be listed.



MDTEST - KeyInFile for MdxrefObject: MDCMST74 / CMC176F1 / *FILE

Key Name	Key Length	Key Type	Decimal Places	Description	Start Pos	Field Alt Name	Key Seq	Unique	Sel Omit	Comp	Sort
MDREQN	11	P	0	Request Number	1904		1	N			ASCENDING
MDILVL	2	P	0	Install Level	340		2	N			ASCENDING

The Source containing Field context menu option is available for a row in the list. This option causes MDXREF to collect the source for all objects either directly using the file or a logical over the file, along with all copybooks used by dependent programs. It then inspects the code for any non-commented lines containing the field and then presents the results.

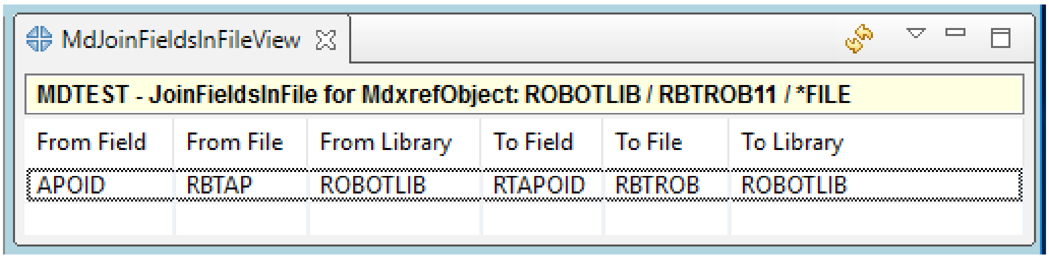
8.1.8 Join Fields in File

Valid option for logical files only.

This option will display the list ***MdJoinFieldsInFileView*** of join fields and duplicate sequencing fields for a join file.

If the selected file had the library entry of '*LIBL' and the file exists in more than one library within the cross-reference database, the user is prompted to select the desired library from a list before the fields are displayed.

The listing is sorted in the same order that the join fields were defined. Fields used to order duplicate records are designated with the To Field name *DUPSEQ.



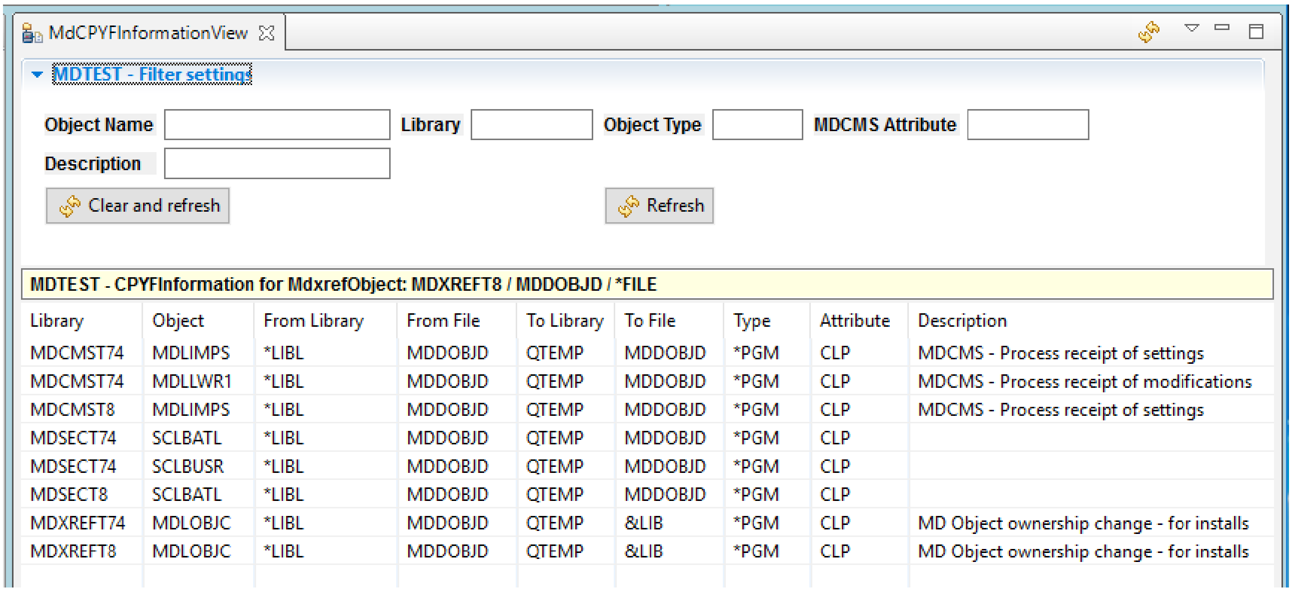
MDTEST - JoinFieldsInFile for MdxrefObject: ROBOTLIB / RBTROB11 / *FILE

From Field	From File	From Library	To Field	To File	To Library
APOID	RBTAP	ROBOTLIB	RTAPOID	RBTROB	ROBOTLIB

8.1.9 CPYF/OVRDBF Information

Valid option for files only.

This option will display the list ***MdCPYFInformationView*** of all CL programs or robot commands that perform an override or a CPYF from the selected file to another file. CL programs will only be listed if the source is specified as Observable.



MDTEST - CPYFInformation for MdxrefObject: MDXREFT8 / MDDOBJD / *FILE

Library	Object	From Library	From File	To Library	To File	Type	Attribute	Description
MDCMST74	MDLIMPS	*LIBL	MDDOBJD	QTEMP	MDDOBJD	*PGM	CLP	MDCMS - Process receipt of settings
MDCMST74	MDLLWR1	*LIBL	MDDOBJD	QTEMP	MDDOBJD	*PGM	CLP	MDCMS - Process receipt of modifications
MDCMST8	MDLIMPS	*LIBL	MDDOBJD	QTEMP	MDDOBJD	*PGM	CLP	MDCMS - Process receipt of settings
MDSECT74	SCLBATL	*LIBL	MDDOBJD	QTEMP	MDDOBJD	*PGM	CLP	
MDSECT74	SCLBUSR	*LIBL	MDDOBJD	QTEMP	MDDOBJD	*PGM	CLP	
MDSECT8	SCLBATL	*LIBL	MDDOBJD	QTEMP	MDDOBJD	*PGM	CLP	
MDXREFT74	MDLOBJC	*LIBL	MDDOBJD	QTEMP	&LIB	*PGM	CLP	MD Object ownership change - for installs
MDXREFT8	MDLOBJC	*LIBL	MDDOBJD	QTEMP	&LIB	*PGM	CLP	MD Object ownership change - for installs

8.1.10 Journal Analysis

Valid option for Physical Files only.

This option allows you to view the journal records for insert, update and delete transactions for rows in a physical file in a customizable and easy-to-read format.

Header Filters:

Start Date	The minimum date of transactions to consider. The earliest actual possible date of a transaction in the report is either the attachment date of the first journal receiver in the current chain, the attachment date of the From Receiver (if entered) or the creation date of the physical file, whichever is newest.
Start Time	The minimum time of day of transactions to consider.
End Date	The maximum date of transactions to consider
End Time	The maximum time of day of transactions to consider
User	The job user that performed the transaction
Job Name	The job name of the job that performed the transaction
Program	The IBM i program that performed the transaction
Action	Insert, Update, Delete or any of the above
Journal / Library	Will be by default the Journal currently observing transactions for the file. Enter a value if inspecting an older set of transactions
From/To Receiver	By default, the current chain of receivers will be used. Use these filters to limit the receivers or to use a different set of receivers.

List Filters:

Include	Select a row to include that field as a column in the report. Click all to select/unselect all fields. Even if a field isn't selected for display, transactions can still be filtered by that field's values
Minimum Value	If not blank, transactions will be limited to values >= the entered value
Maximum Value	If not blank, transactions will be limited to values <= the entered value
Modification Required	No – a transaction will be considered, even if the value for the field didn't change Yes – a transaction will only be considered if the value for the field changed. A change is an insert with a value, a delete with a value, or an update from one value to another. If Modification Required is used for multiple fields, a transaction will be considered if at least 1 of the field's values changed
Yes from within Range to outside Range	a transaction will be considered if the prior value was within the range specified by the Minimum/Maximum Value filters prior to the transaction, but no longer within the range after the transaction.
Yes from outside Range to within Range	a transaction will be considered if the prior value was outside the range specified by the Minimum/Maximum Value filters prior to the transaction, but within the range after the transaction.

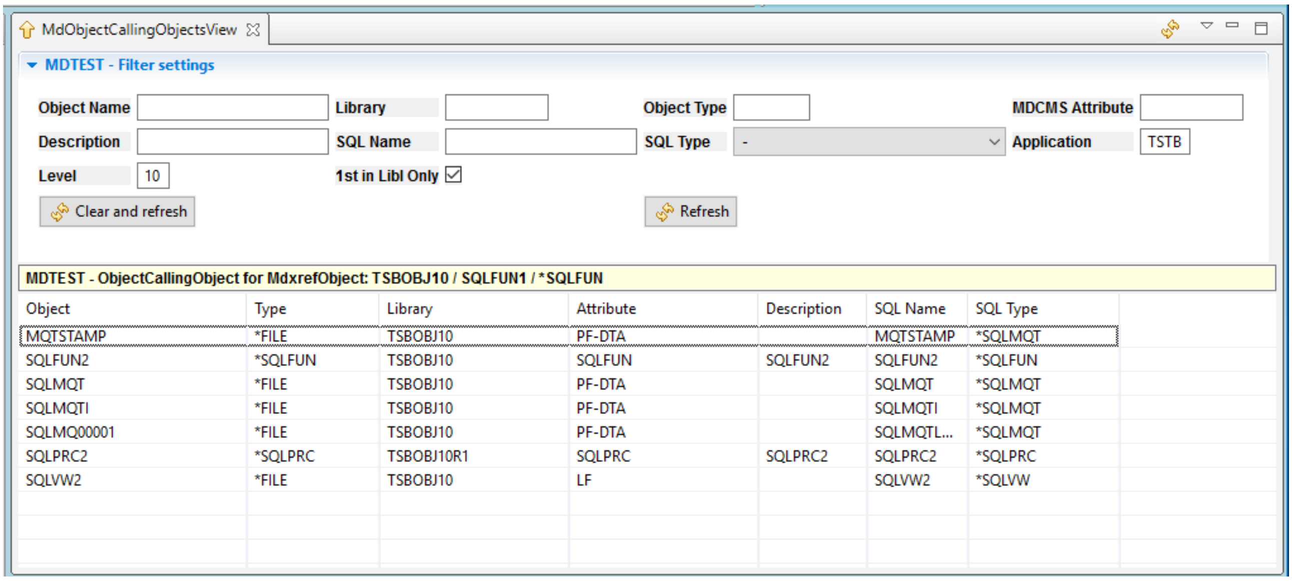
Once all filters are set, click the Create Report button in the header. A job will be submitted to batch to generate the report and a pop-up message will appear with the job is complete. The results are available in Session->Report Output

8.1.11 Up

Valid for many types of objects

This option will display a list of all objects that call or somehow invoke the selected object.
For example, if 'Up' was requested for program JORDCB1, a list would be displayed showing all programs that call JORDCB1, all commands that invoke JORDCB1, or all menus that JORDCB1 is called from.

SQL object types *SQLFUN and *SQLPRC can be the called object types used by calling object types *SQLFUN, *SQLMQT, *SQLPRC and *SQLVW.



Object Name		Library		Object Type		MDCMS Attribute	
Description		SQL Name		SQL Type	-	Application	TSTB
Level	10	1st in Libl Only	☑				

MDTEST - ObjectCallingObject for MdxrefObject: TSBOBJ10 / SQLFUN1 / *SQLFUN

Object	Type	Library	Attribute	Description	SQL Name	SQL Type
MQTSTAMP	*FILE	TSBOBJ10	PF-DTA		MQTSTAMP	*SQLMQT
SQLFUN2	*SQLFUN	TSBOBJ10	SQLFUN	SQLFUN2	SQLFUN2	*SQLFUN
SQLMQT	*FILE	TSBOBJ10	PF-DTA		SQLMQT	*SQLMQT
SQLMQTI	*FILE	TSBOBJ10	PF-DTA		SQLMQTI	*SQLMQT
SQLMQ00001	*FILE	TSBOBJ10	PF-DTA		SQLMQTL...	*SQLMQT
SQLPRC2	*SQLPRC	TSBOBJ10R1	SQLPRC	SQLPRC2	SQLPRC2	*SQLPRC
SQLVW2	*FILE	TSBOBJ10	LF		SQLVW2	*SQLVW

8.1.12 Up Inline / Full Call Stack Up

Same as option Up except the call stack remains in this same list.

When the Full option is used, expansions will automatically occur for objects that call the selected object, and then continue to the objects calling those objects, etc.

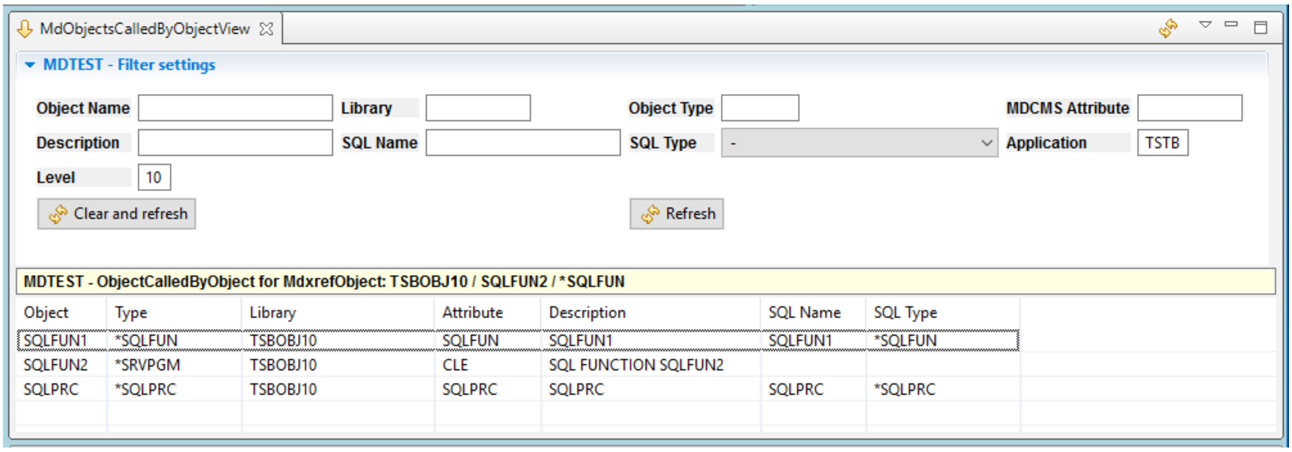
If the Full processing is taking too long, click the cancel button to abort the auto-expand process.

8.1.13 Down

Valid for many types of objects

This option will display a list of all objects that are called by the selected object.
For example, if 'Down' was requested for program JORDCB1, a list would be displayed showing all programs that JORDCB1 calls, all user commands that JORDCB1 invokes, all menus that JORDCB1 starts, or all queries that JORDCB1 runs.

SQL object types *SQLMQT, *SQLVW, *SQLFUN and *SQLPRC can be the calling object types of called object types *SQLFUN and *SQLPRC.



Object Name		Library		Object Type		MDCMS Attribute	
Description		SQL Name		SQL Type	-	Application	TSTB
Level	10						

MDTEST - ObjectCalledByObject for MdxrefObject: TSBOBJ10 / SQLFUN2 / *SQLFUN

Object	Type	Library	Attribute	Description	SQL Name	SQL Type
SQLFUN1	*SQLFUN	TSBOBJ10	SQLFUN	SQLFUN1	SQLFUN1	*SQLFUN
SQLFUN2	*SRVPGM	TSBOBJ10	CLE	SQL FUNCTION SQLFUN2		
SQLPRC	*SQLPRC	TSBOBJ10	SQLPRC	SQLPRC	SQLPRC	*SQLPRC

8.1.14 Down Inline / Full Call Stack Down

Same as option Down except the call stack remains in this same list.

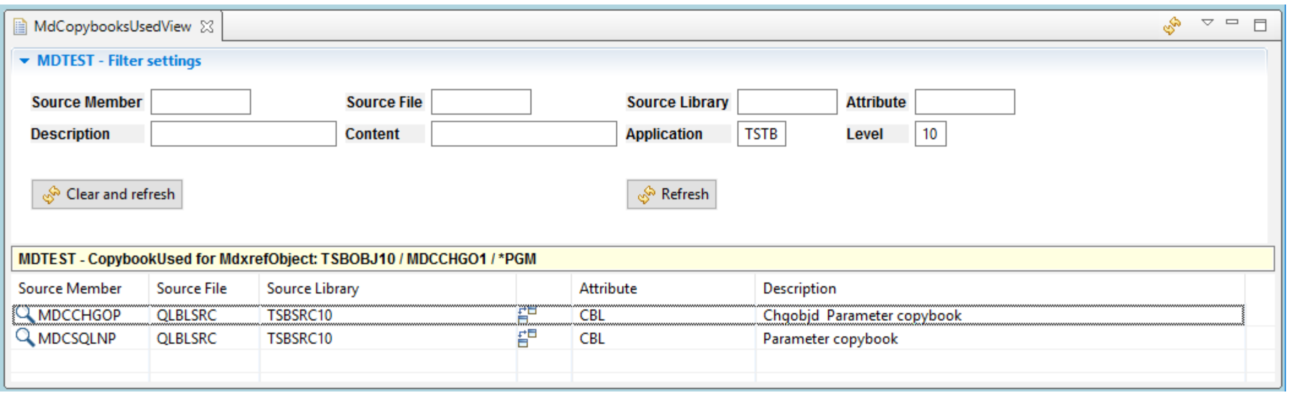
When the Full option is used, expansions will automatically occur for objects that are called by the selected object, and then continue to the objects called by those objects, etc.

If the Full processing is taking too long, click the cancel button to abort the auto-expand process.

8.1.15 Copybooks Used

Valid for CLP, COBOL and RPG programs (ILE or OPM) and modules if the source for those objects is managed by MDCMS.

This option lists all source members that are used as copybooks in the program or module.



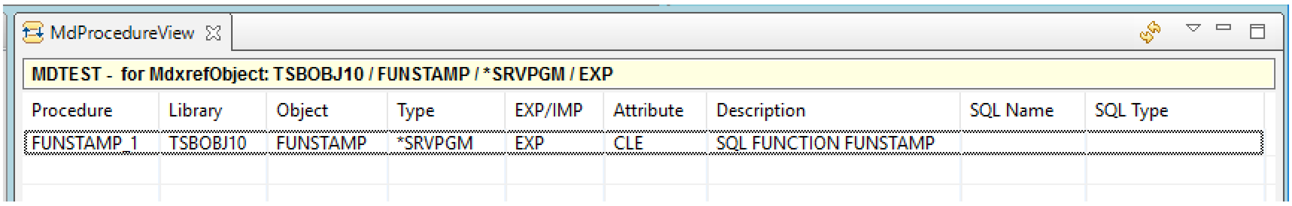
Source Member
Source File
Source Library
Attribute
Description
Content
Application TSTB
Level 10

MDTEST - CopybookUsed for MdxrefObject: TSBOBJ10 / MDCCHGO1 / *PGM

Source Member	Source File	Source Library		Attribute	Description
MDCCHGOP	QLBLSRC	TSBSRC10		CBL	Chgobjd Parameter copybook
MDCSQLNP	QLBLSRC	TSBSRC10		CBL	Parameter copybook

8.1.16 Exported Procedures

This option will display a list ***MdProcedureView*** of all procedures exported by the ILE Module or Service Program.

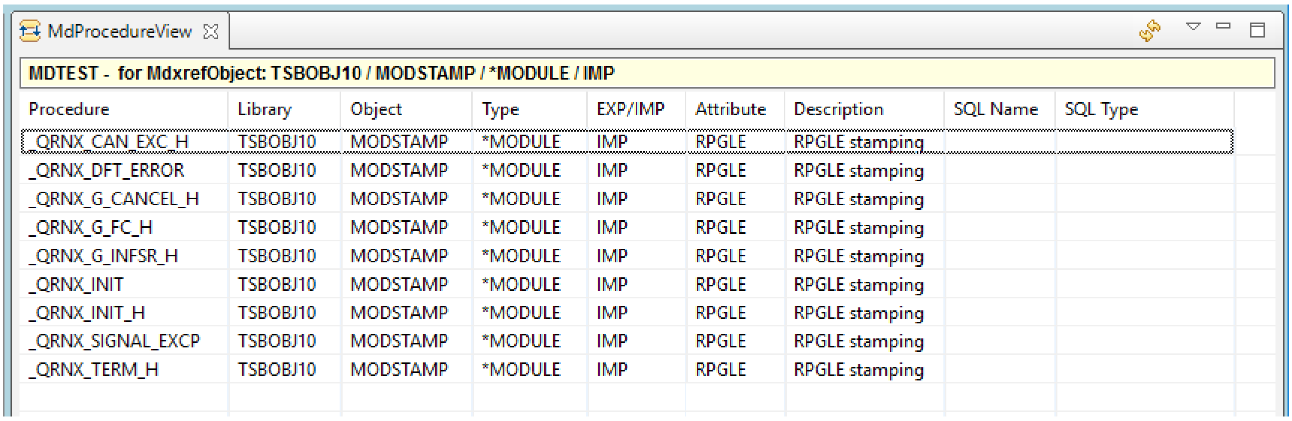


MDTEST - for MdxrefObject: TSBOBJ10 / FUNSTAMP / *SRVPGM / EXP

Procedure	Library	Object	Type	EXP/IMP	Attribute	Description	SQL Name	SQL Type
FUNSTAMP_1	TSBOBJ10	FUNSTAMP	*SRVPGM	EXP	CLE	SQL FUNCTION FUNSTAMP		

8.1.17 Imported Procedures

This option will display a list ***MdProcedureView*** of all procedures imported by the ILE Module.



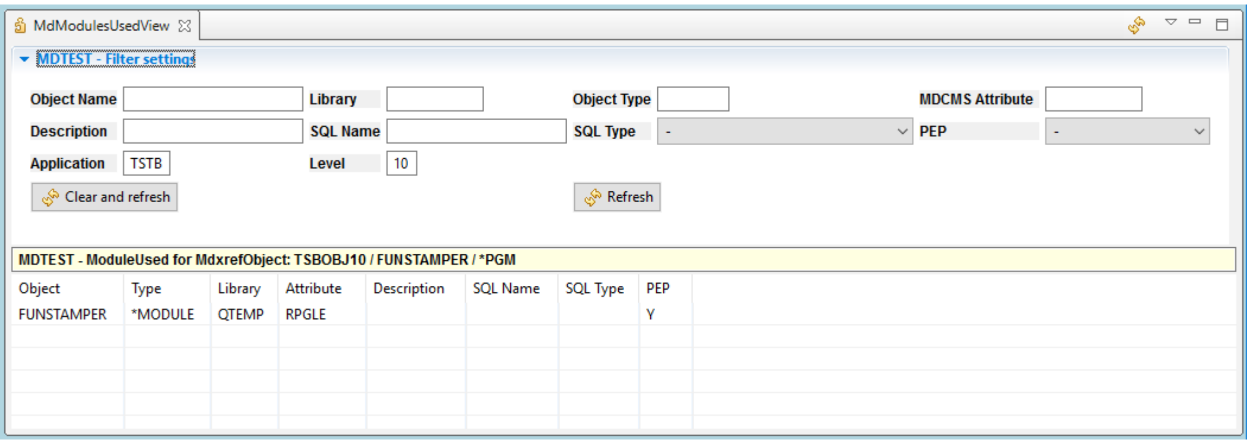
MDTEST - for MdxrefObject: TSBOBJ10 / MODSTAMP / *MODULE / IMP

Procedure	Library	Object	Type	EXP/IMP	Attribute	Description	SQL Name	SQL Type
_QRNX_CAN_EXC_H	TSBOBJ10	MODSTAMP	*MODULE	IMP	RPGLE	RPGLE stamping		
_QRNX_DFT_ERROR	TSBOBJ10	MODSTAMP	*MODULE	IMP	RPGLE	RPGLE stamping		
_QRNX_G_CANCEL_H	TSBOBJ10	MODSTAMP	*MODULE	IMP	RPGLE	RPGLE stamping		
_QRNX_G_FC_H	TSBOBJ10	MODSTAMP	*MODULE	IMP	RPGLE	RPGLE stamping		
_QRNX_G_INFSR_H	TSBOBJ10	MODSTAMP	*MODULE	IMP	RPGLE	RPGLE stamping		
_QRNX_INIT	TSBOBJ10	MODSTAMP	*MODULE	IMP	RPGLE	RPGLE stamping		
_QRNX_INIT_H	TSBOBJ10	MODSTAMP	*MODULE	IMP	RPGLE	RPGLE stamping		
_QRNX_SIGNAL_EXCP	TSBOBJ10	MODSTAMP	*MODULE	IMP	RPGLE	RPGLE stamping		
_QRNX_TERM_H	TSBOBJ10	MODSTAMP	*MODULE	IMP	RPGLE	RPGLE stamping		

8.1.18 Modules Used

Valid for programs and service programs.

This option lists all modules that are bound by the program. The PEP flag designates which module contains the Program Entry Procedure (PEP) for the program.



Object Name		Library		Object Type		MDCMS Attribute	
Description		SQL Name		SQL Type	-	PEP	-
Application	TSTB	Level	10				

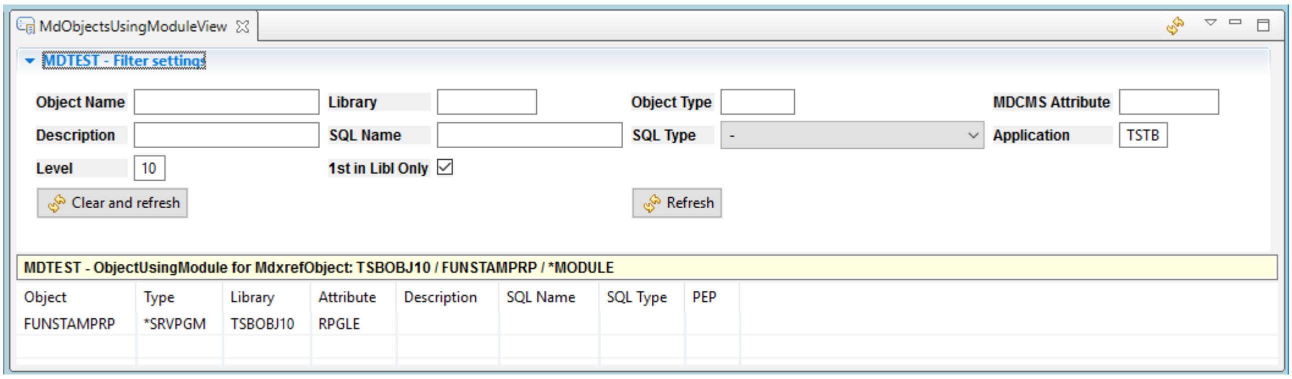
MDTEST - ModuleUsed for MdxrefObject: TSBOBJ10 / FUNSTAMPER / *PGM

Object	Type	Library	Attribute	Description	SQL Name	SQL Type	PEP
FUNSTAMPER	*MODULE	QTEMP	RPGLE				Y

8.1.19 Objects using Module

Valid option for ILE Modules only.

This option will display a list of all programs or service programs that use a specific ILE module.



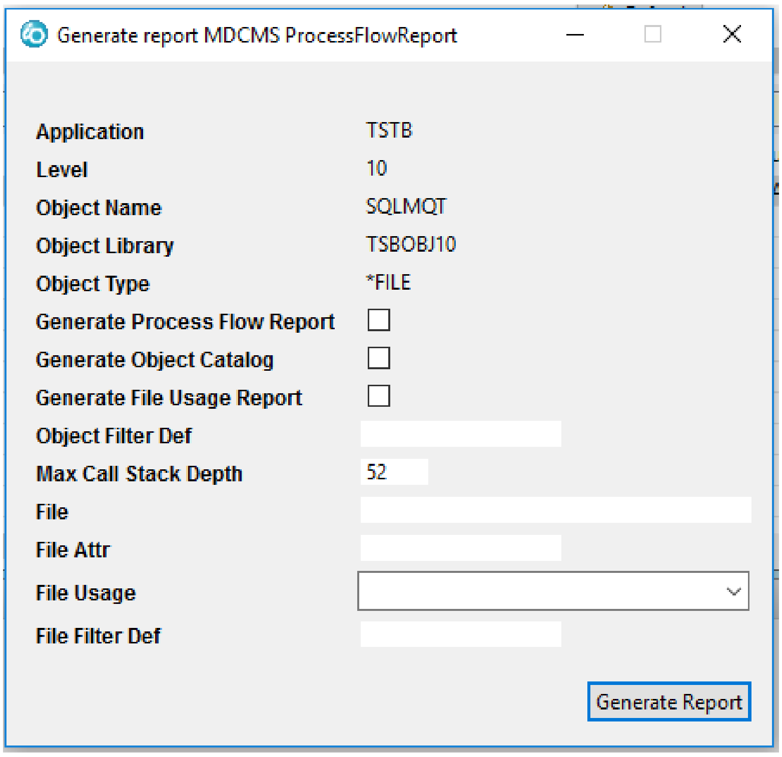
Object Name		Library		Object Type		MDCMS Attribute	
Description		SQL Name		SQL Type	-	Application	TSTB
Level	10	1st in Libl Only	☑				

MDTEST - ObjectUsingModule for MdxrefObject: TSBOBJ10 / FUNSTAMPRP / *MODULE

Object	Type	Library	Attribute	Description	SQL Name	SQL Type	PEP
FUNSTAMPRP	*SRVPGM	TSBOBJ10	RPGLE				

8.1.20 Process Flow Report

Valid option for menus, commands, functions, materialized query tables, modules, procedures, programs, scheduled jobs, table objects and views.



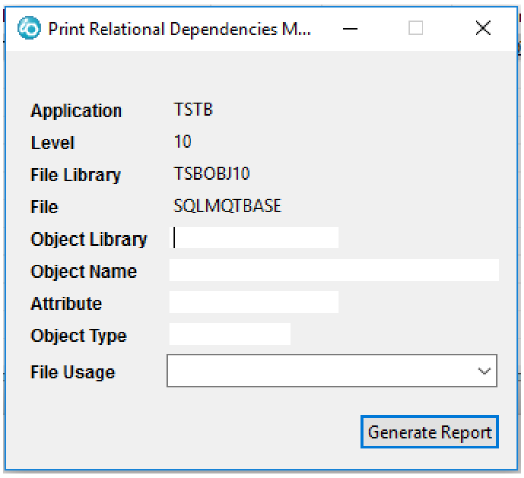
Application	TSTB
Level	10
Object Name	SQLMQT
Object Library	TSBOBJ10
Object Type	*FILE
Generate Process Flow Report	□
Generate Object Catalog	□
Generate File Usage Report	□
Object Filter Def	
Max Call Stack Depth	52
File	
File Attr	
File Usage	
File Filter Def	

Complete descriptions of the Generate Process Flow Report for an object dialog can be found in the MDXREF user manual.

8.1.21 Print Relational Dependencies

Valid option for files only.

This option will result in a formatted report listing all files, programs, queries, etc. that use the file. In addition, if a file uses the base file, all files, programs, queries, etc. are also included for that file.



Application	TSTB
Level	10
File Library	TSBOBJ10
File	SQLMQTBASE
Object Library	
Object Name	
Attribute	
Object Type	
File Usage	

8.1.22 View Source

Highlight 1 row in the list and right-click to select the option View Source to view the contents of the source member or IFS source file used to compile the object. If using RDI, the source will open in the LPEX editor in browse mode.

8.1.23 XREF Refresh for Objects

Valid for all objects. Note there isn't a view displayed for this option.

If an object has been manually changed or deleted since the last build of the MDXREF application database, the object's MDXREF information may be refreshed by using this option.

If the selected object had the library entry of '*LIBL' and the object exists in more than one library within the cross-reference database, the user is prompted to select the desired library from a list before the object's information is refreshed.

8.1.24 Debug – set service entry point

Valid for objects of type *PGM or *SRVPGM when the object attribute is CBLLE, CLE, CLLE, or RPGLE and MDOpen is used within Rational Developer for i.

When the option is selected, the IBM i Service Entry Points view provided by Rational Developer for i is opened and, if debugging is enabled for the program, an entry is inserted into the view.

Once inserted, the next time that any job belonging to the User ID value for the entry invokes the program, the job will be interrupted and debugging can be performed for the program.

If the job to debug belongs to a different user, modify the entry to change the user id.

If the job updates production data, the Eclipse preference Run/Debug->IBM i Debug->Update production files must be set to true.

8.1.25 X-Analysis Options

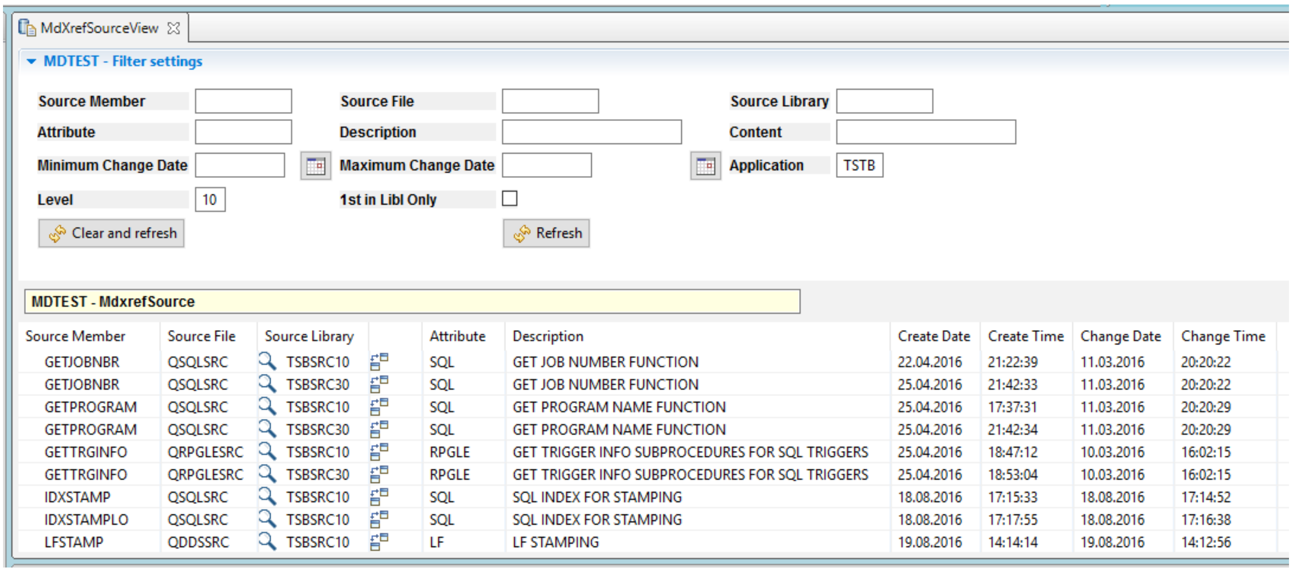
If the X-Analysis Plug-in is installed into the same instance of eclipse, and the version of X-Analysis is at least 13.0.0, then X-A lists and diagrams can be accessed seamlessly for Objects in the MDXREF views.

The first time for a connection that right-click is performed on an object to select an option, a connection is established from X-A to the IBM i partition. This process can take some time, depending on the speed of the network connection. If it is quite slow and you don't wish to use the X-A options, you can disable them from the Repository settings for the connection in the MDRepositoryView.

If enabled, once the X-A connection is established, the X-A options are listed at the bottom of the option list.

8.2 MDXREF Source

The MDXREF Source view provides the user with a filtered list of source members in all libraries that are built in the MDXREF cross reference product.



MdXrefSourceView

MDTEST - Filter settings

Source Member		Source File		Source Library	
Attribute		Description		Content	
Minimum Change Date		Maximum Change Date		Application	TSTB
Level	10	1st in Libl Only	□		

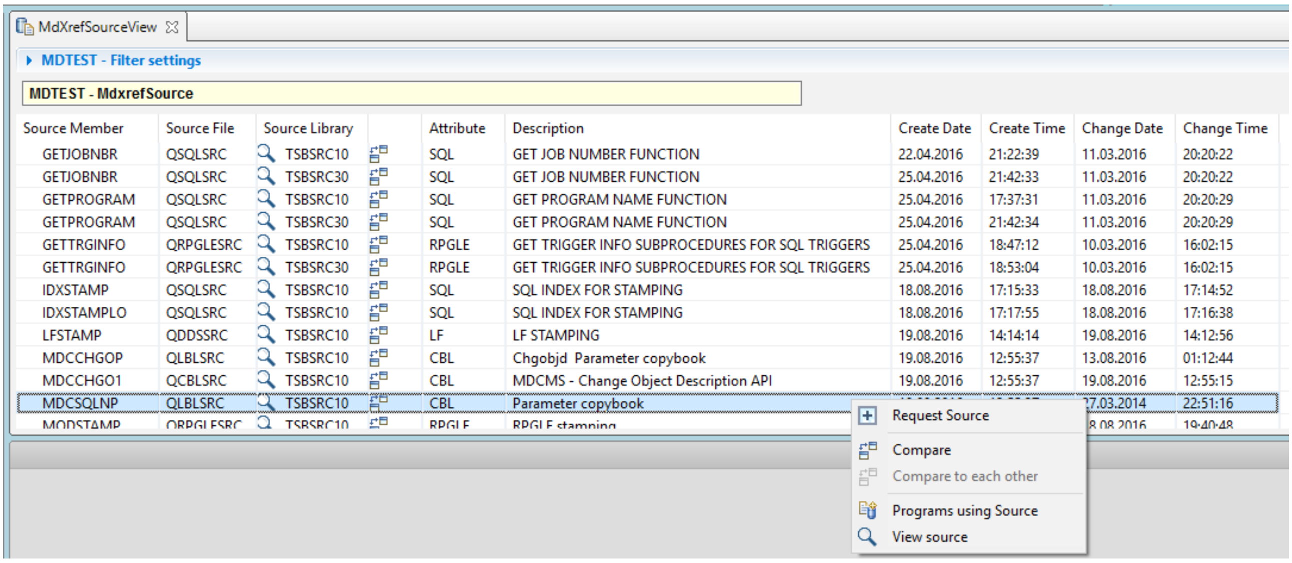
MDTEST - MdxrefSource

Source Member	Source File	Source Library		Attribute	Description	Create Date	Create Time	Change Date	Change Time
GETJOBNBR	QSQLSRC	🔍 TSBSRC10	⇄	SQL	GET JOB NUMBER FUNCTION	22.04.2016	21:22:39	11.03.2016	20:20:22
GETJOBNBR	QSQLSRC	🔍 TSBSRC30	⇄	SQL	GET JOB NUMBER FUNCTION	25.04.2016	21:42:33	11.03.2016	20:20:22
GETPROGRAM	QSQLSRC	🔍 TSBSRC10	⇄	SQL	GET PROGRAM NAME FUNCTION	25.04.2016	17:37:31	11.03.2016	20:20:29
GETPROGRAM	QSQLSRC	🔍 TSBSRC30	⇄	SQL	GET PROGRAM NAME FUNCTION	25.04.2016	21:42:34	11.03.2016	20:20:29
GETTRGINFO	QRPGLESRC	🔍 TSBSRC10	⇄	RPGLE	GET TRIGGER INFO SUBPROCEDURES FOR SQL TRIGGERS	25.04.2016	18:47:12	10.03.2016	16:02:15
GETTRGINFO	QRPGLESRC	🔍 TSBSRC30	⇄	RPGLE	GET TRIGGER INFO SUBPROCEDURES FOR SQL TRIGGERS	25.04.2016	18:53:04	10.03.2016	16:02:15
IDXSTAMP	QSQLSRC	🔍 TSBSRC10	⇄	SQL	SQL INDEX FOR STAMPING	18.08.2016	17:15:33	18.08.2016	17:14:52
IDXSTAMPLO	QSQLSRC	🔍 TSBSRC10	⇄	SQL	SQL INDEX FOR STAMPING	18.08.2016	17:17:55	18.08.2016	17:16:38
LFSTAMP	QDDSSRC	🔍 TSBSRC10	⇄	LF	LF STAMPING	19.08.2016	14:14:14	19.08.2016	14:12:56

🔍 - click this icon for a member to view the contents of the source member

⇄ - click this icon to perform a 2-way or 3-way compare of the source member to another member, IFS file or local file.

Right clicking on a source row entry within the ***MdXrefSourceView*** will display the following options:



MdXrefSourceView

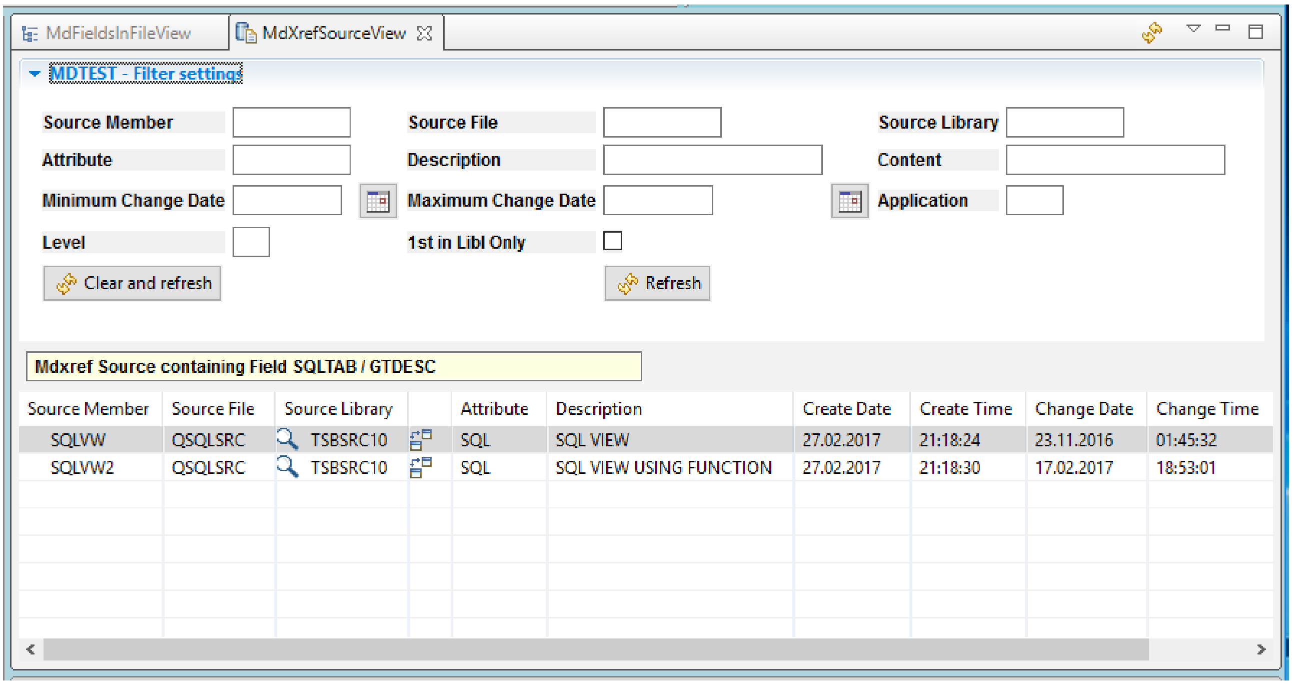
MDTEST - Filter settings

MDTEST - MdxrefSource

Source Member	Source File	Source Library		Attribute	Description	Create Date	Create Time	Change Date	Change Time
GETJOBNBR	QSQLSRC	🔍 TSBSRC10	⇄	SQL	GET JOB NUMBER FUNCTION	22.04.2016	21:22:39	11.03.2016	20:20:22
GETJOBNBR	QSQLSRC	🔍 TSBSRC30	⇄	SQL	GET JOB NUMBER FUNCTION	25.04.2016	21:42:33	11.03.2016	20:20:22
GETPROGRAM	QSQLSRC	🔍 TSBSRC10	⇄	SQL	GET PROGRAM NAME FUNCTION	25.04.2016	17:37:31	11.03.2016	20:20:29
GETPROGRAM	QSQLSRC	🔍 TSBSRC30	⇄	SQL	GET PROGRAM NAME FUNCTION	25.04.2016	21:42:34	11.03.2016	20:20:29
GETTRGINFO	QRPGLESRC	🔍 TSBSRC10	⇄	RPGLE	GET TRIGGER INFO SUBPROCEDURES FOR SQL TRIGGERS	25.04.2016	18:47:12	10.03.2016	16:02:15
GETTRGINFO	QRPGLESRC	🔍 TSBSRC30	⇄	RPGLE	GET TRIGGER INFO SUBPROCEDURES FOR SQL TRIGGERS	25.04.2016	18:53:04	10.03.2016	16:02:15
IDXSTAMP	QSQLSRC	🔍 TSBSRC10	⇄	SQL	SQL INDEX FOR STAMPING	18.08.2016	17:15:33	18.08.2016	17:14:52
IDXSTAMPLO	QSQLSRC	🔍 TSBSRC10	⇄	SQL	SQL INDEX FOR STAMPING	18.08.2016	17:17:55	18.08.2016	17:16:38
LFSTAMP	QDDSSRC	🔍 TSBSRC10	⇄	LF	LF STAMPING	19.08.2016	14:14:14	19.08.2016	14:12:56
MDCCHGOP	QLBLSRC	🔍 TSBSRC10	⇄	CBL	Chgobjd Parameter copybook	19.08.2016	12:55:37	13.08.2016	01:12:44
MDCCHGO1	QCBLSRC	🔍 TSBSRC10	⇄	CBL	MDCMS - Change Object Description API	19.08.2016	12:55:37	19.08.2016	12:55:15
MDCSQLNP	QLBLSRC	🔍 TSBSRC10	⇄	CBL	Parameter copybook	[illegible]	[illegible]	27.03.2014	22:51:16
MODSTAMP	QRPGLESRC	🔍 TSBSRC10	⇄	RPGLE	RPGLE stamping			8.08.2016	19:40:48

8.2.1 Source containing Field

When viewing the list of fields or keys in a file, or viewing the result list from the field search, an option is available to display a list ***MdXrefSourceView*** of all source containing the field.
When this option is taken, MDXREF collects the source for all objects either directly using the file or a logical over the file, along with all copybooks used by dependent programs. It then inspects the code for any non-commented lines containing the field and then presents the results.



Source Member		Source File		Source Library	
Attribute		Description		Content	
Minimum Change Date		Maximum Change Date		Application	
Level		1st in Libl Only	□		

Mdxref Source containing Field SQLTAB / GTDESC

Source Member	Source File	Source Library		Attribute	Description	Create Date	Create Time	Change Date	Change Time
SQLVW	QSQLSRC	TSBSRC10		SQL	SQL VIEW	27.02.2017	21:18:24	23.11.2016	01:45:32
SQLVW2	QSQLSRC	TSBSRC10		SQL	SQL VIEW USING FUNCTION	27.02.2017	21:18:30	17.02.2017	18:53:01

8.2.2 Source Compare

Highlight 1 row in the list and right-click to select the option Compare to display a dialog to capture information about a second and an optional third member to compare to.

Highlight 2 rows in the list and right-click to select the option Compare to each other to immediately perform a 2-way compare of the selected members.

8.2.3 Programs using Source

Highlight 1 row in the list that is a copybook source member and right-click to select the option Programs using Source to display the programs or modules using the copybook.

> The primary source for those objects must be managed by MDCMS for MDXREF to be aware of the copybook references.
>
> The primary source used to compile the object is not considered a copybook and is not listed on this screen.

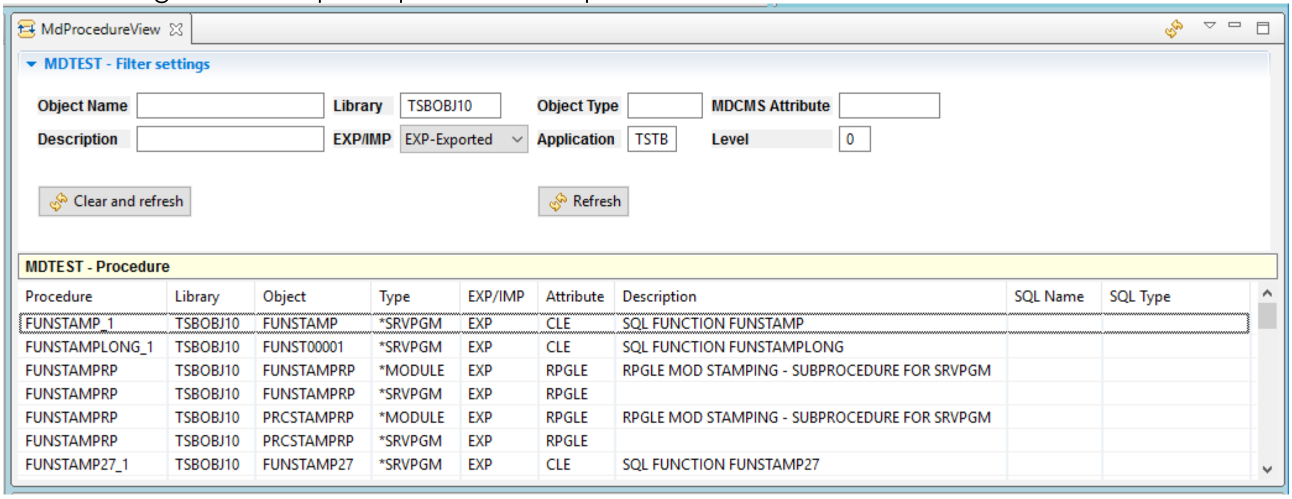
8.2.4 View Source

Highlight 1 row in the and right-click to select the option View Source to view the contents of the source member.

8.3 MDXREF Procedures

MdProcedureView is used to view the list of ILE Procedures (procedures imported or exported by ILE modules and procedures exported by service programs) matching the search criteria instead of listing objects. This is the same functionality as MDXREF using the object type filter of *ILEPRC.

The following is an example of procedures exported:



Object Name		Library	TSBOBJ10	Object Type		MDCMS Attribute	
Description		EXP/IMP	EXP-Exported	Application	TSTB	Level	0

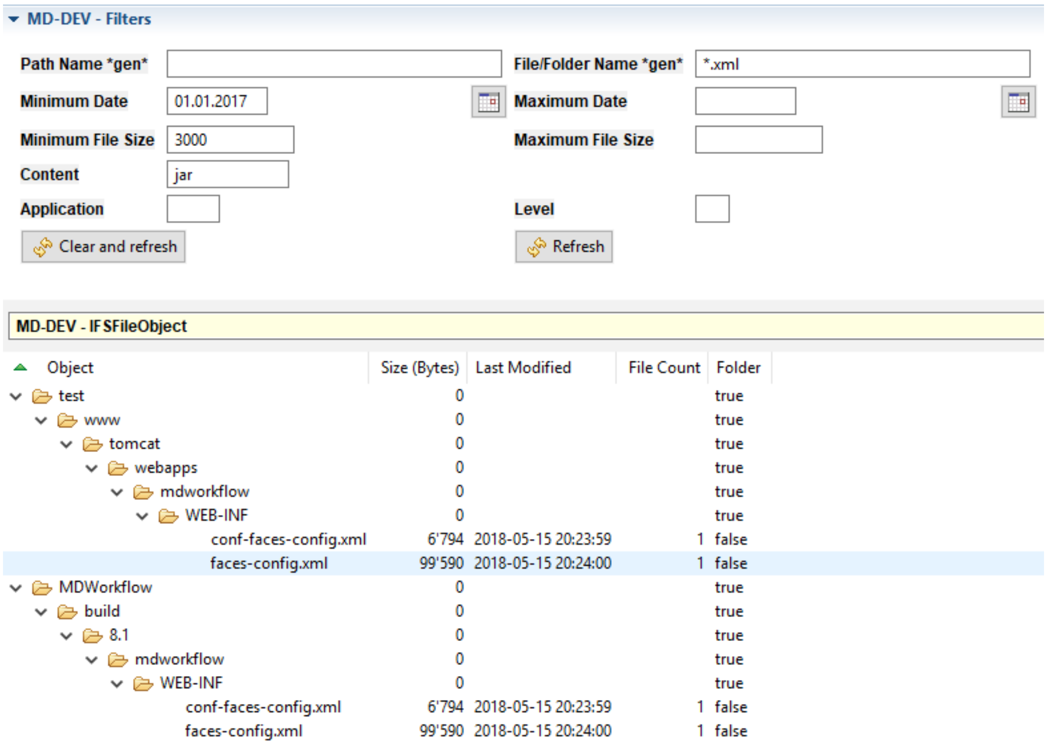
MDTEST - Procedure

Procedure	Library	Object	Type	EXP/IMP	Attribute	Description	SQL Name	SQL Type
FUNSTAMP_1	TSBOBJ10	FUNSTAMP	*SRVPGM	EXP	CLE	SQL FUNCTION FUNSTAMP		
FUNSTAMPLONG_1	TSBOBJ10	FUNST00001	*SRVPGM	EXP	CLE	SQL FUNCTION FUNSTAMPLONG		
FUNSTAMPRP	TSBOBJ10	FUNSTAMPRP	*MODULE	EXP	RPGLE	RPGLE MOD STAMPING - SUBPROCEDURE FOR SRVPGM		
FUNSTAMPRP	TSBOBJ10	FUNSTAMPRP	*SRVPGM	EXP	RPGLE			
FUNSTAMPRP	TSBOBJ10	PRCSTAMPRP	*MODULE	EXP	RPGLE	RPGLE MOD STAMPING - SUBPROCEDURE FOR SRVPGM		
FUNSTAMPRP	TSBOBJ10	PRCSTAMPRP	*SRVPGM	EXP	RPGLE			
FUNSTAMP27_1	TSBOBJ10	FUNSTAMP27	*SRVPGM	EXP	CLE	SQL FUNCTION FUNSTAMP27		

8.4 MDXREF IFS

If library value *IFS is included in the MDXREF build list, then MDXREF will contain information about every folder and file in the IFS file system. This then makes it very fast and simple for the user to search for IFS folders and files as well as to collect information about the size and quantity of folders and files.

The following is an example of a search for all xml files modified since the beginning of 2017 with a file size of at least 3000 bytes that contain the string “jar” in the body of the file:



Path Name *gen*:
File/Folder Name *gen*: *.xml
Minimum Date: 01.01.2017
Maximum Date:
Minimum File Size: 3000
Maximum File Size:
Content: jar
Application:
Level:

MD-DEV - IFSFileObject

Object	Size (Bytes)	Last Modified	File Count	Folder
test	0			true
www	0			true
tomcat	0			true
webapps	0			true
mdworkflow	0			true
WEB-INF	0			true
conf-faces-config.xml	6'794	2018-05-15 20:23:59	1	false
faces-config.xml	99'590	2018-05-15 20:24:00	1	false
MDWorkflow	0			true
build	0			true
8.1	0			true
mdworkflow	0			true
WEB-INF	0			true
conf-faces-config.xml	6'794	2018-05-15 20:23:59	1	false
faces-config.xml	99'590	2018-05-15 20:24:00	1	false

When not filtering (other than Path Name), MDOpen shows the total size and file count of each folder, regardless of the number of subfolders, to provide a very helpful summary of parent folder information.

Options

View	Open the file in browse mode. If the file doesn’t open in a preferred editor, the file associations can be modified in Window->Preferences->General->Editors->File Associations
Attribute Scripts	Open Remote Server Script View with rows filtered to selected Attribute. This option is only applicable for *REMOTE attribute types.
Linked Attributes	Define the Attributes that are linked to the selected Attribute. When a checkout occurs for this attribute, the developer will be prompted to check out objects for the linked attributes.
Request Object	Request checkout for one or more folders and files in MDCMS as *IFS or *REMOTE objects. See chapter Request Objects from MDXREF for more information.
Request Source	Request checkout for one or more files as IFS source for IBM i objects. See chapter Request IFS Source from MDXREF for more information.
Import IFS Folder	Request to migrate the contents of a folder in MDCMS. See chapter Import Objects or Source from IFS Folder for more information.
Compare	Bring up the compare dialog to perform a 2-way or 3-way comparison

	with some other element with the primary file pre-filled.
Compare to each other	Highlight 2 IFS files and then select this option to bring up the compare dialog to perform a 2-way or 3-way comparison with the primary and 2nd files pre-filled.
External References	View/Manage the list of IBM i native objects that are referenced by the selected IFS file. See chapter External References for more information.
External Reference Levels	View/Manage the configuration for External Cross-Referencing for the selected IFS path. See chapter External Reference Levels for more information.
Refresh MDXREF	Immediately refresh the contents and external references for the selected folder
Copy path to clipboard	place the entire path of the IFS folder or file, including the name of the folder or file, into the clipboard so that the path can be pasted somewhere else, such as in a document or command parameter.

8.5 MDXREF External References

This view shows all references to native IBM i or SQL objects from non-native source code in IFS, Git or SVN.

References can be automatically found by configuring the settings for External Reference Levels and External Reference File Extensions (see those chapters for more detail).

References can also be manually created from this view for any existing Git, SVN or IFS file with any IBM i/SQL object.

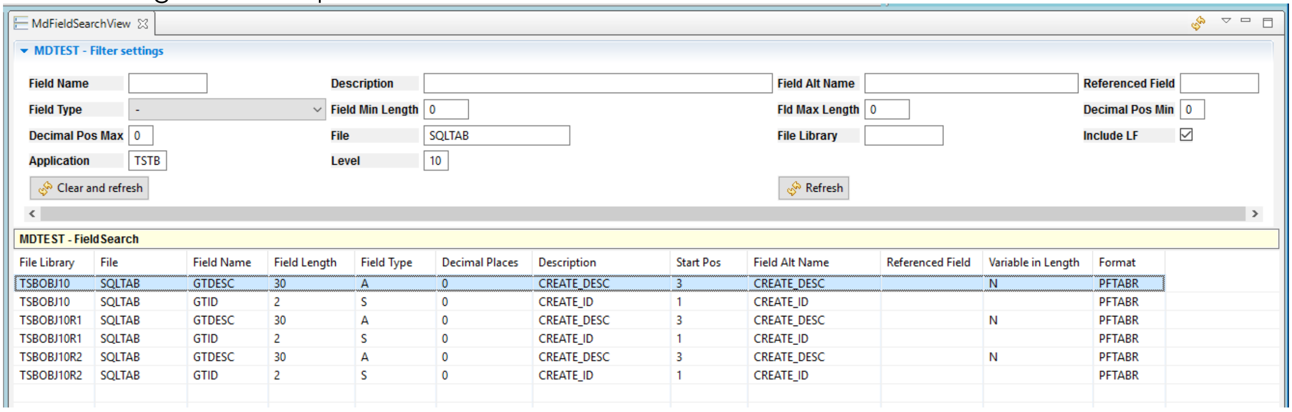
Options

View (🔍)	Open the IFS file in browse mode. If the file doesn't open in a preferred editor, the file associations can be modified in Window->Preferences->General->Editors->File Associations
Request IFS file	Request checkout for one or more files in MDCMS as *IFS or *REMOTE objects. See chapter Request Objects from MDXREF for more information.
Request Object	Request checkout for one or more native objects in MDCMS. See chapter Request Objects from MDXREF for more information.
Ignore	Set the reference to ignore. Ignored references won't appear during the pre-submit validation process for an RFP. If an ignored reference should no longer be ignored, left-click on the reference to edit the details and uncheck the Ignore box.

8.6 MDXREF Search for Fields

This list ***MdFieldSearchView*** is used to search for, and report on, all physical, logical or SQL files that contain a specific field or field attribute. Additionally, to inspect source code for the usage of specific fields via the context menu option Source containing Field.

The following is an example of an SQL table search:



MDTEST - Filter settings

Field Name		Description		Field Alt Name		Referenced Field	
Field Type	-	Field Min Length	0	Fld Max Length	0	Decimal Pos Min	0
Decimal Pos Max	0	File	SQLTAB	File Library		Include LF	☑
Application	TSTB	Level	10				

MDTEST - Field Search

File Library	File	Field Name	Field Length	Field Type	Decimal Places	Description	Start Pos	Field Alt Name	Referenced Field	Variable in Length	Format
TSBOBJ10	SQLTAB	GTDESC	30	A	0	CREATE_DESC	3	CREATE_DESC		N	PFTABR
TSBOBJ10	SQLTAB	GTID	2	S	0	CREATE_ID	1	CREATE_ID			PFTABR
TSBOBJ10R1	SQLTAB	GTDESC	30	A	0	CREATE_DESC	3	CREATE_DESC		N	PFTABR
TSBOBJ10R1	SQLTAB	GTID	2	S	0	CREATE_ID	1	CREATE_ID			PFTABR
TSBOBJ10R2	SQLTAB	GTDESC	30	A	0	CREATE_DESC	3	CREATE_DESC		N	PFTABR
TSBOBJ10R2	SQLTAB	GTID	2	S	0	CREATE_ID	1	CREATE_ID			PFTABR

9 Object Requests

9.1 Overview

The MDCMS Object Manager within MDOpen is a multi-function view that enables the user to access the functions required for making and installing modifications to objects.

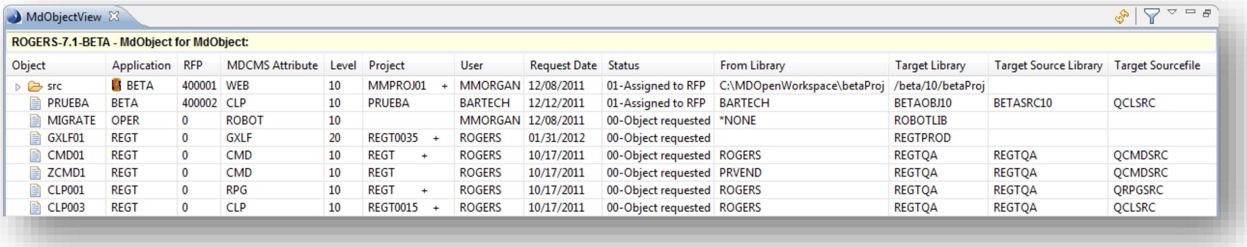
The Object Manager is used for:

- IBMi Source/Object request and retrieval.
- Reserving Object names for new objects.
- Grouping objects for installs.
- Identifying modification requests
- Requesting the installation of objects into application environments

All options discussed from the Object Requests view are also available when working with requests by RFP, Project or Task, allowing the developer to choose the preferred method of grouping object requests.

9.2 Object Request View

The Object Requests in MDOpen are displayed in the ***MdObjectView.*** The ***MdObjectView*** is accessed from the repository connection option **Object Requests**.



ROGERS-7.1-BETA - MdObject for MdObject:

Object	Application	RFP	MDCMS Attribute	Level	Project	User	Request Date	Status	From Library	Target Library	Target Source Library	Target Sourcefile
src	BETA	400001	WEB	10	MMPROJ01 +	MMORGAN	12/08/2011	01-Assigned to RFP	C:\MDOpenWorkspace\betaProj	/beta/10/betaProj		
PRUEBA	BETA	400002	CLP	10	PRUEBA	BARTECH	12/12/2011	01-Assigned to RFP	BARTECH	BETAOBJ10	BETASRC10	QCLSRC
MIGRATE	OPER	0	ROBOT	10		MMORGAN	12/08/2011	00-Object requested	*NONE	ROBOTLIB		
GXLF01	REGT	0	GXLF	20	REGT0035 +	ROGERS	01/31/2012	00-Object requested		REGTPROD		
CMD01	REGT	0	CMD	10	REGT +	ROGERS	10/17/2011	00-Object requested	ROGERS	REGTQA	REGTQA	QCMDSRC
ZCMD1	REGT	0	CMD	10	REGT	ROGERS	10/17/2011	00-Object requested	PRVEND	REGTQA	REGTQA	QCMDSRC
CLP001	REGT	0	RPG	10	REGT +	ROGERS	10/17/2011	00-Object requested	ROGERS	REGTQA	REGTQA	QRPGSRC
CLP003	REGT	0	CLP	10	REGT0015 +	ROGERS	10/17/2011	00-Object requested	ROGERS	REGTQA	REGTQA	QCLSRC

The following fields are displayed in the ***MdObjectView.***

Folder Icon

If the Object is an IFS or Remote Folder, the Folder Icon is displayed. If contents of the folder are also requested, it can be left-clicked to expand or collapse the view of the contents.

Object

The Object name. For IFS or Server requests this value may also include the directory structure associated with the request in addition to the actual object request.

Source Icon

If the Object request contains source in the local workspace, as a source member on the IBMI, or as an IFS file on the IBMI, either the View source Icon and/or the Edit source Icon is displayed, depending on whether or not editing is allowed for that request and if the icons are included in the list of option favourites. The icon can be left-clicked to view or edit the source code.

Favourite Option Icons

All other options that are selected as favourites (click the ☰ icon at the top-right of the view to manage favourites) are displayed after the initial column, if the given option is enabled for that row.

Commit Icon

For Folders and Files that were requested from outside the IBMi, an Icon is displayed to indicate the status of the Commit of a file or for the contents of a folder. The commit process copies files from the a location such as the local workspace to the MDCMS IFS folder on the IBMi for deployment when the RFP is installed.

🟥 = File has not been committed

🟦 = File has been modified in the local workspace since the last Commit

⬛ = File is located in a workspace on another computer and can't be compared

🟨 = Committed file is the same version as file in the local workspace

Application

The Application Group name that will be used to retrieve Application Level and Environment information required for source retrieval and installation of object and/or source.

RFP

An RFP is a Request for Promotion. The RFP number that will be used for the installation of the object within the application environment. When an object or a group of objects are ready to be installed into an application environment, an RFP is required to be assigned to the object request to track and group an installation.

NOTE: An RFP is Application Group and Application Level specific. MDCMS edits the assignment process and will not allow assignments across Application Groups and Levels. When the RFP is submitted for installation, all objects that are assigned to it will be installed.

Left-click on a value in the RFP column to view/edit details of the RFP.

Level

The Application Level is used to define the target source and object libraries that changes are deployed into. If the Application Level is not specified when creating an object request, or if the object is new, the level will be automatically set to the lowest level for the specified application.

Left-click on a value in the Level column to view/edit details of the Level.

Object Type

The system object type, such as *FILE or *PGM.

Or, one of the MDCMS special object types, such as *IFS, *REMOTE or *SOURCE

MDCMS Attribute

The MDCMS Attribute is the Source or Object class code that is used during the Retrieval and Installation processes to define the compile handling or creation behaviour of the object as well as the target folders/files for the object.

Left-click on a value in the Attribute column to view/edit details of the Attribute.

Description

If the object request is for a system object or source member, then the system description of the object is displayed in this column.

Project

The Project(s) that the object request is assigned to. A Project documents the reason, the responsible parties, and timeline for work to be done. If the object is assigned to more than one(1) Project a plus sign (**+**) will be displayed to designate multiple projects.

Req Sts

The current status associated with the request record. As each object is processed through the MDCMS system, a status is applied to the object record to track the modification and installation progress.

Status values:

RP – Request pending - the RFP is waiting for Workflow acceptance of an RFP in the prior level before the new RFP can be used.
UL – Request unlocked - Object is requested in unlocked mode. Other users are able to request the same Object.
00 – Object requested - Object is requested for modification but a Request for Promotion has not been assigned.
01 – Assigned to RFP - An RFP number has been assigned to the requested object.
02 – Waiting for approval - An RFP approval is required before the installation process can continue.
CP – MDRapid Data Copy Pending
CR – MDRapid Data Copy Running
03 – Ready to be installed - The object is ready for installation and requires an RFP installation release.
04 – Installation in progress - RFP has been submitted to batch for an installation.
IC – Portion of Installation requiring lock on object in target application environment is complete. The new version of the object can now be used.

Once the RFP containing the object request is completely finished, the object request will no longer be visible from this view and is instead visible from Object History or RFP Installation History.

Create Sts

The object creation/copy status for the object in the developer library or during the compile step of the RFP.

Created Status values:

blank – an attempt to create the object has not been made
Yes – the object was successfully created or copied
Error – an error occurred when attempting to create the object
Locked – the RFP is unable to proceed due to a lock on the object
00 – 99 – the % of initial data records that have been copied by MDRapid

Res Sts

The lowest Conflict Resolution Status for the object for other versions that are based on the object.

Resolution Status values:

blank – Conflict Resolution not applicable for this request
O – Resolution Open
L – Resolution Ignored for this install level only. Will be reset to Open for next level in migration path.
I – Resolution Ignored for entire migration path
R - Resolved

User

The user that initiated the object request. When an object is requested for modification or a new object name is reserved, it is assigned to a user profile.

Cmds

true – commands are defined to be run for the specific object

Left-click on a value in the Cmds column to view/edit the list of commands defined for the specific object request.

Scripts

true – Qshell or remote platform scripts are defined to be run for the specific object

Left-click on a value in the Scripts column to view/edit the list of scripts defined for the specific object request.

From Object Library

When the object is requested for modification, the From Object Library defines the library or folder that the object or source will be migrated from.

Otherwise, the request reason will be displayed (*DELETE, *RECOMPILE or *UPDATE)

Target Library

The primary target library or target directory for the migration of the object request. This value is based on the Application level and Attribute specified for the request record.

Target Source Library

The target source library for the migration of the object request. This value is based on the Application group level specified for the request record. This value will be blank for IFS or Server Requests.

Target Sourcefile

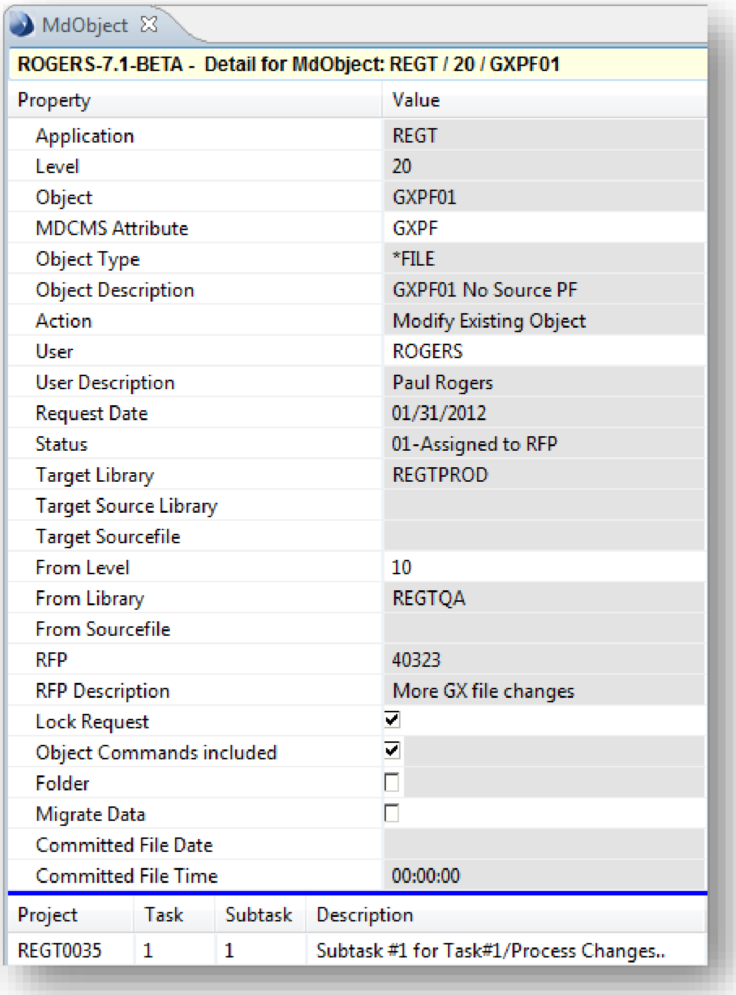
The target source file for the migration of the object request. This value is based on the Application group level specified for the request record. This value will be blank for IFS or Server Requests.

Date

The date that the Object was requested

9.3 Object View Detail

To view the complete details of any object request, left click on the object within the ***MdObjectView*** to display the ***MDObject*** view.



ROGERS-7.1-BETA - Detail for MdObject: REGT / 20 / GXPF01

Property	Value
Application	REGT
Level	20
Object	GXPF01
MDCMS Attribute	GXPF
Object Type	*FILE
Object Description	GXPF01 No Source PF
Action	Modify Existing Object
User	ROGERS
User Description	Paul Rogers
Request Date	01/31/2012
Status	01-Assigned to RFP
Target Library	REGTPROD
Target Source Library	
Target Sourcefile	
From Level	10
From Library	REGTQA
From Sourcefile	
RFP	40323
RFP Description	More GX file changes
Lock Request	☑
Object Commands included	☑
Folder	☐
Migrate Data	☐
Committed File Date	
Committed File Time	00:00:00

Project	Task	Subtask	Description
REGT0035	1	1	Subtask #1 for Task#1/Process Changes..

Some of the information in the ***MdObject*** view is already visible within the ***MdObjectView***. Refer to previous section for details on those fields.

The following are fields that appear in the ***MdObject*** view that are not present, or do not display their full details, in the ***MdObjectView***:

New Description

The description to apply to the object during the deployment. The following special values can also be used:

*SAME – if the object description of the new version of the object is blank, the description will be taken from the object it is replacing.
*BLANK – remove any description from the object

Action

The reason for the Object Request
- Modify Existing Object
- Add New Object
- Recompile Object
- Delete Object
- Update Object

User Description

The description associated with the User specified for this object request. This value is taken from the MDSEC User description.

Source Name

The name of the source member or IFS source file used to create the object

Source Attribute

The system attribute for the source member

Source Description

The text defined for the source member

From Level

If this object request was created from a lower application level this value will represent the Application level from which it originated.

From Sourcefile

The from source file for the migration of the object request. This value is based on the Application level and Attribute specified for the request record.

RFP Description

The user defined description that was assigned to the RFP associated with this object request.

Lock Request

A checkbox indicating if this object request is locked and cannot be requested by another user. A lock will only be granted to a request if the object is not already locked by another request.

Compile Sequence

The primary sort sequence for this object in an RFP based on the attribute

Compile Subsequence

The secondary sort sequence for this specific object, in case other object of the same primary sequence are dependent on it or vice-versa. This value can be modified to correct the order in which the objects will be processed by the RFP. For SQL objects see the MD Best Practice – Managing SQL Entities guide for more information on automatic sorting.

Data Origin

The file that contains the data that should be copied to the new format of this file or table at installation time.

*SAME – the data will be copied from the file of the same name as this object in the target library (default)

*MIGRATE – the contents of the new version of the file will be migrated with the format from the prior environment, replacing any records currently in the target library.

*NONE – data will not be copied to the new version of the file. If changing a file from physical to logical, *NONE must be specified.

The system name of a physical file or the SQL name of a SQL table – the name of an existing file or table in the target library that contains the records to copy to the new version of this file or table.

Data Member

The existing member(s) to migrate from the existing data origin to the new format for the file.

*FIRST – the first member will be copied
*ALL – all members will be copied – not valid when the new file is an SQL table

The name of a member in the Data Origin file

Use MDRapid

MDRapid is an add-on for MDCMS to minimize downtime with making changes to large database files.

*DEFAULT – MDRapid will be used for the physical file if the current number of records in the file are >= the minimum defined in the MDRapid template assigned to the attribute for the file.

*YES – MDRapid will be used, regardless of the number of current records.

*NO – MDRapid will not be used, regardless of the number of current records.

Reapply Journaling

A checkbox indicating if any journaling information for a file should be automatically applied to the new version of the file.

Reapply Constraints

A checkbox indicating if any constraints for a file should be automatically applied to the new version of the file.

Reapply System (non sql) Triggers

A checkbox indicating if any system (non sql) triggers for a file should be automatically applied to the new version of the file.

Reapply Logical File Members

A checkbox indicating if any current members for a logical file should be automatically applied to the new version of the file.

Committed File Date

The last modified date of the file at the time it was committed to the repository.

Committed File Time
The last modified time of the file at the time it was committed to the repository.

Revision Number
If the object was committed from SVN, this number indicates the SVN revision number of the file committed to MDCMS.

Project/Task/Subtask/Description

Basic Project information will appear in the ***MdObjectView***. When an object request is assigned to multiple Projects, Tasks, or Subtasks a plus sign (+) will appear to the right of the Project value in the ***MdObjectView***. Detailed information for these multiple Project entries can be found in the Project/Task/Subtask area of the ***MdObject*** view.
The Project/Task/Subtask section of the ***MdObject*** view allows for the addition and deletion of Project, Task and Subtask values if the user has sufficient authority for that function. Right click within the ***MdObject*** view and select ⊞ **Add to Project** or ✖ **Remove Project**.

Some important points about the Object Request Process:

The From Library, From Source File may only be changed if the object request is for the lowest level of the Application Group.

Any entered User value (programmer ID) for an object request must already exist in the **MDSEC** authorization list. If you are unable to modify the programmer ID, it is because you have not been granted authority to do so in **MDSEC**.

The Object type may not be changed as a result of changing the MDCMS Attribute for an object request.

9.4 Object View Options

The following options are available within the ***MdObjectView*** for objects:

View/Edit

View and edit the details of the object request, such as the assignee, attribute, description, and file handling parameters.

New Object Request

The New Object Request option is used to initiate a request for an object. See details for requesting native IBMi, IFS and server objects in the additional sections on this chapter.

NOTE: When no objects exist within the ***MdObjectView***, or when a right click option is initiated within the view but not against any existing object, the New Object Request option will be the only available option.

Copy

The Copy option is used to create a new Object Request with the values of the existing Request filled into the input fields.

Delete Request

The Delete Request function is used to delete an existing object request. This option is only allowed if the Status of the request record is currently *No Request assigned* (00) or *Request assigned* (01), which means that the installation of the request is not yet in progress. A confirmation window is displayed before the request(s) are actually deleted. For each request that is checked out to the developer's library, the row can be selected in the confirmation window to delete the source and/or object in the developer library, should they exist.

Multiple Object Rows may be selected at the same time using the shift or ctrl keys for this option.

Remove Request from RFP

The Remove Request function is used to remove an object request from an assigned RFP. The request still exists, but the status is returned to No Request assigned (00), allowing for the request to be assigned to a different RFP.

Multiple Object Rows may be selected at the same time using the shift or ctrl keys for this option.

Rename

The Rename function is used to rename the object or source on the object request. This is permitted when authorized to change the request and the request is for a level allowing checkout.
MDOpen can also automatically rename the object and/or source in the developer's library if the relevant checkbox in the Rename dialog is selected.

Request Related Objects

The Request Related Objects option enables the user to review and select from a list of objects that are related to the selected object. For further instructions about requesting related objects, refer to that section in this manual.

Request Linked Objects for Attribute

This option enables the user to check out objects that are linked by attribute to the selected object. For further instructions about requesting linked objects, refer to that section in this manual.

View/Edit source

The ability to view or edit source directly within the programmer's work library is enabled with this option. A user may edit the source if it is requested by that user and if the request is for a level flagged as a Check-out level for the application.

You can also left-click on the source icon in the Object Request row to open the source.

Choosing the Editor: the source editor available to the user for a particular type of source will be dependent upon the File Associations defined within Eclipse (except for LPEX-controlled source discussed below).
To view/set Preferences:
Window -> Preferences -> General -> Editors -> File Associations

For native i source: if MDOpen is used within Rational Developer for i, the associated LPEX editor will be used and control is handled natively by the RSE features that are invoked directly from the MDOpen perspective. In this case, it is important that the same user, password and host names are used in RSE that are defined for the Repository location in MDOpen.
If MDOpen isn't used within Rational Developer for i, the File Type is the system attribute for the source. For example, RPG source would have a File Type of *.rpg in the File Association listing. If an association is not present, the default text editor will be used.

LPEX Editors, in conjunction with Rational Developer for i, are available for most native languages.

Editors, in conjunction with a variety of eclipse plugins are available for most commonly used open source languages.

For SQL Scripts: if MDOpen is used within Rational Developer for i, SQL scripts will be edited by an LPEX editor, unless the MDOpen preference "Use File Associations Preferences for SQL editing" is selected.

View Committed File

If a version of the requested *IFS or *REMOTE file has been committed to MDCMS, that version can be viewed within the associated editor.

Compare workspace file with committed

With this option, the version of the currently committed code for a file is displayed next to the local workspace code and the differences between the two versions are indicated.

Compare

This option allows the requested code to be compared to another request, an archived version of the source, the committed version of the source, any source member, any IFS file, or any file in the local workspace. Optionally, a third source can be selected for a 3-way comparison. For further instructions about the Compare feature, refer to that section in this manual.

Compare history

Select one of the last n compares as a template for a new compare. The number of compares to keep in history can be set in the MDOpen preferences.

Generate REST Service

The MDRest4i SDK generator command MDRGENPRD can be invoked directly for an object request to generate the RPG source code stub for a REST API service.
The following prerequisites must be in place for this to be enabled and functioning:

- The object request is for modification of a *PGM attribute
- The source is stored in a source member
- The level is a check-out level and you have authority to the request
- The MDRGENPRD command exists in either MDRST(instance of MDOpen connection) or MDRST
- A valid SDK license key exists in the instance of MDRest4i

Generate REST Client

The MDRest4i SDK generator command MDRGENCNS can be invoked directly for an object request to generate the RPG source code stub for a REST API consumer.
The following prerequisites must be in place for this to be enabled and functioning:

- The object request is for modification of a *PGM attribute
- The source is stored in a source member
- The level is a check-out level and you have authority to the request
- The MDRGENCNS command exists in either MDRST(instance of MDOpen connection) or MDRST
- A valid SDK license key exists in the instance of MDRest4i

Save and Release Lock

If an editor that is external to eclipse, such as Microsoft Word, is used to edit source located on the IBM i, a mechanism to track when the local copy of the source should be saved and the lock of the member on the system should be released doesn't exist. For this reason, this option should be used when changes to the source are complete in order to save the source back to the system and release the lock.

Release Lock

This option is as described above, except that the local changes are not saved back to the system – only the lock on the member is released.

Create Object

If the Object Request belongs to the current user and is for a level allowing Check-Outs, a Modify, Recompile or Update Request can be compiled into the user's library for unit testing, etc. The pre-compile and compile commands defined for the object's attribute or object itself are used for creating the object so that the user can be certain that it is created exactly like it would be created at deployment time. At compile time, the library list of the target level is set, optionally with the user's own library at the top of the library list, so that references can be found for the compile.

The option to see the create log, which is a combination of job log messages generated during the create process, and compile messages, if option *EVENTF is included for the compile command, can be entered.

If the create log is displayed in Rational Developer for i, and the LPEX editor for the source is currently open, the compile message can be clicked to position the cursor in the editor to the row causing the message.

Multiple Object Rows may be selected at the same time using the shift or ctrl keys for this option.
You can also create all objects in an RFP by right-clicking on the RFP in the RFP view and selecting option Create Objects

Compile log

Display the log from the most recent creation attempt. More details about the Compile log in the Create Object option section above.

View Spooled File

If an attempt has been made to compile the source locally in the developer's library or during the compile step of the RFP, this option can be used to directly view the spooled file generated by the compiler or SQL runtime engine.

Commit File to MDCMS

Upload the newest version of the requested *IFS or *REMOTE file in the local workspace, Git or SVN to the MDCMS repository on the system. When the MDCMS RFP containing this request is executed, the committed version will be deployed to the target IFS folder, remote server or converted to the target source member.

MDXREF

Open the MDXREF Object view automatically filtered by the object's name, application and level.

Object History

Open the Object History view with the rows filtered to the specific object.

Version Conflict Resolution

This option enables the user to view/manage Resolution of conflicts for other versions of the object that are based on this request.
For further instructions about Version Conflict Resolution, refer to that section in this manual.

Bound ILE Objects

This option enables the user to view and manage the ILE Modules and Service Programs that are bound to this ILE program/service program.
For further instructions about Bound ILE Objects, refer to that section in this manual.

Object commands

The ability to view, add and edit commands related to the specific object is enabled with this option. The management and fields for Object Commands are the same as those for Attribute commands.

Object scripts

The ability to view, add and edit scripts related to the specific *IFS or *REMOTE object is enabled with this option. The management and fields for Object Scripts are the same as those for Attribute scripts.

Data Transformation

If the MDTransform add-on for MDCMS is licensed on the connected partition, you can use this option to define how existing data in each column in a modified table should be migrated to the new version of the table.

For further instructions about Data Transformation, refer to that section in this manual.

Assign to RFP

The Assign to RFP function is used to assign an object request to an RFP. The RFP must be for the same Application and Level as the request and the current status of the RFP must be *No Request assigned* (00) or *Request assigned* (01).

A window is then displayed where an existing RFP number can be entered or selected using content assist. Or, the New RFP button can be pressed to create a new RFP number.

Multiple Object Rows may be selected at the same time using the shift or ctrl keys for this option.

Assign to Project

The Assign to Project function is used to assign an object request to an active Project, Task or Subtask.

A window is then displayed where an existing Project ID can be entered or selected using content assist. To optionally specify a Task within the Project, enter or select a Task number after having entered a valid Project ID. To optionally specify a Subtask within the Task, enter or select a Subtask number after having entered a valid Task Number.

Multiple Object Rows may be selected at the same time using the shift or ctrl keys for this option.

Open RFP

If the Object Request is assigned to an RFP, this option will open the detail view for the RFP in order to view/edit the details of the RFP or to submit the RFP for processing.

RFP Listing

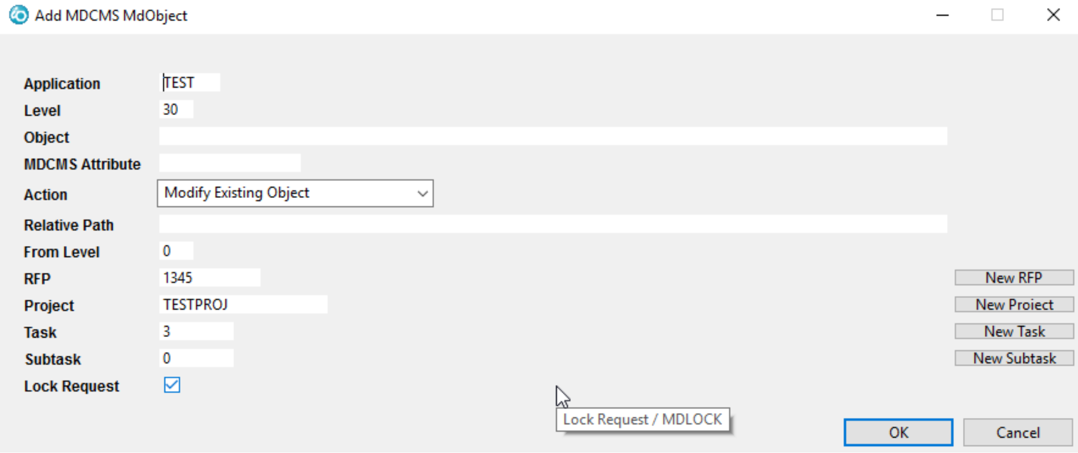
This option opens or navigates to the RFP Listing view

Open RFP and Submit

If the Object Request is assigned to an RFP in status 01-Assigned to RFP, this option will open the detail view for the RFP and begin the pre-validation processing for the submission of the RFP for deployment.

10 Request Objects directly on IBMi

To initiate an object request directly from the Object, RFP, Project, Task or Subtask listings, right click on an Object Row, or its parent row, and select ⊞ **New Object Request**. That action will display the ***Add MdObject*** dialog.



Add MDCMS MdObject

Application: TEST
Level: 30
Object:
MDCMS Attribute:
Action: Modify Existing Object
Relative Path:
From Level: 0
RFP: 1345
Project: TESTPROJ
Task: 3
Subtask: 0
Lock Request: ☑

In the ***Add MdObject*** dialog, the following Content-Assist enabled parameters are presented:

Application	The 4 character application code
Level	The application level (environment) – if left blank, the lowest level will be used
Object	the object name - enter a portion of the name for Content-Assist to list objects starting with that text
MDCMS Attribute	the attribute indicating the type and location of the object to be requested. If left blank, MDCMS will attempt to discern the attribute based on MDXREF and MDCMS history
Action	DDL to decide to Modify, Add new, Recompile, Update or Delete the object
Relative Path	The relative path, if requesting an IFS object, that resides in a subfolder of the folder defined for the attribute. Content-Assist may be used multiple times to drill down to the appropriate folder
From Level	If multiple levels migrate into the request level, the level to be migrated from can be entered here
RFP	The RFP number to use for deploying the object – use content assist to select from list or press the New RFP button to create a new RFP. The RFP field can also be left blank and the object assigned at a later time
Project	the Project, or reason, for the request - enter a portion of the name for Content-Assist to list Projects starting with that text. Or, press the New Project button to create a new Project. The Project field can also be left blank and the object assigned at a later time.
Task	a Task number within the Project, if necessary. Press the New Task button to create a new Task.
Subtask	a Subtask within the Project Task, if necessary. Press the New Subtask button to create a new subtask.
Lock Request	if checked, the request is locked for you if unchecked, another programmer can also request the object

10.1 Modify Existing Object

When an object is requested, MDCMS checks to see if another user already has the object requested. If another request of the object already exists the user will receive an *'Error'* dialog with the message *'Object is already requested by user USERNAME'*. When the user clicks OK they will be returned to the ***Add MdObject*** view with the Object field highlighted in red. The user has the option of completing the object request in unlocked mode by unchecking the *'Lock Request'* value.

MDCMS then checks if the source/object already exists in the target library of the lowest level of the Application for the entered Attribute. If not, the request process searches the library list for the Attribute to see if the object is in one of the libraries defined in the list.

The ***Add MDCMS MdObject_M*** dialog is then displayed where the user may alter where to copy the source from and where to copy the source to.

If the Source for the selected MDCMS Attribute resides in a Source Member, the following parameters are presented:

Lock Request	The source/object is reserved exclusively for this request. Others can still request the item in unlocked mode, but they won't be able to deploy it.
Compile Subsequence	Objects are sorted in the RFP (Installation package) based on the sequence of the MDCMS Attribute. If objects of the same primary sequence are dependent on each other, such as SQL Views, then this field can be used to ensure that the sort sequence of the compiles is correct. See the MD Best Practice – Managing SQL Entities guide regarding smart sorting.
Developer Object Library	The name of the Library where Development of the Object will occur. This is the location that will be used if the programmer compiles the checked out source for compile or unit testing prior to installing the RFP. Or, if the object doesn't contain source. The library can't be a library that is managed by MDCMS as a target environment library.
Copy from Source Library	The Library to copy existing source from. By default, this will be the location where MDCMS found the source
Copy from Source File	The Source File to copy existing source from. By default, this will be the location where MDCMS found the source
Copy from Source Member	The name of the Source Member to copy. By default, this will be the name of the target source
Location	When blank, the system location of the source is the same as the system location of the developer library. Otherwise, another location can be selected using Content-Assist and MDCMS will copy the source from that system to the developer library on the local system
Developer Source Library	The name of the Library where changes to the Source will occur. The library can't be a library that is managed by MDCMS as a target environment library.
Developer Source File	The name of the Source File to contain the source member to change.
Source Member Name	The name of the Source member. By default, it is the same as the object name, but can be a different name when necessary and MDCMS will then remember the name for future check-outs of that object.

If the Source for the selected MDCMS Attribute resides in IFS, the following parameters are presented:

Lock Request	The source/object is reserved exclusively for this request. Others can still request the item in unlocked mode, but they won't be able to deploy it.
Compile Subsequence	Objects are sorted in the RFP (Installation package) based on the sequence of the MDCMS Attribute. If objects of the same primary sequence are dependent on each other, such as SQL Views, then this field can be used to ensure that the sort sequence of the compiles is correct. See the MD Best Practice – Managing SQL Entities guide regarding smart sorting.
Developer Object Library	The name of the Library where Development of the Object will occur. This is the location that will be used if the programmer compiles the checked out source for compile or unit testing prior to installing the RFP. Or, if the object doesn't contain source. The library can't be a library that is managed by MDCMS as a target environment library.
Copy from IFS Source Path	The Folder to copy existing source from. By default, this will be the location where MDCMS found the source
Copy from IFS Source File	The name of the IFS File to copy. By default, this will be the name of the target source.
Developer IFS Source Path	The full path of the IFS Folder where changes to the Source will occur. The folder can't be a folder that is managed by MDCMS as a target environment folder.
Developer IFS Source Name	The name of the Source. By default, it is based on the naming defined for the selected MDCMS Attribute, but can be a different name when necessary and MDCMS will then remember the name for future check-outs of that object.

The following buttons are available:

Save	Complete the Object Request and copy the source into the developer library/folder. If the source already exists in the developer library/folder, you will be provided the choice of whether or not to overwrite the existing source
Request without Copy	Complete the Object Request, but don't copy any source to the developer library/folder
Cancel	Cancel the Object Request – MDOpen returns to the Add MDCMS MdObject window where changes can be made to the request parameters or the process can be cancelled

If clicking Save to copy the source, MDCMS checks if the source being copied differs from the source in Production (if the Compare Source level definition is set in the Location settings). If they are different, a dialog is presented providing the opportunity to compare the source versions and change or cancel the checkout.

MDCMS then checks if the last checkout for the object was from an Emergency level. If so, a dialog is presented providing the opportunity to change or cancel the checkout.

10.2 Add New Object

This option is used to reserve an object name for the purpose of creating a new object. The new object name may not already exist in the destination library for the Attribute.

The Copy parameters are the same as for a Modify

The following buttons are available:

Save	Complete the Object Request and copy the source into the developer library/folder. If the source already exists in the developer library/folder, you will be provided the choice of whether or not to overwrite the existing source.
Generate Empty Source	Complete the Object Request without copying existing source to use as a template. Instead, create an empty source member of IFS source file to code from scratch.
Request without Copy	Complete the Object Request, but don't copy any source to the developer library/folder
Cancel	Cancel the Object Request – MDOpen returns to the Add MDCMS MdObject window where changes can be made to the request parameters or the process can be cancelled

If the Save or Generate Empty Source buttons are pressed, and the source is stored in a Source Member, you are then prompted for the Attribute and Description of the new Source Member.

10.3 Recompile Object

This option is used to recompile an object at the requested level. This type of request can be initiated when work is in progress for the object and source. The user specifies a value of **Recompile Object** for the Action parameter of the ***Add MdObject*** dialog and the object will be added to the ***MdObjectView*** with a value of '**RECOMPILE*' in the '*From Library*' column.

When the source for the object is not found in the migration chain or search template based on the selected MDCMS attribute, the user will receive an '*Error*' dialog with the message '*MDCMS could not find the Source for this object.*'

10.4 Delete Object

This option is used to delete an object and the source for the object. When the promotion occurs to delete the object the source or object is archived prior to deletion. Any cross-reference information about the object is also removed.

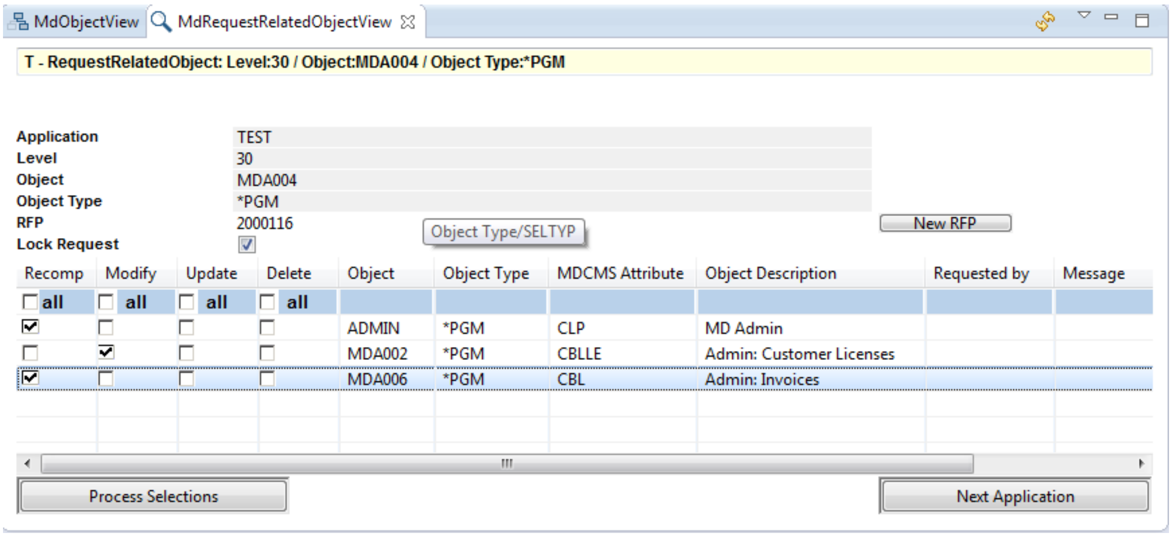
10.5 Update Object

This option is used to update an existing object without modifying any source or migrating the object from a lower level. An Update command must be defined for the object or Attribute. An example of an Update command would be the UPDPGM command for updating existing ILE programs. The Update option may be requested even when work is in progress for the source or object.

10.6 Request Related Objects

The Request Related Objects function enables the user to review and request objects that are related to the selected object. This function is used to request objects that are dependent upon a requested file, ILE module, Service Program, copybook, or that invoke a requested program, command, query, menu, etc... The **MDXREF** function of MDCMS is used to identify these objects.

To Request Related Objects, right click on an existing object request in Object Listing and select 🔍 **Request Related Objects**. That action will bring up the ***MdRequestRelatedObjectView***.



T - RequestRelatedObject: Level:30 / Object:MDA004 / Object Type:*PGM

Application	TEST
Level	30
Object	MDA004
Object Type	*PGM
RFP	2000116
Lock Request	☑

Recomp	Modify	Update	Delete	Object	Object Type	MDCMS Attribute	Object Description	Requested by	Message
☐ all	☐ all	☐ all	☐ all						
☑	☐	☐	☐	ADMIN	*PGM	CLP	MD Admin		
☐	☑	☐	☐	MDA002	*PGM	CBLLE	Admin: Customer Licenses		
☑	☐	☐	☐	MDA006	*PGM	CBL	Admin: Invoices		

Header Parameters:

Application	The Application Code of the Related Objects. The initial listing will be for the same Application as the selected Object. When the Next Application button is clicked, and references are found in another linked application, the application code will change and its dependencies will be listed.
Level	The Level of the selected Object
Object	The selected Object name
Object Type	The System or MDCMS Type of the selected Object
RFP	The RFP number to assign to the objects that are selected in the list. This field is editable. Options: - Blank out the field to Request objects without already assigning them to an RFP. - Enter an existing RFP number that is in status 00 or 01 for the given Application and Level, or use Content-Assist to select a valid RFP from a list. - Press the New RFP button to create a new RFP number
Lock Request	A checkbox indicating if these object requests will be locked and cannot be requested by another user. A lock will only be granted to a request if the object is not already locked by another request.

Listing Parameters:

Recomp	Check the box if the Object should be requested for Recompile. The all checkbox can be used to select or unselect all rows that are qualified to be recompiled based on the MDCMS Attribute.
Modify	Check the box if the Object should be requested for Modify. The all checkbox can be used to select or unselect all rows that are qualified to be modified based on the MDCMS Attribute.
Update	Check the box if the Object should be requested for Update. The all checkbox can be used to select or unselect all rows that are qualified to be updated based on the MDCMS Attribute.
Delete	Check the box if the Object should be requested for Delete. The all checkbox can be used to select or unselect all rows that are qualified to be deleted based on the MDCMS Attribute.
Object	Object name
Object Type	The System or MDCMS Type of the Object
MDCMS Attribute	The Attribute that defines how an object is to be created and the target source and object libraries for the object and level. MDCMS predetermines the attribute based on Installation History for the object and cross-referencing information. This value can be modified and content assist can be used to select a different attribute from a list.
Object Description	The system description for the Object
Requested by	The user that has already requested this object in lock mode, or, the RFP request reason when the object is already requested and assigned to the RFP number selected in the header.
Message	The error or success message based on results of attempting to process the selection.

Buttons:

Process Selections	Once one or more Object Rows have been selected, press this button to generate the requests. If a row is selected for Modify, the Modify Object window will be displayed to confirm the From and To locations for the source.
Next Application	If Linked Applications are defined for the Application of the original object, the Next Application button can be clicked to check if references exist in another linked application. If found, those dependencies will be listed for selection and the Next Application can be pressed again to continue looping through the Linked Applications. If no further dependencies are found, the process will end.

Some examples where the Request Related Objects function can be used:

Example 1: the length of a field in file COHDRP needs to be changed. The user would first select COHDRP for modification. Then, the user would use the Request Related Objects option for the file so that all of the objects that use COHDRP (or other files that use COHDRP, such as logical files) can be selected for modification, deletion, or recompilation.

Example 2: the *ENTRY parameters of a RPG program needs to be changed. The user would first select the program for modification. Then, the user would use the Request Related Objects option for the program so that all of the objects that invoke the program can be selected for modification, deletion, or recompilation.

10.7 Request Linked Objects for Attribute

When an object is checked out with an attribute that has other attributes linked to it, the developer is automatically presented a dialog with the names and attributes of objects that are suggested to be checked out at the same time. This is based on the Linked Attribute configuration in the settings.

This option is also available for an active request from the object list.

Header Parameters:

Application	The Application Code of the selected Object
Level	The Level of the selected Object
Object	The selected Object name
Object Type	The System or MDCMS Type of the selected Object
Attribute	The Attribute of the selected Object
Lock Request	A checkbox indicating if these object requests will be locked and cannot be requested by another user. A lock will only be granted to a request if the object is not already locked by another request.

If a request is processed from this dialog, the RFP and Project information for the selected Object will be automatically applied to the linked objects.
Listing Parameters:

Modify	Check the box if the Object should be requested for Modify. The all checkbox can be used to select or unselect all rows that are qualified to be modified based on the MDCMS Attribute.
New	Check the box if the Object is new for the application and should be requested to add to the application. The all checkbox can be used to select or unselect all rows that are qualified to be modified based on the MDCMS Attribute.
Recomp	Check the box if the Object should be requested for Recompile. The all checkbox can be used to select or unselect all rows that are qualified to be recompiled based on the MDCMS Attribute.
Update	Check the box if the Object should be requested for Update. The all checkbox can be used to select or unselect all rows that are qualified to be updated based on the MDCMS Attribute.
Delete	Check the box if the Object should be requested for Delete. The all checkbox can be used to select or unselect all rows that are qualified to be deleted based on the MDCMS Attribute.
Object	Object name, which is editable and enabled for content-assist in the dialog
Object Type	The System or MDCMS Type of the Object
MDCMS Attribute	The Attribute that defines how an object is to be created and the target source and object libraries for the object and level. The attribute is fix to the value in the linked attribute settings.
Message	The error or success message based on results of attempting to process the selection.

Buttons:

Process Selections	Once one or more Object Rows have been selected, press this button to generate the requests. If a row is selected for Modify or New, the window will be displayed to confirm the From and To locations for the source.

10.8 Bound ILE Objects

When an ILE Program or Service Program is requested, the bound modules and service programs for the request can be viewed/modified and checked out from the list. This option is available as an option for an active program or service program from the object list.

Header Parameters:

Application	The Application Code of the selected Object
Level	The Level of the selected Object
Object	The selected Object name
Object Type	The System or MDCMS Type of the selected Object
Lock Request	A checkbox indicating if these object requests will be locked and cannot be requested by another user. A lock will only be granted to a request if the object is not already locked by another request.

If a request is processed from this dialog, the RFP and Project information for the selected Object will be automatically applied to the linked objects.

Listing Parameters:

Modify	Check the box if the Object should be requested for Modify. The all checkbox can be used to select or unselect all rows that are qualified to be modified based on the MDCMS Attribute.
Remove	Check the box if the Object should be remove from the binding list for the requested program. The all checkbox can be used to select or unselect all filled rows.
Object	Object name, which is editable and enabled for content-assist in the dialog for empty rows
Object Type	The System Type of the Object. *MODULE or *SRVPGM can be selected from the drop down list for empty rows
PEP	Checkbox indicating if Module contains the Program Entry Procedure for an ILE program. When editable, a module can be selected as the PEP.
MDCMS Attribute	The Attribute that defines how an object is to be created and the target source and object libraries for the object and level. MDCMS predetermines the attribute based on Installation History for the object and cross-referencing information. This value can be modified and content assist can be used to select a different attribute from a list.
Object Description	The system description for the Object
Requested by	The user that has already requested this object in lock mode, or, the RFP request reason when the object is already requested and assigned to the RFP number selected in the header.
Message	The error or success message based on results of attempting to process the selection.

Buttons:

Process Selections	Once one or more Object Rows have been selected, press this button to generate the requests. If a row is selected for Modify or New, the window will be displayed to confirm the From and To locations for the source.

10.9 Version Conflict Resolution

This option is used to manage resolution of objects belonging to multiple versions of the same Application. Conflict Resolution is automatically prompted for an object when it is checked out for New, Modify or Delete from a level that other levels are based on. In the case of Modify or Delete, the prompt is only given when the object also exists in at least one of the dependant levels.

Since Resolution may not be possible immediately when an object is checked out, or to view/change the status of Resolution, it can be managed at any time by using option Version Conflict Resolution for an object showing a Resolution Status in the Object Listing or by using option Version Conflict Resolution for an RFP to see all relevant objects in the RFP in one view.

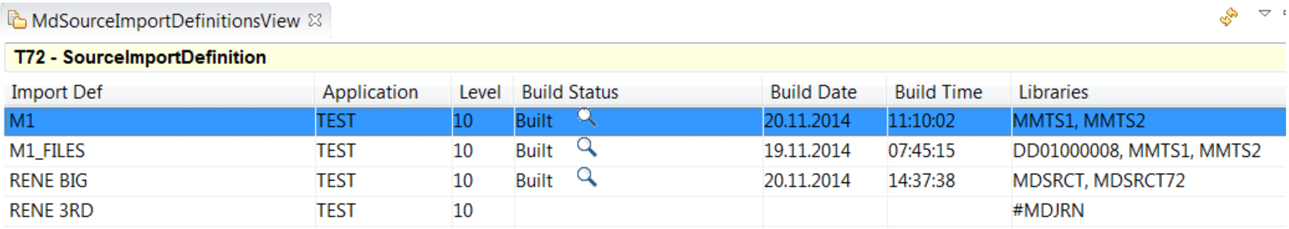
Options:

Sel	This checkbox will appear for each row at checkout time to make it simple so request the object from some or all based-on levels. The action that will be performed for a selected row will be the action listed in column Recommended. Click all to select/unselect all rows. Once satisfied with the selections, click button Process Selected Recommended Options
Bind Request	The object is already requested in the other level – resolve conflict by binding this request to the existing request
Clear Status	Remove resolution of the level
Modify	Request the Object in the other level for modification
Recompile	Request the Object in the other level for recompile
Update	Request the Object in the other level for update
Delete	Request the Object in the other level for deletion
Ignore Version	Ignore resolution for this level and all levels that this level migrates to
Ignore Version for this Level only	Ignore resolution for this level only. Higher levels that this level migrates to will require resolution.

10.10 Import Source

The Import Source function enables the user to search for source members in non-managed libraries, compare those members to the members in the application libraries and then request the members for deployment into the application.

To Work with the function, right click on a Repository Location and select **Import Source**. That action will bring up the ***MdSourceImportDefinitionsView***.



MdSourceImportDefinitionsView

T72 - SourceImportDefinition

Import Def	Application	Level	Build Status	Build Date	Build Time	Libraries
M1	TEST	10	Built	20.11.2014	11:10:02	MMTS1, MMTS2
M1_FILES	TEST	10	Built	19.11.2014	07:45:15	DD01000008, MMTS1, MMTS2
RENE BIG	TEST	10	Built	20.11.2014	14:37:38	MDSRCT, MDSRCT72
RENE 3RD	TEST	10				#MDJRN

Right-Click any row to select the option to add a new Source Import definition or any filled row to select to Copy or Delete the definition.

10.10.1 Source Import Definition

A Source Import Definition is a reusable set of search criteria for the listing of Source member candidates to import into an Application Level.

Header Parameters:

Application	The Application Code of the application to import source into
Level	The Level of the Application that allows checkout
Member Name	Limit members to those with the given name * can be used before and/or after the value for generic searching
Source File	Limit members to those residing in the entered source file. * can be used before and/or after the value for generic searching
Source Type	Limit members to those with the given type * can be used before and/or after the value for generic searching
Source Text	Limit members to those with the given text description * can be used before and/or after the value for generic searching
Minimum Create Date	Limit members to those created on or after the entered date
Only New/Different	Limit members to those that are new or different when compared to the members in the application
Omit Comments	True - Don't include blank or comment lines in the source code for the comparison. A member will only be considered different when uncommented code varies when compared to the application. False – any difference in the source member code will cause the member to be considered different.

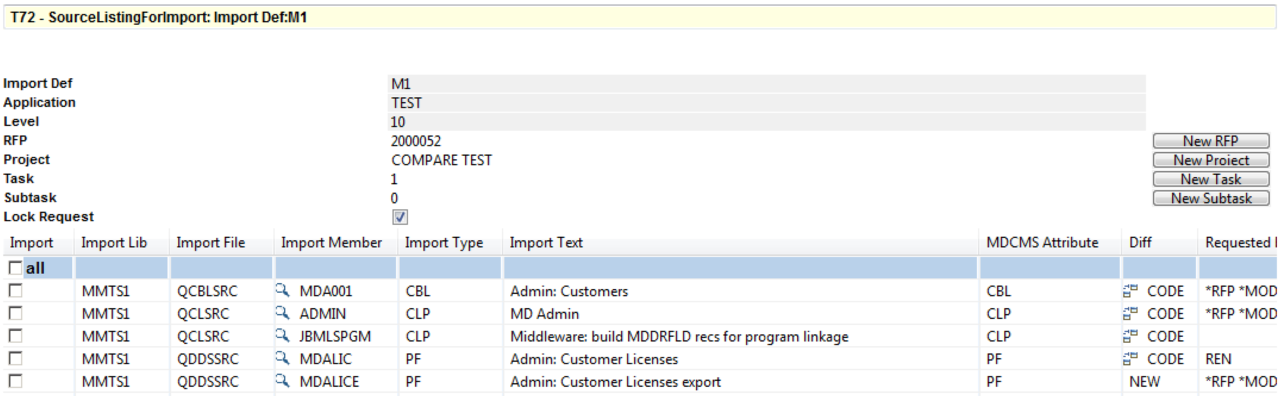
Libraries – a list of 1-n libraries to be included in the search. A library defined as a target library for an MDCMS attribute is not permitted.

Buttons:

Build List	Once at least one library is included for the definition, this button is enabled. When clicked, a job is submitted to batch to generate a list of all members matching the filter criteria. A message will be displayed when the batch job is complete.
Work with Results	Once a Build is complete, this button is enabled. When clicked, the list of found source members are listed.

10.10.2 Source Import Result List

The Result List displays all source members found in the Definition's Libraries that meet the search criteria.



T72 - SourceListingForImport: Import Def:M1

Import Def	M1
Application	TEST
Level	10
RFP	2000052
Project	COMPARE TEST
Task	1
Subtask	0
Lock Request	☑

Import	Import Lib	Import File	Import Member	Import Type	Import Text	MDCMS Attribute	Diff	Requested I
□ all								
□	MMTS1	QCBLSRC	MDA001	CBL	Admin: Customers	CBL	CODE	*RFP *MOD
□	MMTS1	QCLSRC	ADMIN	CLP	MD Admin	CLP	CODE	*RFP *MOD
□	MMTS1	QCLSRC	JBMLSPGM	CLP	Middleware: build MDDRFLD recs for program linkage	CLP	CODE	
□	MMTS1	QDDSSRC	MDALIC	PF	Admin: Customer Licenses	PF	CODE	REN
□	MMTS1	QDDSSRC	MDALICE	PF	Admin: Customer Licenses export	PF	NEW	*RFP *MOD

Header Parameters:

RFP	The RFP number to assign to any selected members to request
Project	The Project ID to assign to any selected members to request
Task	The Project Task number to assign to any selected members to request
Subtask	The Project Subtask number to assign to any selected members to request
Lock Request	Specify if members should be requested in locked or unlocked modes

Import Column
Click for any row to select to create an Object Request to migrate the source from the import library into the defined Application Level.
The all checkbox can be clicked to select all rows that aren't already requested by a user.

Diff Column
Blank – no difference between source in Import Library and source in application
CODE – differences found in source code
NEW – source member only in Import Library and not in application. For a source file to be considered for the application, an MDCMS attribute must be defined that refers to the source file as the target for the level or a level higher up in the chain on the same system.
TYPE – the source type differs between the member in the Import Library and the member in the application.

MDCMS Attribute Column
The attribute to apply to the object request when processed

The 🔍 icon next to the import member or next to the existing member can be clicked to view the source code in that member.

The icon in the Diff column can be clicked to show the differences between the import member and the existing member

Button:

Process Selections	Generate the Object Requests for all selected rows

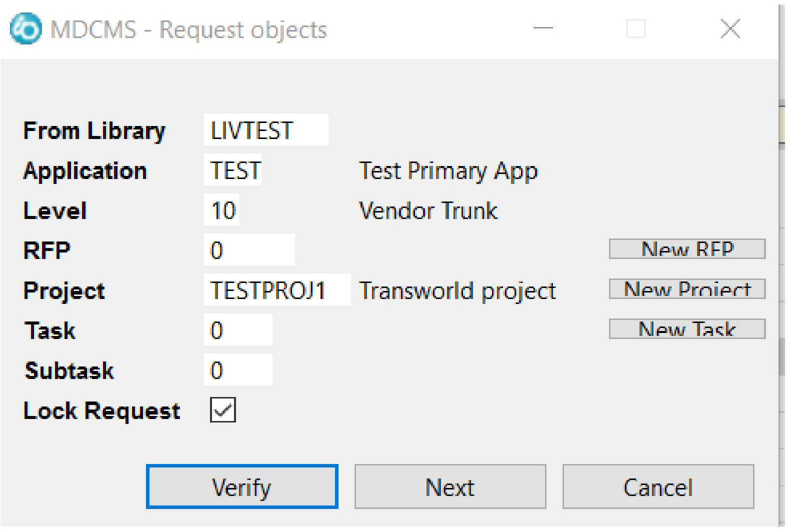
10.11 Import Library

The Import Library function enables the user to select one or more objects to be migrated from a non-managed library into a target environment.

A typical use for the Import Library function is when a vendor delivers object modifications in a temporary library.

This feature is accessed from the repository connection option Import Library .

The initial dialog



MDCMS - Request objects

From Library	LIVTEST		
Application	TEST	Test Primary App	
Level	10	Vendor Trunk	
RFP	0		New RFP
Project	TESTPROJ1	Transworld project	New Project
Task	0		New Task
Subtask	0		
Lock Request	☑		

From Library	The library containing the objects to be imported into an application
Application	The Application Code
Level	The Promotion Level. This level must allow checkouts to be allowed.
RFP	The RFP number to assign to the objects that are selected in the list- Enter an existing RFP number that is in status 00 or 01 for the given Application and Level, or use Content-Assist to select a valid RFP from a list. - Press the New RFP button to create a new RFP number
Project	An open and authorized Project to apply to any selected objects - Press the New Project button to create a new Project
Task	An open task within the selected project - Press the New Task button to create a new task for the project
Subtask	An open subtask within the selected project task - Press the New Subtask button to create a new subtask for the task
Lock Request	A checkbox indicating if the selected object requests will be locked and cannot be requested by another user. A lock will only be granted to a request if the object is not already locked by another request.

After pressing the Next button, the list of all objects in the library are listed.
Select each object in the list that should be requested for migration to the select application level. Alternatively, select **all** to select every object in the list. Afterwards, objects can be de-selected.

The MDCMS Attribute column specifies which attribute will be applied to each selected object. This value can be changed within the list, when necessary.

10.12 Request Objects from MDXREF

The Request Object option is available from any Object-Level listing within MDXREF. Mark some or all of the rows in the object list, then right-click on one of the marked rows and select Request Object to check out the object(s).

When the request option is selected, an initial dialog captures the following primary information for the request of the object(s):

Application	The Application Code
Level	The Promotion Level. This level must allow checkouts to be allowed.
RFP	The RFP number to assign to the objects that are selected in the list- Enter an existing RFP number that is in status 00 or 01 for the given Application and Level, or use Content-Assist to select a valid RFP from a list. - Press the New RFP button to create a new RFP number
Project	An open and authorized Project to apply to any selected objects - Press the New Project button to create a new Project
Task	An open task within the selected project - Press the New Task button to create a new task for the project
Subtask	An open subtask within the selected project task - Press the New Subtask button to create a new subtask for the task
Lock Request	A checkbox indicating if the selected object requests will be locked and cannot be requested by another user. A lock will only be granted to a request if the object is not already locked by another request.

Once the Next button is clicked. All selected objects are listed.

Listing Parameters:

Modify	Check the box if the Object should be requested for Modify. The all checkbox can be used to select or unselect all rows that are qualified to be modified based on the MDCMS Attribute.
Recomp	Check the box if the Object should be requested for Recompile. The all checkbox can be used to select or unselect all rows that are qualified to be recompiled based on the MDCMS Attribute.
Update	Check the box if the Object should be requested for Update. The all checkbox can be used to select or unselect all rows that are qualified to be updated based on the MDCMS Attribute.
Delete	Check the box if the Object should be requested for Delete. The all checkbox can be used to select or unselect all rows that are qualified to be deleted based on the MDCMS Attribute.
Object	Object name
Object Type	The System or MDCMS Type of the Object
MDCMS Attribute	The Attribute that defines how an object is to be created and the target source and object libraries for the object and level. MDCMS predetermines the attribute based on Installation History for the object and cross-referencing

	information. This value can be modified and content assist can be used to select a different attribute from a list.
Object Description	The system description for the Object
Requested by	The user that has already requested this object in lock mode, or, the RFP request reason when the object is already requested and assigned to the RFP number selected in the header.
Message	The error or success message based on results of attempting to process the selection.

10.13 Request Source Members from MDXREF

Highlight one or many rows in the MDXREF Source list and right-click to select the option Request Source to check out the source member(s).

When the request option is selected, an initial dialog captures the following primary information for the request of the member(s):

Application	The Application Code
Level	The Promotion Level. This level must allow checkouts to be allowed.
RFP	The RFP number to assign to the objects that are selected in the list- Enter an existing RFP number that is in status 00 or 01 for the given Application and Level, or use Content-Assist to select a valid RFP from a list. - Press the New RFP button to create a new RFP number
Project	An open and authorized Project to apply to any selected objects - Press the New Project button to create a new Project
Task	An open task within the selected project - Press the New Task button to create a new task for the project
Subtask	An open subtask within the selected project task - Press the New Subtask button to create a new subtask for the task
Lock Request	A checkbox indicating if the selected object requests will be locked and cannot be requested by another user. A lock will only be granted to a request if the object is not already locked by another request.

Once the Next button is clicked. All selected objects are listed.

Listing Parameters:

Modify	Check the box if the Object should be requested for Modify. The all checkbox can be used to select or unselect all rows that are qualified to be modified based on the MDCMS Attribute.
Recomp	Check the box if the Object should be requested for Recompile. The all checkbox can be used to select or unselect all rows that are qualified to be recompiled based on the MDCMS Attribute.
Update	Check the box if the Object should be requested for Update. The all checkbox can be used to select or unselect all rows that are qualified to be updated based on the MDCMS Attribute.
Delete	Check the box if the Object should be requested for Delete. The all checkbox can be used to select or unselect all rows that are qualified to be deleted based on the MDCMS Attribute.
Source Library	The library containing the source member
Source File	The source file containing the source member
Source Member	The name of the source member

Attribute	The system attribute of the source member
Description	The Text description of the source member
Object	The Object name to use for the request. This value is editable and can differ from the name of the source.
MDCMS Attribute	The Attribute that defines how an object is to be created and the target source and object libraries for the object and level. MDCMS predetermines the attribute based on Installation History for the object and cross-referencing information. This value can be modified and content assist can be used to select a different attribute from a list.
Requested by	The user that has already requested this object in lock mode, or, the RFP request reason when the object is already requested and assigned to the RFP number selected in the header.
Message	The error or success message based on results of attempting to process the selection.

10.14 Request IFS Objects from MDXREF

Highlight one or many rows in the MDXREF IFS list and right-click to select the option Request Object to check out the IFS folder(s) and file(s).

When the request option is selected, an initial dialog captures the following primary information for the request of the object(s):

Developer Folder	The non-managed development folder that any requested objects should be copied to
Overwrite any existing Files	If checked, then any files that already exist in developer folder that match the name of the requested files will be overwritten. Otherwise, existing files will be left alone, but the request will still be generated.
Application	The Application Code
Level	The Promotion Level. This level must allow checkouts to be allowed.
MDCMS Attribute	The *IFS or *REMOTE attribute to apply to the requested folders and files to indicate the target server/root folder.
Action	Migrate – check out the objects with the intent of migrating them to the target environment when the RFP is installed Delete – create object deletion requests. The objects will not be copied to the developer folder.
RFP	The RFP number to assign to the objects that are selected in the list- Enter an existing RFP number that is in status 00 or 01 for the given Application and Level, or use Content-Assist to select a valid RFP from a list. - Press the New RFP button to create a new RFP number
Project	An open and authorized Project to apply to any selected objects - Press the New Project button to create a new Project
Task	An open task within the selected project - Press the New Task button to create a new task for the project
Subtask	An open subtask within the selected project task - Press the New Subtask button to create a new subtask for the task
Lock Request	A checkbox indicating if the selected object requests will be locked and cannot be requested by another user. A lock will only be granted to a request if the object is not already locked by another request.

Target Relative Path	By default, the requested folder structure will be deployed directly to the target folder defined by the attribute. If the folder structure should be deployed to a subfolder for the target environment, enter the relative path of that folder from the point where the attribute folder leaves off. Content assist can be used to browse the currently defined subfolders on the target file system or a new path can be entered.

Once the Next button is clicked. All selected objects are listed and pre-selected. The All, None, Files and Directories options can be clicked to mass-change the selected items, or individual rows can be selected/unselected.

Once the selections are chosen, click Request to process the selections.

10.15 Request IFS Source from MDXREF

Highlight one or many rows in the MDXREF IFS list and right-click to select the option Request Source to check out the IFS file(s) to use as source for system objects.

When the request option is selected, an initial dialog captures the following primary information for the request of the source:

Developer Folder	The non-managed development folder that any requested files should be copied to
Overwrite any existing Files	If checked, then any files that already exist in developer folder that match the name of the requested files will be overwritten. Otherwise, existing files will be left alone, but the request will still be generated.
Application	The Application Code
Level	The Promotion Level. This level must allow checkouts to be allowed.
Action	Migrate – check out the objects with the intent of migrating them to the target environment when the RFP is installed Delete – create object deletion requests. The objects will not be copied to the developer folder.
RFP	The RFP number to assign to the objects that are selected in the list- Enter an existing RFP number that is in status 00 or 01 for the given Application and Level, or use Content-Assist to select a valid RFP from a list. - Press the New RFP button to create a new RFP number
Project	An open and authorized Project to apply to any selected objects - Press the New Project button to create a new Project
Task	An open task within the selected project - Press the New Task button to create a new task for the project
Subtask	An open subtask within the selected project task - Press the New Subtask button to create a new subtask for the task
Lock Request	A checkbox indicating if the selected object requests will be locked and cannot be requested by another user. A lock will only be granted to a request if the object is not already locked by another request.

Once the initial information is correct, click Next to continue to the selection listing.

Listing Parameters:

Selection	If checked, the file will be included in the checkout
File	The name of the IFS file to be used as source
Request Path	The IFS location where the file will be copied from

Action	the selected action from the initial screen – Migrate or Delete
MDCMS Attribute	The Attribute that defines how an object is to be created and the target source and object libraries for the object and level. MDCMS predetermines the attribute based on Installation History for the object and cross-referencing information. This value can be modified and content assist can be used to select a different attribute from a list. When selecting to check out IFS source, the attribute must be defined with a target source file of *IFS
Object Name	The system or SQL name of the Object that will be created from the requested source. This value can be modified and content assist can be used to select a different object from a list.
Member Name	not applicable when requesting IFS source from MDXREF
Message	The error or success message based on results of attempting to process the selection.

10.16 Import Objects or Source from IFS Folder

MDOpen can import the folders and files in an IFS folder directly into an RFP or Project for deployment to target IFS folders or to target Remote Servers. Or, the files can be imported to be used as source for system or SQL objects.

Option Import IFS Folder can be selected from the following MDOpen locations:

- Repository list under the Import section – fields will be filled with any valid values from the most recent import
- MDXREF IFS list – fields will be filled valid values from most recent import
- RFP list for any RFP in status 00 or 01 – Application and RFP number field will be pre-filled with selected RFP values
- Project list for any open Project – Project will be pre-filled
- Task list for any open Task – Project and Task will be pre-filled
- Subtask list for any open Subtask – Project, Task and Subtask will be pre-filled

The initial dialog includes the following checkboxes:

- Request IFS/Remote Objects – select this to handle the selected folders/files as objects. See section Request Objects from MDXREF for further information.
- Request Source for IBMi Objects – select this to handle the selected files as source for IBMi objects. See section Request Source from MDXREF for further information.

11 Request Objects from Other Locations

Requesting and committing folders and files from outside the IBMi can be done in the following ways:

- From a user's workspace
- From a server (any server other than the IBM i)
- From Local Folders
- From Git (Git Repository)
- From SVN (Subversion Repository)

The deployment targets of these requests can be:

- Remote Linux or Windows Servers
- IFS folders for IFS object files
- IFS folders for IFS source used to create IBMi objects
- Source members used to create IBMi objects (MDCMS automatically converts them from stream files to members if the target attribute requires the source to be in a source file)

NOTE 1: The context menu options for requesting objects from various views and perspectives in Eclipse are always duplicated.

The 1st listing of an option will include the text ***(default repository)***. Selecting this option will indicate to MDOpen to use the MDCMS Repository location defined as the default. To set a default repository location, check the box labelled ***Default repository*** for the appropriate Repository connection that you have created.

The 2nd listing of an option won't include the text ***(default repository)***. If this option is selected, MDOpen will provide a dialog with the list of defined repositories to select from.

NOTE 2: Anything requested from outside the IBMi must be committed to MDCMS before an RFP can deploy the files to the target file system. A commit copies the local or team repository copy of files to the IBM i in preparation for RFP installation.

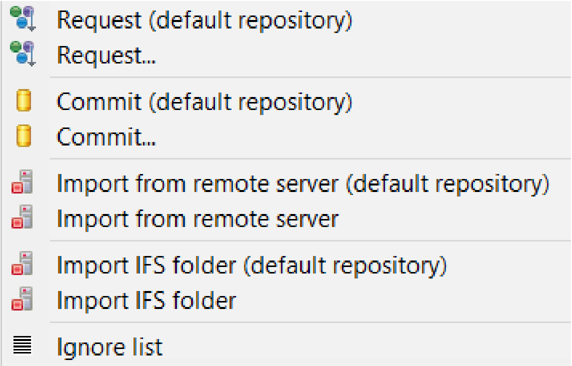
The Commit option is available anywhere the request option is available as well as directly from the Objects listing in the MDOpen perspective.

If changes occur to the local copy of a file since the last commit, and those changes should also be included in the deployment, then reselect the commit option for the file.

NOTE 3: Sections 11.1 – 11.5 explain the different areas within Eclipse where folders and files can be requested. Sections 11.6 - 11.8 explain the central dialogs to process the selections.

11.1 Request Objects to/from Workspace

MDOpen provides the following context menu options within any Eclipse perspective when right-clicking on a folder or file within a local workspace navigator view:


Request (default repository)
Request...
Commit (default repository)
Commit...
Import from remote server (default repository)
Import from remote server
Import IFS folder (default repository)
Import IFS folder
Ignore list

Request – the folders and files already exist in the local workspace and need to be requested for deployment by MDCMS. Right-click on the parent folder or one or more files in a folder in a workspace navigator view to select them.

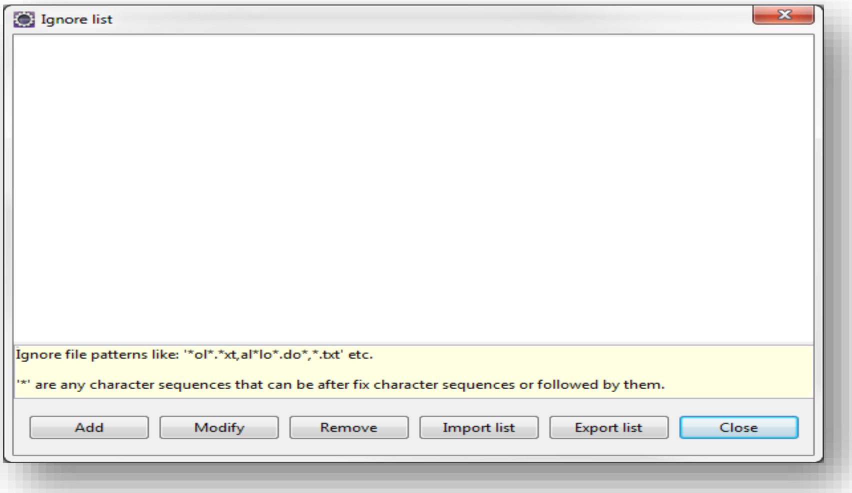
Commit – the folders and files already exist in the local workspace and are already requested for MDCMS. Any changes that have been made to the files since the request can then be repeatedly committed to the MDCMS repository. The commit can also occur directly from the request record.

Import from Remote Server – the current version of the folders and files exist on a remote server and need to be copied into the local workspace for modification, or merely to have in the local workspace. Right-click on the folder in a workspace navigator view to indicate the destination that the soon-to-be-selected folders and files will be copied to.

Import from IFS – the current version of the folders and files exist in IFS (folder structure on the IBM i) and need to be copied into the local workspace for modification, or merely to have in the local workspace. Right-click on the folder in a workspace navigator view to indicate the destination that the soon-to-be-selected folders and files will be copied to.

11.1.1 Create an Ignore List for Requests and Commits from Local Workspace

Right click and select ***MDCMS => Ignore list*** and the following dialog is presented:



Ignore file patterns like: '*ol*.*xt,al*lo*.do*,*.txt' etc.

'*' are any character sequences that can be after fix character sequences or followed by them.

The Ignore list dialog provides a function for the user to specify a list of file patterns that are to be ignored during requests.

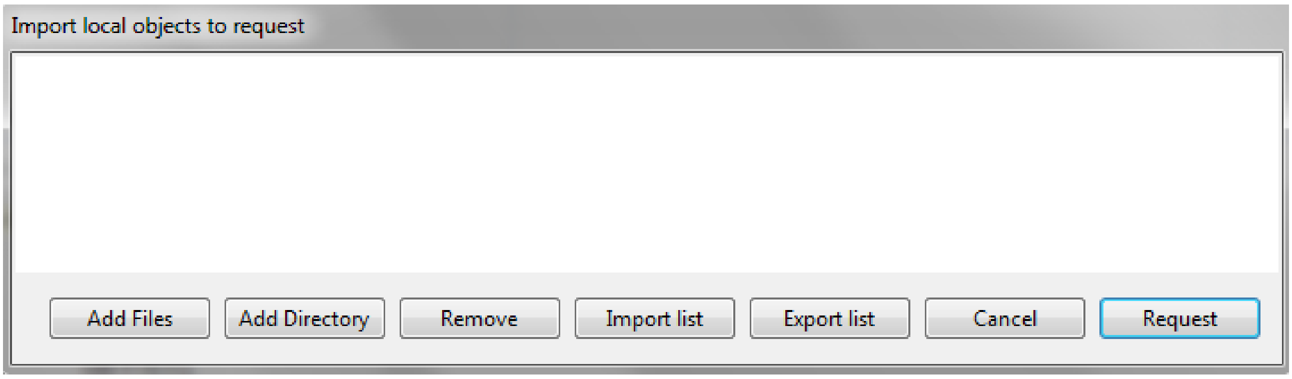
Ignore file pattern examples may be: *ol*, *xt, al*lo*.do* or *.txt where '*' is any character sequence that can be after a fixed character sequence or followed by them.

11.2 Import Objects from Local Folders

MDOpen can import any files and folders from the local file system by selecting Repository option:

Import Local Objects

This brings up the following window:



Files and Folders can be dragged from a file explorer such as Windows Explorer and dropped into this window. Alternatively, the following buttons can be used:

Add Files – browse the file system for files to individually add to the list in the window.
Add Directory – browse the file system for folders to add to the list in the window. Once the request button is pressed, all subfolders and files will be listed for possible selection/de-selection.
Remove – remove some of the files or folders from the list.
Export list – export the currently displayed list of files and folders to a file for simple reuse in future requests.
Import list – import a previously export list of files and folders into the window.

Once the list is complete, click the Request button.

11.3 Import Objects from Remote Server

MDOpen can import the folders and files on a remote system directly into an RFP or Project for deployment to target IFS folders or to target Remote Servers. Or, the files can be imported to be used as source for system or SQL objects.

Option Import from Remote Server can be selected from the following MDOpen locations:

- Repository list under the Import section – fields will be filled with any valid values from the most recent import
- RFP list for any RFP in status 00 or 01 – Application and RFP number field will be pre-filled with selected RFP values
- Project list for any open Project – Project will be pre-filled
- Task list for any open Task – Project and Task will be pre-filled
- Subtask list for any open Subtask – Project, Task and Subtask will be pre-filled

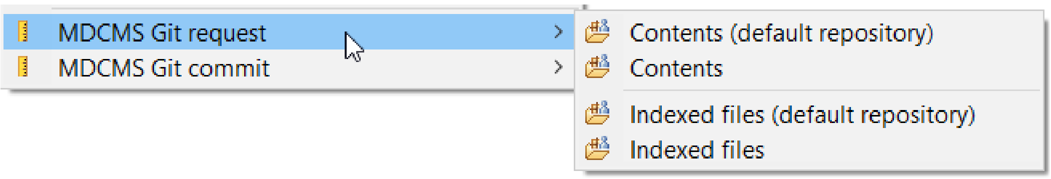
The Remote Server that is selected in the initial dialog must already be defined in the Remote Server Locations settings. MDOpen then uses the defined FTP properties to get any selected folders and files.

11.4 Request Objects from the Git Perspective

Prerequisites:

- The Eclipse Git Plugin (EGit) must be installed in the IDE with a minimum version of 3.2.
- Each local Git repository to pull objects from must be defined in the Git Repositories view.

From the Git Repositories view, expand the repository and right-click on any file or folder within the Working Tree or right-click on any tag in the Tags list. The following options are displayed:


MDCMS Git request
MDCMS Git commit
Contents (default repository)
Contents
Indexed files (default repository)
Indexed files

Request Contents – request some or all of the files/folders that are contained in the selection.

Request Indexed files – request some or all of the files/folders that are contained in the selection and have staged changes. Not applicable from the Tags list.

Commit Contents – re-commit any files that have changed since they were first requested.

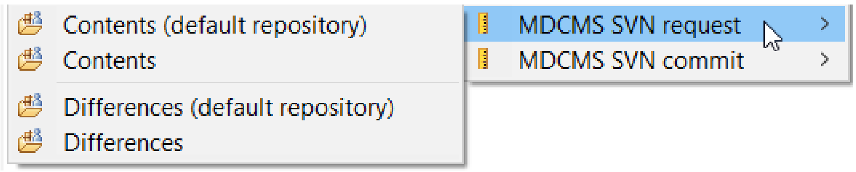
Commit Indexed files – re-commit any staged files that have changed since they were first requested. Not applicable from the Tags list.

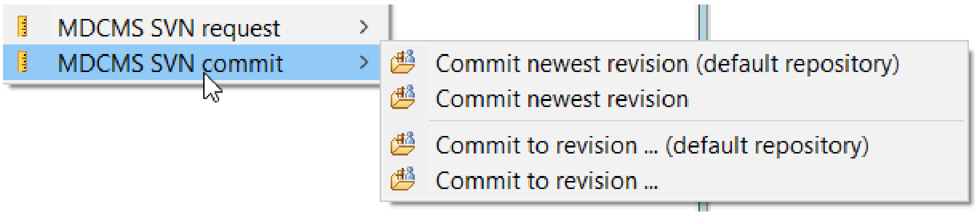
11.5 Request Objects from the SVN Repository Exploring Perspective

Prerequisites:

- The Polarian SVN Plugins for Connectors and Team Provider must be installed in the IDE.
- Each SVN repository to pull objects from must be defined in the SVN Repositories view.

From the SVN Repositories view, expand the repository and right-click on any file or folder within a trunk or tag. The following options are displayed:


Contents (default repository)
Contents
Differences (default repository)
Differences
MDCMS SVN request
MDCMS SVN commit


MDCMS SVN request
MDCMS SVN commit
Commit newest revision (default repository)
Commit newest revision
Commit to revision ... (default repository)
Commit to revision ...

Request Contents – request some or all of the files/folders that are contained in the selection.

Request Differences – request adds, changes or deletes that have occurred between 2 revisions. If both revisions are tagged, the tags themselves can be selected at the same time to automatically check the differences between them.

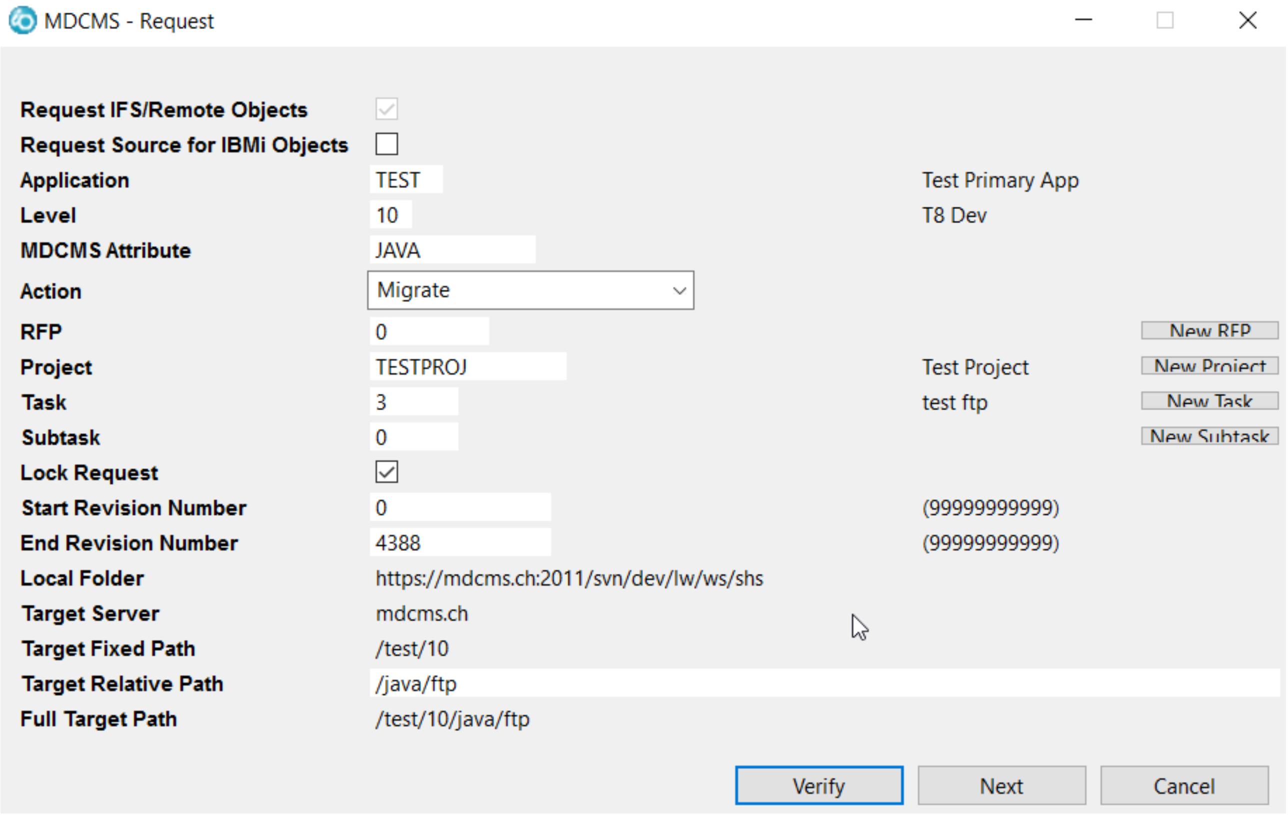
Commit newest revision – copy the newest version of a file to MDCMS for any matching existing requests.

Commit to revision… – copy specific version of a file to MDCMS for any matching existing requests.

11.6 Request – Initial Dialog

Regardless of the request option, the following initial Request dialog is presented to provide the developer with the opportunity to select the originating location, RFP, Project, Task and target attribute and relative path. The fields enabled will vary slightly depending on the origin of the request.

Example Parameter Selection for SVN:



MDCMS - Request

Request IFS/Remote Objects	☑		
Request Source for IBMi Objects	☐		
Application	TEST	Test Primary App	
Level	10	T8 Dev	
MDCMS Attribute	JAVA		
Action	Migrate		
RFP	0		New RFP
Project	TESTPROJ	Test Project	New Project
Task	3	test ftp	New Task
Subtask	0		New Subtask
Lock Request	☑		
Start Revision Number	0	(99999999999)	
End Revision Number	4388	(99999999999)	
Local Folder	https://mdcms.ch:2011/svn/dev/lw/ws/shs		
Target Server	mdcms.ch		
Target Fixed Path	/test/10		
Target Relative Path	/java/ftp		
Full Target Path	/test/10/java/ftp		

Possible Request parameters:

From Server	The Remote Server Location to request from. The location must already be defined in settings/Remote Server Locations
From Folder	The folder on the Remote Server or in IFS from which to copy folders and files
Request IFS/Remote Objects	When selected, the selected folders and files will be deployed as objects to the selected target file system (IFS or a remote server).
Request Source for IBMi Objects	When selected, the selected files will be deployed as IFS source or source members used to create system or SQL objects on the IBMi
Copy without Requesting	If a workspace folder was selected with option Import from Remote Server or option Import IFS folder, this box can be checked to simply copy the contents from a remote server or IFS folder into the local workspace. If checked, the remaining parameters can be ignored.
Application	MDCMS Application Group name
Level	Target Application level for the request
MDCMS Attribute	The MDCMS Attribute for this object request. Must be a valid MDCMS Attribute for this application level and must be either of type *IFS or *REMOTE
Action	Migrate – add/replace objects in target level with requested objects

	Delete – delete objects in target level
RFP	The RFP number to contain the request (optional at this time, required prior to deployment)
Project	The Project value (optional at this time, required prior to deployment)
Task	A Project Task number (optional)
Subtask	A Project Subtask number (optional)
Lock Request	By default, the Lock Request checkbox will be checked, resulting in the Object Request being locked to prevent someone else from also requesting the same object. Unchecking the box will create the Object Request as unlocked and available for other users to request. A lock will only be granted to the request if the object is not already locked by another Object Request.
Start Revision Number	If requesting from SVN, enter the smallest (oldest) revision number that should be considered for any folders/files in the selected folder.
End Revision Number	If requesting from SVN, enter the largest (newest) revision number that should be considered for any folders/files in the selected folder. If the same file has been committed for more than one of the revisions in the range, the newest revision in the range will be selected.
Target Relative Path	By default, the requested parent folder or file will be deployed directly to the target folder defined by the attribute. If the parent folder or file should be deployed to a subfolder on the target system, enter the relative path of that folder from the point where the attribute folder leaves off. Content assist can be used to browse the currently defined subfolders on the target server or a new path can be entered. This parameter is not used when requesting from Git. When from Git the Relative Path is always based on the path of the Working Tree itself to ensure consistency. If an exception to this rule occurs, the relative path can be overridden by editing the Object Request details.

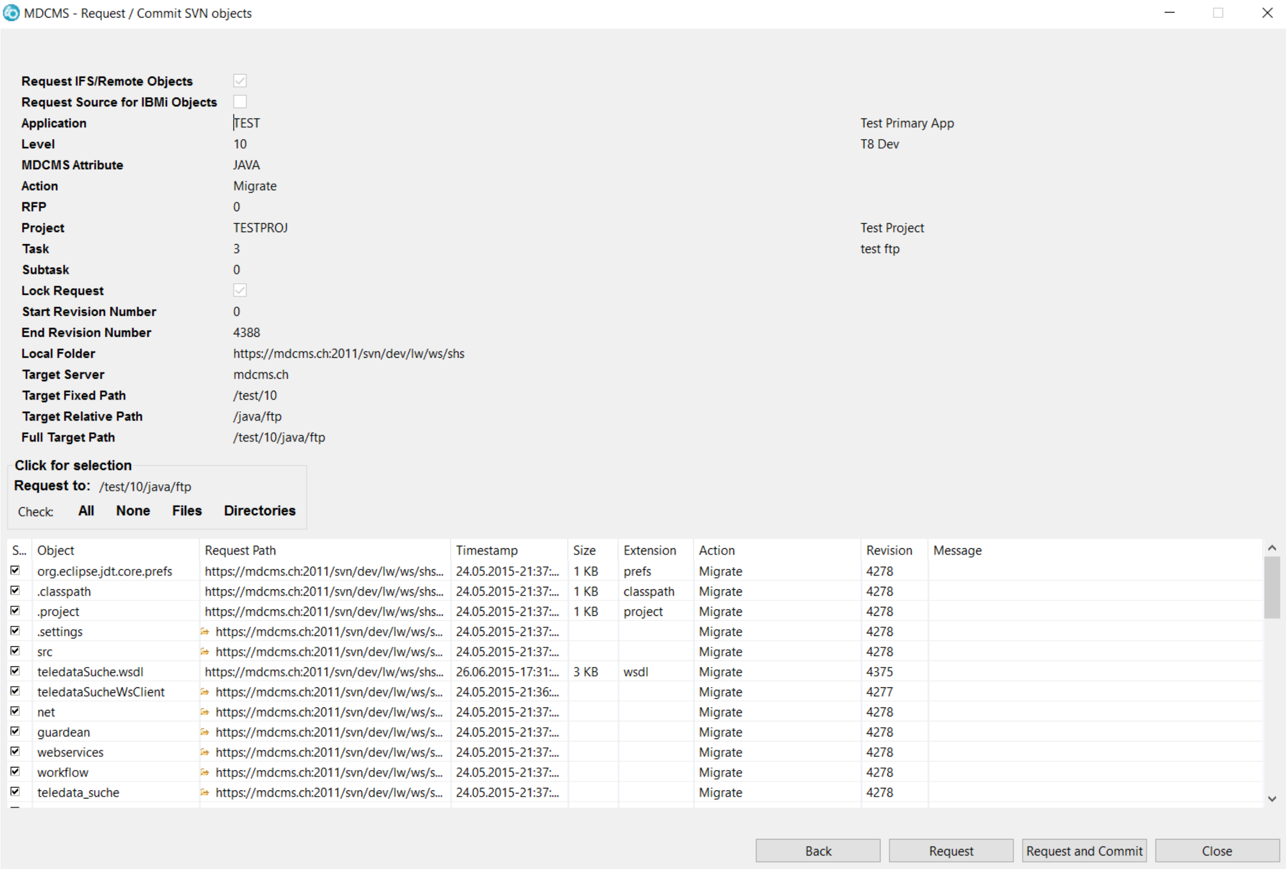
Buttons:
Verify
Verify that the information supplied within the MDCMS request object dialog is valid and refresh the view with descriptions of the fields as well as the full target path.

Next
Continue to the selection and confirmation dialog

Cancel
Cancel the Request

11.7 Request - Selection Confirmation for Objects

The following dialog is displayed when checkbox Request IFS/Remote Objects is selected:



MDCMS - Request / Commit SVN objects

Request IFS/Remote Objects	☑	
Request Source for IBMi Objects	□	
Application	TEST	Test Primary App
Level	10	T8 Dev
MDCMS Attribute	JAVA	
Action	Migrate	
RFP	0	
Project	TESTPROJ	Test Project
Task	3	test ftp
Subtask	0	
Lock Request	☑	
Start Revision Number	0	
End Revision Number	4388	
Local Folder	https://mdcms.ch:2011/svn/dev/lw/ws/shs	
Target Server	mdcms.ch	
Target Fixed Path	/test/10	
Target Relative Path	/java/ftp	
Full Target Path	/test/10/java/ftp	

Click for selection
Request to: /test/10/java/ftp
Check: All None Files Directories

S...	Object	Request Path	Timestamp	Size	Extension	Action	Revision	Message
☑	org.eclipse.jdt.core.prefs	https://mdcms.ch:2011/svn/dev/lw/ws/shs...	24.05.2015-21:37:...	1 KB	prefs	Migrate	4278	
☑	.classpath	https://mdcms.ch:2011/svn/dev/lw/ws/shs...	24.05.2015-21:37:...	1 KB	classpath	Migrate	4278	
☑	.project	https://mdcms.ch:2011/svn/dev/lw/ws/shs...	24.05.2015-21:37:...	1 KB	project	Migrate	4278	
☑	.settings	https://mdcms.ch:2011/svn/dev/lw/ws/s...	24.05.2015-21:37:...			Migrate	4278	
☑	src	https://mdcms.ch:2011/svn/dev/lw/ws/s...	24.05.2015-21:37:...			Migrate	4278	
☑	teledataSuche.wsdl	https://mdcms.ch:2011/svn/dev/lw/ws/shs...	26.06.2015-17:31:...	3 KB	wsdl	Migrate	4375	
☑	teledataSucheWsClient	https://mdcms.ch:2011/svn/dev/lw/ws/s...	24.05.2015-21:36:...			Migrate	4277	
☑	net	https://mdcms.ch:2011/svn/dev/lw/ws/s...	24.05.2015-21:37:...			Migrate	4278	
☑	guardean	https://mdcms.ch:2011/svn/dev/lw/ws/s...	24.05.2015-21:37:...			Migrate	4278	
☑	webservices	https://mdcms.ch:2011/svn/dev/lw/ws/s...	24.05.2015-21:37:...			Migrate	4278	
☑	workflow	https://mdcms.ch:2011/svn/dev/lw/ws/s...	24.05.2015-21:37:...			Migrate	4278	
☑	teledata_suche	https://mdcms.ch:2011/svn/dev/lw/ws/s...	24.05.2015-21:37:...			Migrate	4278	

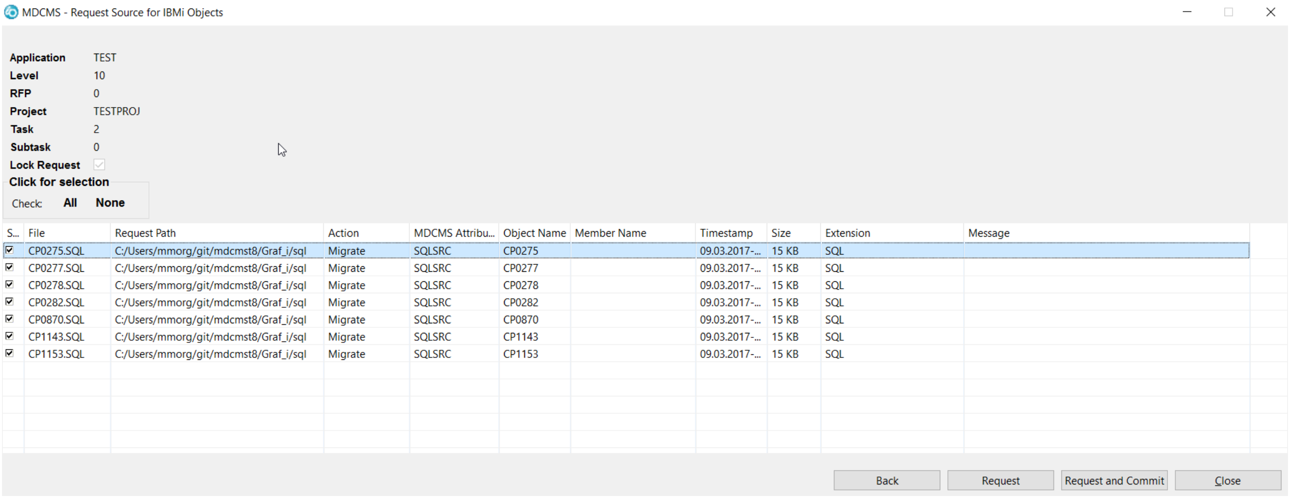
MDOpen displays all selected files and folders as well as any subfolders and files in those subfolders. Select one or more of the items to have them requested. Additionally, All can be clicked to select all items, or None, Files or Directories can be clicked to select only those items.

Buttons:

Back	Return to the Initial Dialog to adjust the parameters
Request	Request the selected items, but don't yet commit them to the MDCMS repository.
Request and Commit	Request the selected items and commit them to the MDCMS repository.
Copy to Workspace	Confirm to copy the selected folders and files to the local workspace without requesting them in MDCMS.
Close	Close the Request Dialog

11.8 Request - Selection Confirmation for Source

The following dialog is displayed when checkbox Request Source for IBMi Objects is selected:



MDCMS - Request Source for IBMi Objects

Application TEST
Level 10
RFP 0
Project TESTPROJ
Task 2
Subtask 0
Lock Request ☑
Click for selection
Check: All None

S...	File	Request Path	Action	MDCMS Attribu...	Object Name	Member Name	Timestamp	Size	Extension	Message
☑	CP0275.SQL	C:/Users/mmorg/git/mdcmst8/Graf_i/sql	Migrate	SQLSRC	CP0275		09.03.2017-...	15 KB	SQL	
☑	CP0277.SQL	C:/Users/mmorg/git/mdcmst8/Graf_i/sql	Migrate	SQLSRC	CP0277		09.03.2017-...	15 KB	SQL	
☑	CP0278.SQL	C:/Users/mmorg/git/mdcmst8/Graf_i/sql	Migrate	SQLSRC	CP0278		09.03.2017-...	15 KB	SQL	
☑	CP0282.SQL	C:/Users/mmorg/git/mdcmst8/Graf_i/sql	Migrate	SQLSRC	CP0282		09.03.2017-...	15 KB	SQL	
☑	CP0870.SQL	C:/Users/mmorg/git/mdcmst8/Graf_i/sql	Migrate	SQLSRC	CP0870		09.03.2017-...	15 KB	SQL	
☑	CP1143.SQL	C:/Users/mmorg/git/mdcmst8/Graf_i/sql	Migrate	SQLSRC	CP1143		09.03.2017-...	15 KB	SQL	
☑	CP1153.SQL	C:/Users/mmorg/git/mdcmst8/Graf_i/sql	Migrate	SQLSRC	CP1153		09.03.2017-...	15 KB	SQL	

MDOpen displays all selected files. Select one or more of the items to have them requested. Additionally, All can be clicked to select all items, or None can be clicked to unselect all items.

Listing Parameters:

Selection	If checked, the file will be included in the checkout
File	The name of the IFS file to be used as source
Request Path	The IFS location where the file will be copied from
Action	the selected action from the initial screen – Migrate or Delete
MDCMS Attribute	The Attribute that defines how an object is to be created and the target source and object libraries for the object and level. MDCMS predetermines the attribute based on Installation History for the object and cross-referencing information. This value can be modified and content assist can be used to select a different attribute from a list. When selecting to check out IFS source, the attribute must be defined with a target source file of *IFS
Object Name	The system or SQL name of the Object that will be created from the requested source. This value can be modified and content assist can be used to select a different object from a list.
Member Name	not applicable when requesting IFS source from MDXREF
Message	The error or success message based on results of attempting to process the selection.

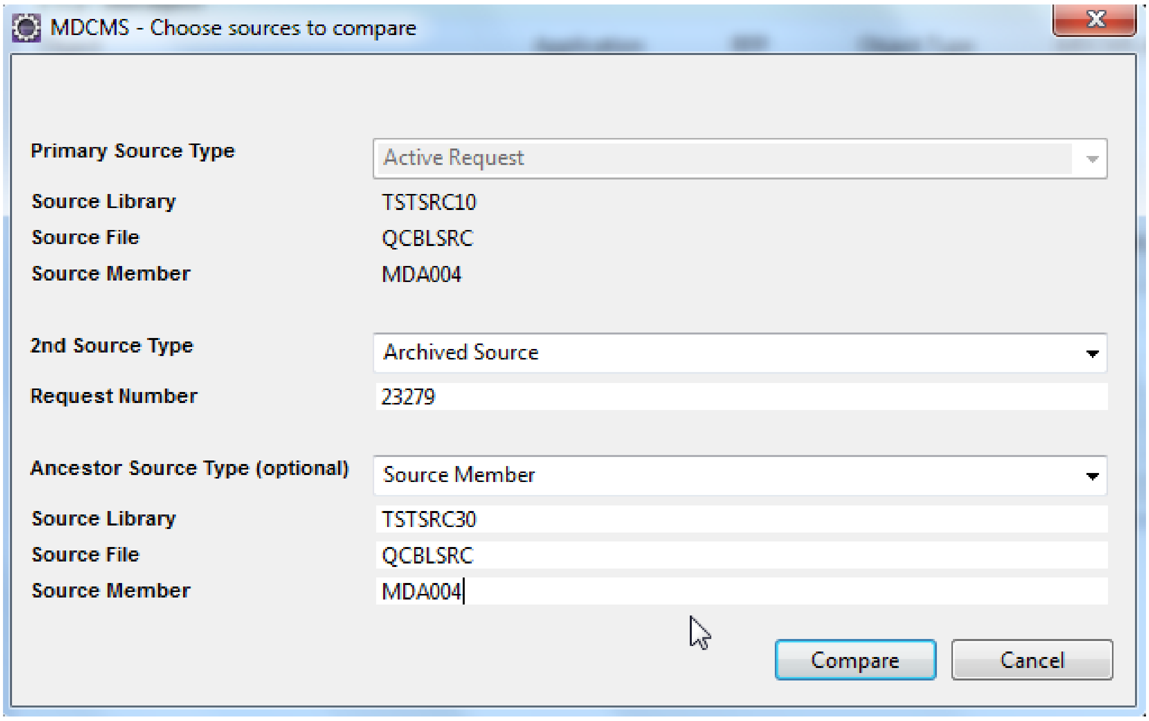
Buttons:

Back	Return to the Initial Dialog to adjust the parameters
Request	Request the selected items, but don't yet commit them to the MDCMS repository.
Request and Commit	Request the selected items and commit them to the MDCMS repository.
Close	Close the Request Dialog

12 Compare Object Source

2-Way and 3-Way comparisons of source can be performed within MDOpen. To initiate a comparison, Right-Click on an Object row within the Object, RFP, Installation History, Project, Task or Subtask listings. You can also click on a file in the workspace to compare that to anything else.

The primary source will be based on the selected object. A 2nd source must be selected for the comparison and a 3rd source can be selected as the Ancestor for a 3-Way comparison.
The type of source (source member, IFS file, local file) can vary for each selection – in other words, you can compare a source member to an IFS file or local file.



MDCMS - Choose sources to compare

Primary Source Type: Active Request
Source Library: TSTSRC10
Source File: QCBLSRC
Source Member: MDA004

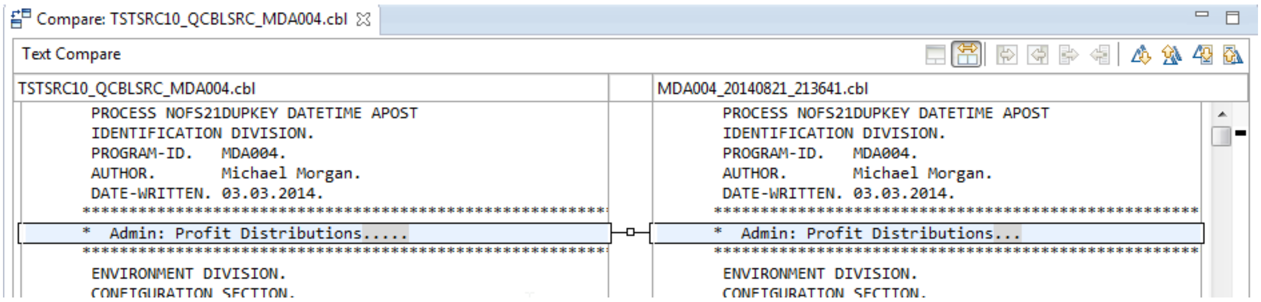
2nd Source Type: Archived Source
Request Number: 23279

Ancestor Source Type (optional): Source Member
Source Library: TSTSRC30
Source File: QCBLSRC
Source Member: MDA004

Possible Source Types:

Active Request	The source in the developer library or folder that has been checked out on behalf of an Object Request. The Request Number of the Request is required and Content Assist can be used to select from a list.
Source Member	A source member residing in a library on the local system. The Source Library, Source File and Source Member name is required and Content Assist can be used to select from a list.
IFS File	A file residing in IFS on the local system. The IFS Folder and Source name is required and Content Assist can be used to select from a list.
Archived Source	The source (member or IFS file) that was replaced by an RFP Installation. The Request Number is required and Content Assist can be used to select from a list of installations for the Primary Object.
File in Workspace	The full path of a file in the local workspace of the IDE. The Browse button can be used to select the file.
Committed Version of File	The currently committed version of the Active Request of the Primary Object.

2-Way Comparison Results:


Compare: TSTSRC10_QCBLSRC_MDA004.cbl
Text Compare
TSTSRC10_QCBLSRC_MDA004.cbl
MDA004_20140821_213641.cbl

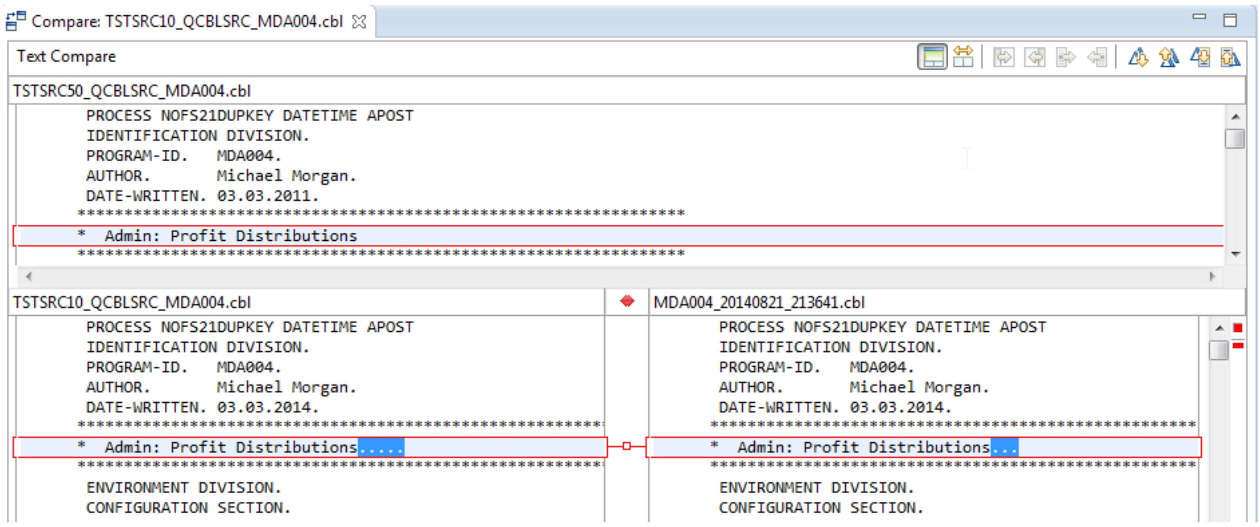
```
       PROCESS NOFS21DUPKEY DATETIME APOST
       IDENTIFICATION DIVISION.
       PROGRAM-ID.    MDA004.
       AUTHOR.        Michael Morgan.
       DATE-WRITTEN. 03.03.2014.
      ***********************************************************
      *  Admin: Profit Distributions.....
      ***********************************************************
       ENVIRONMENT DIVISION.
       CONFIGURATION SECTION.
```

```
       PROCESS NOFS21DUPKEY DATETIME APOST
       IDENTIFICATION DIVISION.
       PROGRAM-ID.    MDA004.
       AUTHOR.        Michael Morgan.
       DATE-WRITTEN. 03.03.2014.
      ***********************************************************
      *  Admin: Profit Distributions...
      ***********************************************************
       ENVIRONMENT DIVISION.
       CONFIGURATION SECTION.
```

The left pane shows the source code of the Primary Source and the right pane shows the source code of the 2nd source. Differences between the 2 versions of the source are marked in gray.

The icons can be clicked to navigate up or down the source to the next difference.

3-Way Comparison Results:


Compare: TSTSRC10_QCBLSRC_MDA004.cbl
Text Compare
TSTSRC50_QCBLSRC_MDA004.cbl
TSTSRC10_QCBLSRC_MDA004.cbl
MDA004_20140821_213641.cbl

```
       PROCESS NOFS21DUPKEY DATETIME APOST
       IDENTIFICATION DIVISION.
       PROGRAM-ID.    MDA004.
       AUTHOR.        Michael Morgan.
       DATE-WRITTEN. 03.03.2011.
      **************************************************************************
      *  Admin: Profit Distributions
      **************************************************************************
```

```
       PROCESS NOFS21DUPKEY DATETIME APOST
       IDENTIFICATION DIVISION.
       PROGRAM-ID.    MDA004.
       AUTHOR.        Michael Morgan.
       DATE-WRITTEN. 03.03.2014.
      ***********************************************************
      *  Admin: Profit Distributions.....
      ***********************************************************
       ENVIRONMENT DIVISION.
       CONFIGURATION SECTION.
```

```
       PROCESS NOFS21DUPKEY DATETIME APOST
       IDENTIFICATION DIVISION.
       PROGRAM-ID.    MDA004.
       AUTHOR.        Michael Morgan.
       DATE-WRITTEN. 03.03.2014.
      ***********************************************************
      *  Admin: Profit Distributions...
      ***********************************************************
       ENVIRONMENT DIVISION.
       CONFIGURATION SECTION.
```

The left pane shows the source code of the Primary Source and the right pane shows the source code of the 2nd source. Differences between the 2 versions of the source are marked in gray. Differences between the 2 versions of the source that are also different to the original version of the ancestor are marked in red which denotes a conflict.

The ancestor pane, when displayed, is located above the 2 comparison panes. The icon can be clicked to show or hide the ancestor pane.

The icons can be clicked to navigate up or down the source to the next difference.

The icons can be clicked to navigate up or down the source to the next conflict.

If differences on a row are only due to a difference in the amount of whitespace, the compare function can be set to ignore this difference by setting the MDOpen preference “Ignore Whitespace in Compare Function” to true.

13 Data Transformation

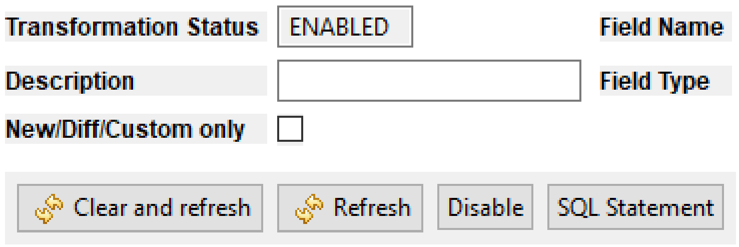
Data Transformation gives you the ability to manipulate the contents of existing columns, or set a value for new columns, when a Physical File (or SQL Table) is modified. During the installation of the file, MDTransform will then be used for the copy of the existing data from the old version of the file to the new version, using SQL syntax to map the column values for each row.

This feature allows you to avoid having to code special programs to handle data copying when a CPYF command would fail or be insufficient.

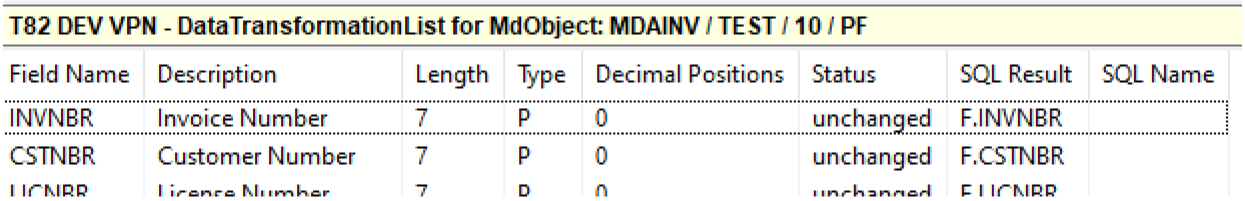
To view/modify the mapping values for each column in the new version of the file, right-click on the Object Request for the file and select option Data Transform.

Prerequisites

- MDTransform must be licensed on the partition that you are connecting to
- The file must already have been created in your developer library. Use option Create Object to do so.
- The file must exist in the target library of the Application Level or chain of levels for the Object Request.



Transformation Status	ENABLED	Field Name	
Description		Field Type	-
New/Diff/Custom only	□		



T82 DEV VPN - DataTransformationList for MdObject: MDAINV / TEST / 10 / PF

Field Name	Description	Length	Type	Decimal Positions	Status	SQL Result	SQL Name
INVNBR	Invoice Number	7	P	0	unchanged	F.INVNBR	
CSTNBR	Customer Number	7	P	0	unchanged	F.CSTNBR	
LICNBR	License Number	7	P	0	unchanged	F.LICNBR	

Disable/Enable Button

By default, MDTransform is used to copy rows from the old version of a changed file to a new version, and it's required if MDRapid will be used to perform the copy.

However, if you wish to use the traditional CPYF command or a custom Data Copy exit point command instead, click the Disable button in the header.

MDTransform will also disable by default if the file contains columns introduced in V7R3 with the GENERATED ALWAYS expression, since using an SQL INSERT statement would cause the original values for those columns to be replaced by new values.

If MDTransform had been disabled for a file, but should be used, then click Enable. Any Data Copy exit point commands are ignored when enabled.

SQL Statement

Click this button to see the syntax of the SQL statement that MDTransform will use during the copy. The statement can be copied to the clipboard to attempt in an SQL session to test and troubleshoot the syntax.

Defining a Custom Value
To define a custom value for a column, left-click on that column to open the editor. Then modify the value for the SQL Result. This must be in SQL syntax and can be anything a SELECT statement allows, including the use of functions.

Removing a Custom Value
To return to the default value, right-click on the column and select option Remove Custom Transformation.

14 Object History

The Object History view provides a listing of all objects that have been installed using MDCMS. There are a significant number of filters available to limit the listing.
Left-clicking on the row will bring up all details about the object installation.

The following right-click or favourite options for a row in the Object History view are available:

View Archived Source
If source is archived for a given object in the installed RFP, this option will be enabled. When selected, the source code is pulled out of the archive and displayed in the editor defined for the given file type in display mode.

Get Archived Source
If source is archived for a given object in the installed RFP, this option will be enabled. When selected, the archived source member can be copied to a source file in a non-managed library or the archived IFS file can be copied to a non-managed IFS folder.

Request Archived Source
If source is archived for a given object in the installed RFP, this option will be enabled. When selected, a new object request can be created and the archived source will be placed in the developer's library to work with and migrate from. A dialog will appear to confirm/change the object, RFP and project information for creating the object request.

Compare
If source is archived for a given object in the installed RFP, this option will be enabled. When selected, the Compare prompt is provided to compare the Archived Source with 1 or 2 other versions of source.
The Ctrl key can be used to multi-select 2 or 3 rows. When multiple rows are selected and the Compare option is selected, the compare dialog will be pre-filled with all of the selected archived generations to easily compare different versions to each other.
See the Section Compare Object Source for more details.

Compare with Current Version
If source is archived for a given object in the installed RFP, this option will be enabled. When selected, the Compare prompt is provided to compare the Archived Source with the current version of the source in the target level or level chain. See the Section Compare Object Source for more details.

Object Commands
If Commands were executed for a given object in the installed RFP, this option will be enabled. When selected, the list of commands will be displayed.

Object Scripts
If Scripts were executed for a given object in the installed RFP, this option will be enabled. When selected, the list of scripts will be displayed.

Custom Data Transformations
If a table was installed with custom column values for the copy of data from the old version of the table to the new version of the table using MDTransform, this option will be enabled. When selected, the list of custom column values will be displayed.

MDXREF
Open the MDXREF Objects view filtered by the object or SQL name, application and level

15 Projects

Projects are the uppermost element in the project hierarchy to explain why a change is being performed. If permitted, Object Requests can be assigned directly to a project, or to tasks or subtasks within a project.

15.1 Project Listing

The ***MdProjectView*** is accessed by selecting ***Projects*** in the Repository window.

Listing Parameters:

Project	A 12-character unique ID for the project
Tasks	Total number of tasks defined for the project. Left-click on this field in a row to open the Task view filtered by the row's project.
Priority	the project priority from 1 (critical) to 5 (optional)
Status	the project status. May be a fixed value delivered with MDCMS or can be a custom value. Left-click on a status value to transition the project to a new status.
Requester	The creator of the project
Project Type	the project category to aid in filtering and to set certain rules
Project Title	A short description of the project
URL	A link, if defined, to either the project in MDWorkflow or to the first custom URL value for the project
Assign to Group	The user group responsible for carrying out the project
Assign to User	A specific user responsible for carrying out the project
Application	The primary application for the project – used only for filtering
Creation Date	the date the project was created
Completion Date	the date the project is expected to complete if active or the close date if closed.

If a Project row includes a › to the left of the Project name in the Project column, that entry can be expanded to show the Object Requests currently assigned to that Project or a Task within the Project.

By default, objects that are requested for recompile or installed objects aren't included. They can be included by checking the boxes in the filters section for the project view.

The information and options available for the object requests are described in Object Requests section.

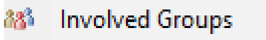
Left-Clicking on a Project row will display the ***Project*** view. The ***Project*** view will contain the following additional detailed information about the project that does not appear in the ***MdProjectView***:

Creation Time	The time of day that the project was created
Hours Estimated	The number of hours estimated to complete the project
Cost Estimated	The estimated cost to complete the project
Hours Actual	The actual number of hours that have been added to time entry for the project
Cost Actual	The accumulated cost based on time entry for the project
Number of Subtasks	The number of Subtasks defined for tasks for the selected project
Custom Fields	Any custom fields that are defined to appear for the given project type and are visible based on the current status of the Project. Custom fields are only permitted if a valid License Key exists for MDWorkflow.

Description	An extended description of the project

Right-click on a project row to add or copy a project, for Time Entry or to request or import objects for the project.

15.2 Involved Groups for Project

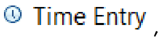
The involvement of User Groups or specific users can be defined at the project level by right-clicking on a project and selecting option  Involved Groups. If authorized, you can then edit the list of currently involved groups or add additional groups to the project.

Fields

Role	Acceptance/Test – the users are involved in testing changes made on behalf of the project Technical – the users are involved in making changes on behalf of the project
Group Type	The type of group that is listed in the Group column (read-only)
Required	Will be true if MDWorkflow requires the Group Type to accept test results before an RFP can continue for certain Application Levels (read-only)
Group	The name of a Group – use content-assist to browse defined values. This is optional for non-required rows
User	A specific user. If for a group, then only that user in the group is involved. Additional rows can be added for other specific users in the group. If a group isn't defined, then any user defined in MDSEC.

15.3 Time Entry Listing

The Time Entry Listing displays all hours entered for Projects, Tasks or Subtasks based on the filter values. The Total Hours field displays the sum of all hours based on the current filter values.

If the listing is invoked from the repository view option  Time Entry, then the filters are set to the last used values for your user profile. If the listing is started from a Project, Task or Subtask, then the filters are set to show all hours for that Project, Task or Subtask.

The date range filters can be easily set to the most common values using the buttons shown representing different spans of time.

15.4 Time Entry Details

Fields

Date	The date that the work was performed
Project	The project for which the work was performed
Task	The task within the project, if applicable, for which the work was performed
Subtask	The subtask within the task, if applicable, for which the work was performed
Phase	The project phase during which the work was performed
User	The user that carried out the work
Hours	The number of hours worked. Any amount after the decimal point is interpreted as a fraction of an hour.
Comment	Any further comments or descriptions of the work performed

Any time entered is then summarized in the Project, Task and Subtask detail views and can also be reported on by defining a report using the MDCMS Time Reports configurator.

16 Tasks

Tasks are a description of a smaller chunk of work to be performed on behalf of a project.

16.1 Task Listing

The ***MdTaskView*** is accessed by selecting ***Tasks*** in the Repository window or by selecting the Tasks option for a specific Project in the Projects view.
By default, the task listing is sorted in ascending order by Due Date and Priority.

Listing Parameters:

Due Date	The date that the task should be completed by
Priority	the task priority from 1 (critical) to 5 (optional)
Project	A 12-character unique ID for the project that the task belongs to
Task	The task number
Subtask	The subtask number, if the row is for a subtask
Task Type	the task category to aid in filtering and to set certain rules and flows
Summary	A short description of the task
Status	the task status. May be a fixed value delivered with MDCMS or can be a custom value. Left-click on a status value to transition the task to a new status.
Requester	The creator of the task
Assign to Group	The user group responsible for carrying out the task
Assign to User	A specific user responsible for carrying out the task
Creation Date	the date the project was created
URL columns	Depending on which web-based project management tools that the given Task is connected to, 0-n columns will be available to navigate to the web page for the task. To do so, click on the 🌐 icon in the column for a given task row. For external tools such as Jira and Service Now, the issue key value within that product is also displayed.

If a Task row includes a > on the left side of the row, that entry can be expanded to show the Object Requests currently assigned to that Task or a subtask within the Task.

By default, subtasks, objects that are requested for recompile, or installed objects aren't included. They can be included by checking the boxes in the filters section of the task view.

The information and options available for the object requests are described in Object Requests section.

Left-Clicking on a Task row will display the ***Task*** view. The ***Task*** view will contain the following additional detailed information about the task that does not appear in the ***MdTaskView***:

Due Time	The time of day that the task is due
Creation Time	The time of day that the task was created
Test Group	The user group responsible for testing the changes made on behalf of the task
Test User	The specific user responsible for testing the changes made on behalf of the task
Application	An informational field specifying an MDCMS application
Location	An informational field specifying a defined OS/400 Location ID
Level	An informational field specifying an MDCMS level

Hours Estimated	The number of hours estimated to complete the task
Cost Estimated	The estimated cost to complete the task
Hours Actual	The actual number of hours that have been added to time entry for the task
Cost Actual	The accumulated cost based on time entry for the task
Subtasks	The number of Subtasks defined for the selected task
Custom Fields	Any custom fields that are defined to appear for the given task type and are visible based on the current status of the task. Custom fields are only permitted if a valid License Key exists for MDWorkflow.
Description	An extended description of the task

Right-click on a task row to add or copy a task, to manage subtasks, for Time Entry or to request or import objects for the task.

16.2 Subtask Listing

Subtasks provide the means to further granulate the work to be performed within a Project Task.

The ***MdSubTaskView*** is accessed by selecting the Subtasks option for a specific Task in the Task view. The fields and options for subtasks are the same as for Tasks.

17 RFP

An RFP is an installation package containing 1 or more objects to migrate from one environment to another.

17.1 RFP Listing

The ***MdRFPView*** is accessed by selecting ▦ **RFP Listing** option in the ***MdRepositoryView.***

Listing Parameters:

User	The owner of the RFP
Application	The target application to which objects should be deployed
Level	The level number of the target environment to which objects should be deployed. Left-click on a level value to view/manage the level details
Status	The RFP Status • **Request pending** (RP) – the RFP is waiting for Workflow acceptance of an RFP in the prior level before the new RFP can be used • **No Requests assigned** (00) – Open, no object request records assigned to RFP • **Requests assigned** (01) – Open, one or more request records assigned to RFP • **Submission Pending** (SP) – RFP has been scheduled for submission and is waiting for the MD Submission service to submit the RFP • **Submission in JOBQ** (YY) – RFP Submission currently in Job Queue • **Submission in Progress** (XX) – RFP Submission in progress, which is the portion of the RFP process that validates and bundles the package into a temporary library. • **Waiting for Approval** (02) – RFP is ready for approval to be granted before the installation steps can occur. • **Data Copy Pending** (CP) – RFP is waiting for launch of MDRapid • **Data Copy Running** (CR) – MDRapid is currently mapping data to new file formats • **Waiting for Installation** (03) – RFP is ready for installation • **Installation Pending** (IP) – RFP has been scheduled for installation and is waiting for the MD Installation service to install the RFP • **Installation in JOBQ** (04)-Installation is submitted and in JOBQ • **Installation in Progress** (XY) – RFP Installation in progress and the target application environment is in the process of being updated. • **Installation Complete** (IC) – The installation of objects is complete and the target environment can be used again. The final archiving and clean-up steps are still running though. • **RFP Complete** (05) – RFP is installed and complete • **RFP Closed without Install** (09) – RFP closed without Install
Problems	If any problems occurred during the most recent attempt to run the RFP • blank – no problems or attempt not yet made • **Errors** – a fatal exception occurred, causing the RFP to automatically roll back • **Warnings** – a warning occurred and has been logged in the deployment log, but the RFP continued.
Description	a description of the RFP
Send Status	if the RFP is installed and the target level contains 1 or more distribution levels, a status of the send to those levels will be shown.

	• **Open** – the RFP hasn't been sent to any distribution levels • **Partial** – the RFP has been sent to some of the distribution levels • **Closed** – the RFP has been successfully sent to all levels or has been closed in the Send Listing. Left-Click on a value in the Send Status column to bring up the Send dialog
Test Status	if the RFP is installed and requires test acceptance before the objects in the RFP can continue to the next step, a status of the testing will be shown. • **Ongoing** – not all of the test groups have accepted the RFP • **Accepted (Provisional)** – all test groups have accepted the RFP, but an authorized user hasn't confirmed acceptance yet. • **Accepted** – the RFP has been accepted and can continue to the next step. • **Rejected (Provisional)** – one or more test groups have rejected the RFP, but an authorized user hasn't confirmed rejection yet. • **Rejected** – the RFP has been rejected and any next steps have been automatically cancelled Left-Click on a value in the Test Status column to bring up the Test Status dialog
RFP Commands	if true, commands are defined to be executed for the specific RFP Left-Click on a value in the RFP Commands column to manage/view the list of commands
RFP Scripts	if true, Qshell or remote platform scripts are defined to be executed for the specific RFP Left-Click on a value in the RFP Scripts column to manage/view the list of scripts

17.2 RFP Details

To view an RFP, click on the RFP entry in the ***MdRFPView.*** That action will display the ***RFP*** view.

Fields

Application	The target Application of the object promotion
Level	The target Level of the object promotion
Description	Description of the RFP
User	The MDCMS User ID of the programmer assigned to this RFP. More than one programmer may have objects requested for an RFP, but only one may be entered here.
Submission Date/Time	For RFPs past the submission step, this is when the submission step was started. For open RFPs, this can be used as a recommended date/time to submit
Reserved until Date/Time	The amount of time to block on the MDWorkflow web app calendar for testing of the installed RFP
Delete Source from Developer Library	If the RFP will be used to promote source that was checked out to a developer's library and will be migrated into the lowest level of an application, the choice may be made to delete the source from the developer's library at the completion of the promotion.
Delete Object from Developer Library	If the RFP will be used to promote an object that was checked out to a developer's library and will be migrated into the lowest level of an application, the choice may be made to delete the object from the developer's library at the completion of the promotion. The object will also be deleted if compiled from source for a request to modify or recompile source.
Delete Source from Import Library	If the RFP will be used to promote source that was requested through an import function and will be migrated into the lowest level of an application, the choice may be made to delete the source from the import library at the completion of the promotion.
Delete Object from Import Library	If the RFP will be used to promote an object that was requested through an import function and will be migrated into the lowest level of an application, the choice may be made to delete the object from the import library at the completion of the promotion.
Delete Job Log when no Warnings Occur	If the RFP completes without any errors or warnings, it will automatically delete the job log spooled file if this box is checked. The RFP Log, including job log entries, will still be stored in the database and accessible from the completed RFP.
Create Requests for next level	If a higher application level exists for direct migration from this target level, new request records can automatically be generated for that level.
Assign new RFP to next level Requests	If the option to automatically create requests for the next level is active, a new RFP number can be automatically generated and assigned to the new request records.
Place RFP in Send List	If a Distribution Level is defined, then a send package can be automatically generated for the level(s) and placed in the RFP Send Listing. The actual sending of the RFP to a remote system is initiated within the RFP Send Listing, unless automatic.

The Set as Default button can be pressed from the RFP editor to retain the preferred value for the Delete checkboxes.

17.3 Copy RFP

To copy an existing RFP in order to create a new RFP right click and select 📋 **Copy.** That action will display the ***Copy MDCMS RFP*** dialog with all fields pre-filled with the values from the selected RFP.

If copying a completed RFP, 2 additional fields are presented:

Copy Object Requests – if selected, a list of all objects on the original RFP are listed and pre-selected with the same action (Modify, Recompile, Update or Delete) as before. Unselect any unneeded objects and click button Process Selections to add those objects to the new RFP. This is a quick way to repeat work for a large block of objects.

Copy RFP Commands/Scripts – if selected, all commands and scripts defined to run at the RFP level for the specific original RFP will be copied to the new RFP.

17.4 Merge Multiple RFPs into One RFP

Any RFPs for the same application level that are in status 00 OR 01 may be merged together. To do so, select each of the rows of RFPs to be merged and then right-click on one of the selected rows and select option Merge.

A confirmation window is then displayed where some of the selected RFPs can be unchecked and the description of the result RFP can be edited. Click Merge to carry out the merge process.

17.5 Reset Status of RFP

This function will return the status of any active RFP to the next lower status. For example, if the RFP is in status 03 (Ready to Install), a reset will return it to 02 (Waiting for Approval) or 01 (Requests Assigned), depending on if automatic approval occurs for the level or not.

If the RFP was in the process of compiling or installing and the job ended abnormally, this function should be used to reset the status so that the RFP can be submitted again.

If the RFP is currently in status 01 (Requests Assigned) and a reset occurs, all object requests currently assigned to the RFP will be removed from the RFP.

17.6 Close RFP

If an RFP has been created, but is not intended to ever by used, and the RFP has no objects assigned to it, it can be closed by right clicking on the RFP and selecting ☒ **CLOSE** . This action will display the 'Close confirmation' dialog with the message '*Are you sure you want to close RFP APPL / nn / nnnnnnn?'* Click OK to close RFP or Cancel to cancel the process.

17.7 Create RFP Objects into Developer Library

If the objects in the RFP are checked out to the developer's library, this option can be taken to create each of the objects in the order of the compile sequence. An attempt to compile will be made for applicable object. Upon completion, review column Create Sts in the embedded object list for any objects with status Error.

17.8 List Spooled Files for an RFP

If the RFP has previously been submitted, the spooled files generated by the RFP, such as compile listings or job logs, can be viewed and/or deleted using this function.

17.9 View the RFP Deployment Log

If an attempt has been made to submit the RFP for deployment, the deployment log can be viewed. Each entry in the log provides a description of the step taken. The step can then be expanded to see all job log entries that occurred during the processing of that step. This can be very helpful in understanding what occurred during the RFP processing, particularly if the RFP failed or warnings occurred.

17.10 List Projects for an RFP

To view the list of all Projects assigned to Object Requests for an RFP, right-click on the RFP and select option Projects. The standard Project List view is then displayed with the filters set to the selected Application and RFP.

17.11 List Tasks for an RFP

To view the list of all Tasks assigned to Object Requests for an RFP, right-click on the RFP and select option Tasks. The standard Task List view is then displayed with the filters set to the selected Application and RFP.

17.12 List Object Requests for an RFP

To view the list of all active Object Requests for an RFP, left-click on the expansion ▷ arrow to the left of the RFP. If the expansion arrow isn't displayed, then there aren't any Object Requests currently assigned to it.

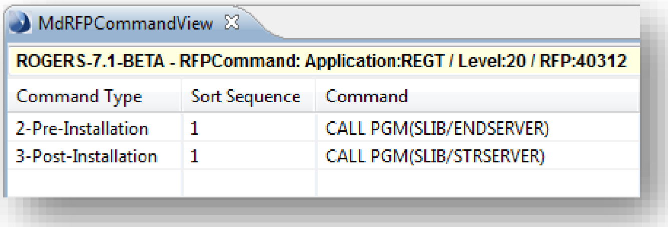
All options available from the main Object Listing are also available from the Object Listings within RFPs.

17.13 Adding a New Object Request to RFP

Right click any RFP within their RFP List view and select ⊞ **New Object Request** and the ***Add MdObject*** dialog is displayed with the Application, Level and RFP number preselected. The remainder of the Object Request Process is identical to the process described in the Objects chapter.

17.14 RFP Commands

To view and manage the commands that will each run once for a specific RFP, right-click on that RFP and select option **RFP Commands.** That action will display the ***MdRFPCommandView***.



MdRFPCommandView

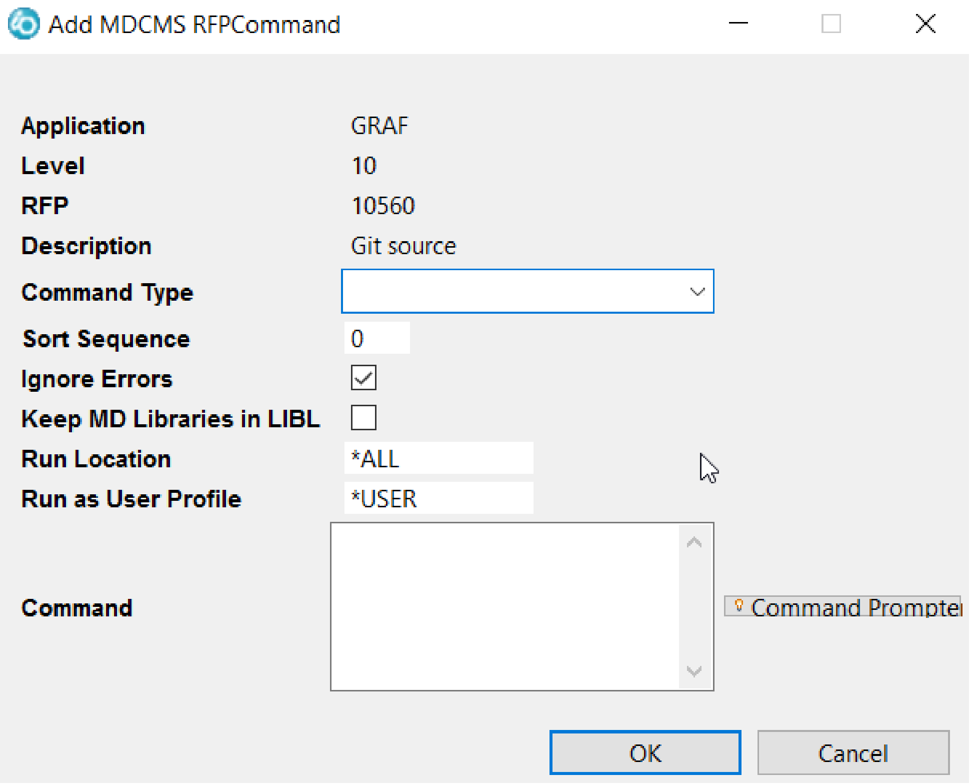
ROGERS-7.1-BETA - RFPCommand: Application:REGT / Level:20 / RFP:40312

Command Type	Sort Sequence	Command
2-Pre-Installation	1	CALL PGM(SLIB/ENDSERVER)
3-Post-Installation	1	CALL PGM(SLIB/STRSERVER)

A right click option within the ***MdRFPCommandView*** will enable context menu options to add a new command, copy an existing command to a new command and to delete an existing command.

Command Details

To create a new command right click within the view and select ⊞ **New Command.** That action will display the ***Add MDCMS RFPCommand*** dialog.



Add MDCMS RFPCommand

Application	GRAF
Level	10
RFP	10560
Description	Git source
Command Type	
Sort Sequence	0
Ignore Errors	☑
Keep MD Libraries in LIBL	□
Run Location	*ALL
Run as User Profile	*USER
Command	

Command Parameters

Command Type	The type or exit point within the RFP process when the command should be executed
Sort Sequence	If multiple commands are defined for the same RFP and Command Type, this value is used to ensure they are executed in the appropriate order
Ignore Errors	Whether or not the RFP processing should continue if the commands fails to execute. This can only be false for types up to and including Post-Installation.
Keep MD Libraries in LIBL	Whether or not the MDCMS product libraries should remain in the library list during the execution of the command. This should be true if the command is a MDCMS API. This should be false if the underlying MDCMS objects could have the same name as objects within your application
Run Location	*ALL – command will run for every level that the RFP is promoted to *LOCAL – run command only for levels on this partition *LOCLVL – run command only for this level *REMOTE – run command on all target partitions, but not this partition specific location – only run command for levels on the specified partition
Run as User Profile	*USER – the user defined for the job description for the level executed the command otherwise, entire the User profile ID of a user to run the command. You must have *USE authority to the profile in order to save it on the command.
Command	The command string to be executed. Content assist is available within the string to insert wildcards that are replaced at runtime by the applicable values. The Command Prompter can be used to prompt for all valid parameters for the command

Options are also available from the list to copy or delete a command.

17.15 RFP Scripts

IFS or Remote Server Scripts can be defined to run for a specific RFP. See the section for Attribute Scripts for more information about the script parameters.

One additional parameter provided when for a specific RFP is the Attribute for Server/Folder Settings. A valid *IFS or *REMOTE attribute must be entered here so that MDCMS will know the location for the execution of the script.

17.16 Compare RFP with Branch

This function allows you to compare the source in an RFP with source in a target branch.

The most common use case is the following:
A long-running project has been worked on in a separate Application Branch, for example a branch using levels 11 for branch development and 31 for branch testing. When the project is finally ready to be sent to the trunk (level 10) for integration into the trunk and eventual deployment to production, all of the accepted changes are merged into an RFP and sent locally to the trunk. Once received into level 10, the source is ready for the merge of any trunk changes that occurred while the project was going on.
In this case, the Target Branch Level would be set to 10 and the Conflict Branch Level would be set to 11.

This option is permitted for open or closed RFPs. If the option is used on an open RFP with checked -out source, the source can be edited directly from the function.

Filter Settings

Target Branch Level	The lowest level of a chain of levels for the same application that the RFP belongs to. It can be for the same level chain as the RFP or something else. This is a required field.
Only New/Different	False – every source in the RFP is listed True – only source that is new or different compared to the target branch is listed
Omit Commits	If the only difference between the source for an object in the RFP and the source in the target branch is due to comments, this checkbox can be set to True to omit that source from the list.
Only Conflicts	When true, and a Conflict Branch Level value is entered in the field, then only source that is different and was installed on the target branch since it was checked out for the conflict branch will be considered.
Conflict Branch Level	The lowest level of a chain of levels for a different branch than the Target Branch. The Conflict Branch Level will typically be the level where the Project work was performed. Source is considered to be in Conflict when it is different and was installed on the target branch since it was checked out for the conflict branch.
Source Name	Filter the list to source names containing the filter value
MDCMS Attribute	Filter the list to source with an MDCMS Attribute containing the filter value

Once the filter values have been entered, click the Create/Refresh List button to perform the comparison and list the results.

Status Values

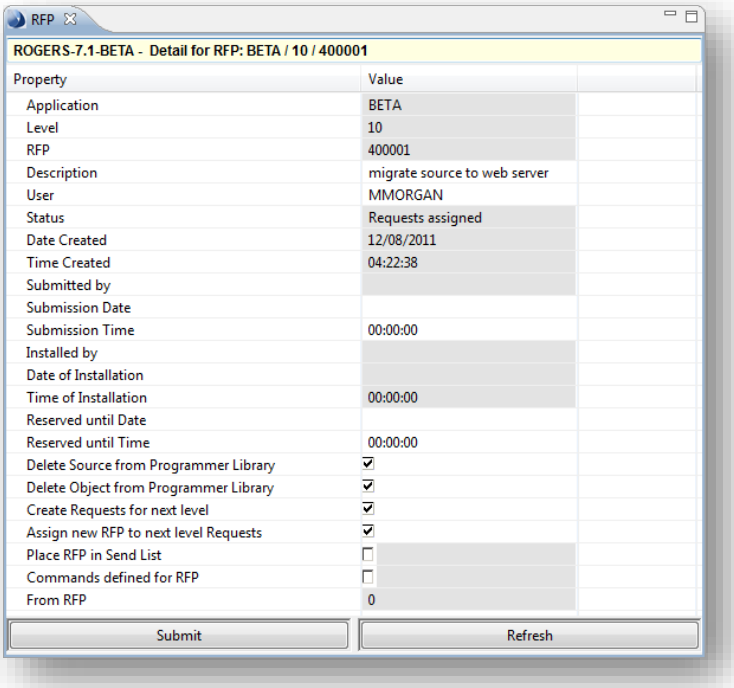
Conflict	The source is different and was installed on the target branch since it was checked out for the conflict branch
Different	The source is different, but nothing has occurred on the target branch with the source since it was checked out in the other branch. Or, a Conflict Branch Level value wasn't used for the comparison.
New	The source doesn't exist in the target branch
Same	No differences in the source code

Possible Actions

🔍	View the source in the RFP, by clicking on the icon after the RFP File column. View the source in the target branch by clicking on the icon after the Branch File icon.
✏	Edit the source in the RFP
⇄	Compare and merge the source code into the RFP version of the source

17.17 Promoting an RFP

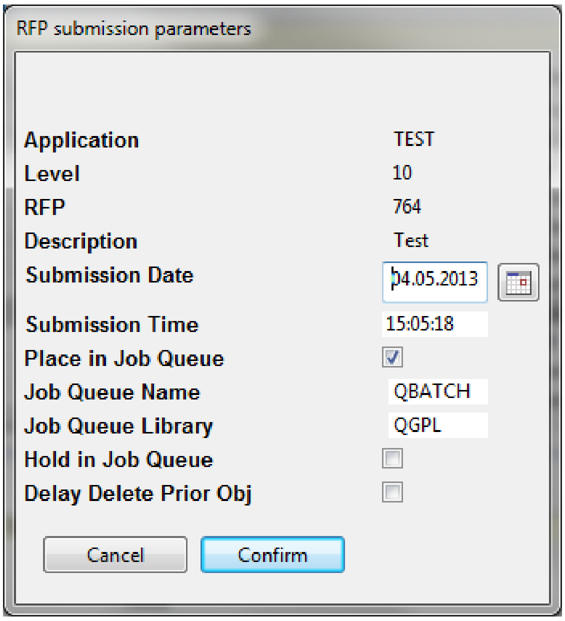
When one (1) or more objects are assigned to an RFP, the Request for Promotion becomes eligible to be submitted. From the ***MdRFPView,*** left-click on an RFP. That action will display the ***RFP*** view.



ROGERS-7.1-BETA - Detail for RFP: BETA / 10 / 400001

Property	Value
Application	BETA
Level	10
RFP	400001
Description	migrate source to web server
User	MMORGAN
Status	Requests assigned
Date Created	12/08/2011
Time Created	04:22:38
Submitted by	
Submission Date	
Submission Time	00:00:00
Installed by	
Date of Installation	
Time of Installation	00:00:00
Reserved until Date	
Reserved until Time	00:00:00
Delete Source from Programmer Library	☑
Delete Object from Programmer Library	☑
Create Requests for next level	☑
Assign new RFP to next level Requests	☑
Place RFP in Send List	☐
Commands defined for RFP	☐
From RFP	0

Clicking the ***Submit*** button causes MDCMS to make a series of checks for potential problems. Any warnings or errors detected will be presented in a series of windows. Right-Click on a row in the error windows to select an option to fix the issue directly from the window (where applicable). Once all errors are eliminated, the following confirmation window is presented where the date, time and job queue information can be modified:



RFP submission parameters

Application	TEST
Level	10
RFP	764
Description	Test
Submission Date	04.05.2013
Submission Time	15:05:18
Place in Job Queue	☑
Job Queue Name	QBATCH
Job Queue Library	QGPL
Hold in Job Queue	☐
Delay Delete Prior Obj	☐

NOTE: If the RFP job is not placed in the Job Queue, then the status is changed to SP for Submission Pending and will wait until the RFP Submission API (MDSBMRFP) submits the RFP.

17.18 The Batch Installation Process

Once a request for promotion has been submitted, the actual process of installing the new or modified source and objects takes place. Listed below are the steps that this process goes through and the result of each step.

17.18.1 The Source/Object Preparation Steps

Temporary MDCMS libraries (CMSxxxxxx and SAVxxxxxx where xxxxxx is the RFP number) are created to handle the entire process so that no permanent changes take place until MDCMS is certain that all is ready.

1- Object-Level pre-compile commands are performed.
2- Attribute-Level pre-compile commands are performed.
3- RFP-Level pre-compile commands are performed
4- The source members and non-compiled objects are checked for existence.
5- If source is to be compiled, it is compiled at this time using the job description's library list that is defined for the Application Level. The compile command itself is based on the command definition for the attribute, unless an object-level compile command has been defined.
6- The source and/or objects are placed in the CMSxxxxxx library
7- Object-Level post-compile commands are performed
8- Attribute-Level post-compile commands are performed.
9- RFP-Level post-compile commands are performed
10- The source and objects are signed by MDCMS to ensure that manual changes are detected.

If a failure occurs during these steps, a message will be sent to the user that submitted the job describing why the failure occurred. For additional detail, the spooled files for the job should be reviewed. The RFP will remain at status Requests assigned (01) and can be re-requested for submittal from the **MdRFPView**.

17.18.2 Source and Object Signing

MDCMS uses an SHA-1 encryption algorithm with a HMAC key unique to each distinct object migration chain to create a 1-way signature for each source and object that has been prepared to be installed. The initial signature is applied when it is installed into the initial Application Level and this signature is compared to the newly generated signature at higher levels.

If the signature does not match, then this indicates that the source or object was manually modified since installation in the prior level. MDCMS will then require that an authorized user approves the installation before the installation steps will be started.

MDCMS uses a java program running under job name **MDSIGN(instance)** to perform the signature process. The job queue to be used for this job is, by default, the same as for the RFP. If this job should be submitted to a different queue, it can be defined in Data Area MDCMS/MDSTRSIGN whereby characters 1-10 contain the job queue name and characters 11-20 contain the job queue library. The default value of *JOBD will cause the job to be submitted to the job queue defined for the job description.

If the signing job does not function, troubleshooting can be performed by running command MDTESTSIGN from a command line within MDCMS.

17.19 The Installation Steps

1. Object-Level pre-installation commands are performed.

2. Attribute-Level pre-installation commands are performed.
3. RFP-Level pre-installation commands are performed
4. The source and objects are either moved into the next level for modifications and recompiles, removed from the level for deletes, or updated for updates.
5. The authorities are set for each of the promoted objects.
6. All prior members for modified physical files are copied to the new file with option *map/*drop (unless an overriding data copy command is specified) and all system (non sql) triggers and journals are reapplied (if data area MDCMS/MDAUTOJRN = 'Y' and MDCMS/MDAUTOTRG = 'Y'). If a logical file is being replaced, all prior members of the file are created for the new file (if data area MDCMS/MDAUTOLFM = 'Y').
7. The MDXREF information is refreshed for the installed objects.
8. Physical file data is copied into the next level for *DATA requests.
9. Object-Level post-installation commands are performed.
10. Attribute-Level post-installation commands are performed.
11. RFP-Level post-installation commands are performed

If a failure or warning occurs during these steps, a message will be sent to the user that submitted the job describing why the failure occurred. For additional detail, the spooled files for the job should be reviewed. The RFP will remain at status 03 and can be re-requested from the Install Promotions option (menu option 4). If the RFP is not in the list, it needs to be reset by RFP Maintenance.

17.20 The Archiving and Cleanup Steps

1. All replaced source is archived if the Application Level is defined to allow archiving. Replaced objects will be zip compressed and archived to the MDCMS IFS path, if they are not compiled from source.
2. If the installation occurred at the lowest Application Level and the RFP is defined to remove the source or objects from the programmer's library, the removal is performed at this time.
3. Installation History records are created for each object.
4. The finished Request detail records are removed.
5. The temporary libraries and spool files are deleted.

17.21 The Set for Next Level Steps

1. If a Distribution Queue is defined, the RFP is placed in the send list. If Auto-Send is set to Y for this Level, the RFP will immediately sent to all Distribution Queues where the Default flag is set to Y.
2. New Request records are created for each promoted source member for the next defined level for that member.
3. Any user-defined commands are set to handle the next level.
4. A new RFP number is generated and automatically assigned to the new Request records.
5. If a source member is already requested for the next level, a Request record will not be created. To help ensure that the programmer realizes this, an error report is generated which lists all sources which could not be requested at the next level because of existing requests.
6. If Auto-Submit is set to Y for the next level, and no errors exist at the next level, and Workflow acceptance of this RFP is not required, the new RFP is submitted to batch.

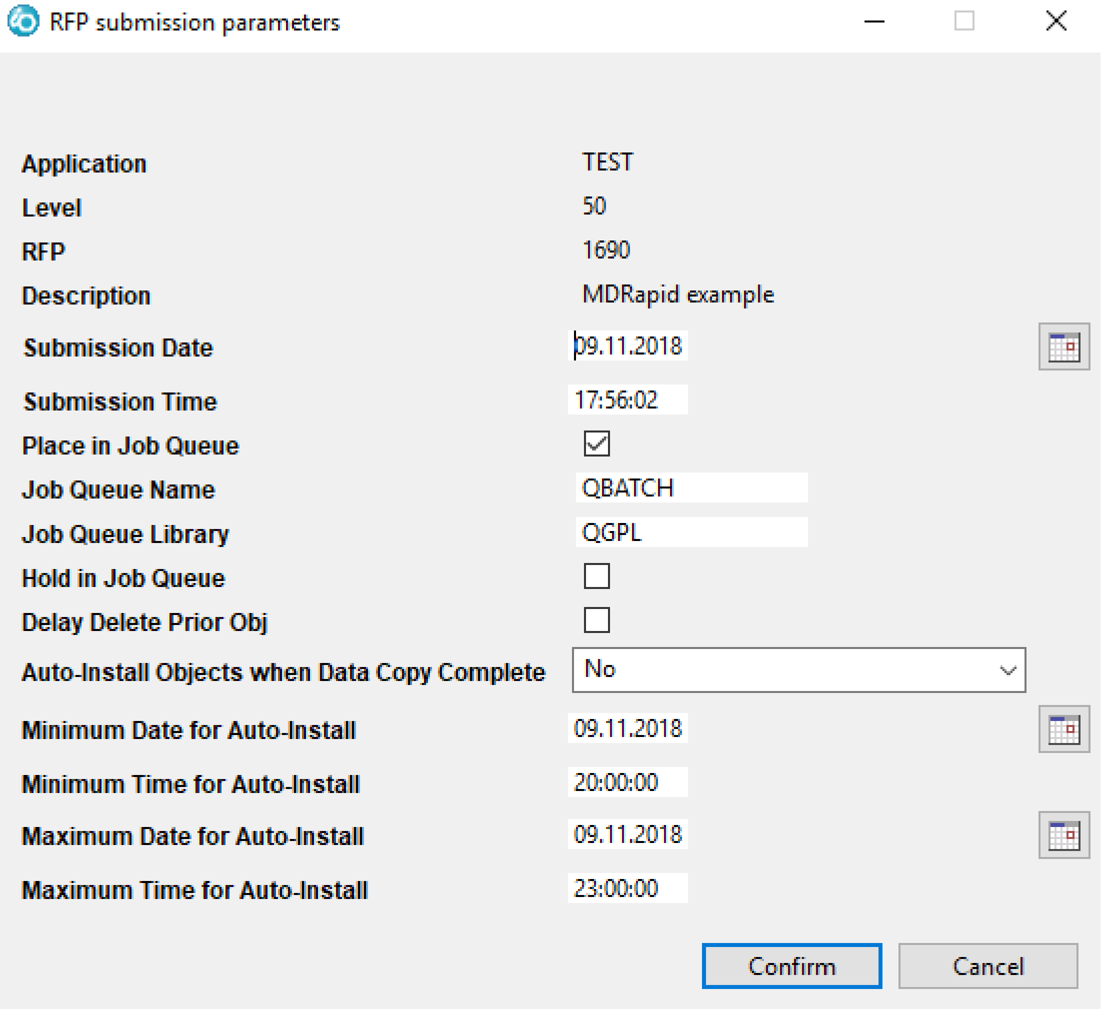
17.22 Approving a Promotion

An RFP is approved for promotion from the ***MdRFPView*.** The RFP creates a temporary library, CMSxxxxxx, and all objects are compiled into that library. If there are no errors, the installation process checks the Application Level parameters and if the Approval flag is set to 'Y', the RFP status is set to *Waiting for Approval* (02). An authorized user must then approve the promotion before it can be installed. To do this, the authorized user will select the RFP from the **MdRFPView** and enter their approval by clicking the **Approve** button at the bottom of the RFP. That action will display the 'Confirm approve' dialog with the message 'Do you wish to approve RFP?' Click OK to Approve or Cancel to cancel the process.

17.23 Launching MDRapid for a Promotion

If one or more physical files on an RFP require MDRapid processing, in order to minimize the amount of downtime for an application, the RFP will go into status CP-Copy Pending once the RFP has been approved. If the RFP's level is set to auto-launch MDRapid, then the launch will commence immediately. Otherwise, the user can launch MDRapid for the RFP from the RFP Detail view by clicking button Launch MDRapid.

This will bring up the following dialog:



RFP submission parameters

Application	TEST
Level	50
RFP	1690
Description	MDRapid example
Submission Date	09.11.2018
Submission Time	17:56:02
Place in Job Queue	☑
Job Queue Name	QBATCH
Job Queue Library	QGPL
Hold in Job Queue	□
Delay Delete Prior Obj	□
Auto-Install Objects when Data Copy Complete	No
Minimum Date for Auto-Install	09.11.2018
Minimum Time for Auto-Install	20:00:00
Maximum Date for Auto-Install	09.11.2018
Maximum Time for Auto-Install	23:00:00

MDRapid Parameters:

Submission Date/Time	When the copy of data from the current version of the files to the new version should commence.
Place in Job Queue	If true, then the SBMJOB command is immediately performed with the scheduled date/time parameters set for actual running in the subsystem. If false, then the RFP will go into pending status and wait for the MDINSRFP command to start the copy.
Job Queue Name/Library	The job queue to submit the MDRapid job to, if applicable. This job only launches MDRapid – the jobs that perform the copying are then submitted to the job queue defined on the MDRapid Usage Template
Hold in Job Queue	If true, the job will be submitted, but not released for processing until released in the job queue.
Delay Delete Prior Obj	If true, the backup libraries won't be deleted after the completion of the installation until the next day.
Auto-Install Objects when Data Copy Complete	Yes – once the Data Copy is complete, the installation will occur automatically No – once the Data Copy is complete, an authorized user or process will manually start the installation Yes when in Time Window – once the Data Copy is complete, the date/time is checked. If prior to the window, the sync process will continue until the window is hit and then commence with the installation. If after the window, an authorized user or process will need to manually start the installation.
Min/Max Date/Time	The time window for auto-install, if Auto-Install Objects set to Yes when in Time Window

While MDRapid is running, the live files in the application are still fully accessible to users. The locking of the files won't occur until the installation steps are performed. At that point, MDCMS will only need to move the objects rather than copy any data so that the lock window is very short (a few seconds).

This is done by first copying all the existing data from the old to the new, while performing any column transformations necessary. Then, all journal transactions that occur until the installation happens are also transformed and copied to the new version of the files, including inserts, updates and deletes.

The status of MDRapid can be viewed using option MDRapid Copy Status on the RFP or on one of the files in the RFP. This will bring up a view listing all of the files being handled by MDRapid.

Status Options:

End Job	End the job responsible for the copy of the specific file
Hold Job	Hold the job responsible for the copy of the specific file
Restart from Beginning	If the job was ended, restart the copy process for the file from the very beginning
Restart at Next Record	If the job was ended, restart the copy process where it left off

17.24 Installing a Promotion

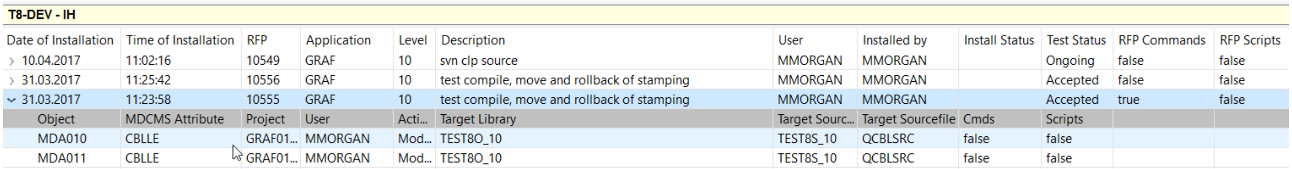
An RFP is submitted for promotion from the ***MdRFPView*.** The RFP creates a temporary library, CMSxxxxxx, and all objects are compiled into that library. If there are no errors, the installation process checks the Application Level parameters and if the Auto-Install flag is set to ‘N’, the RFP status is set to *Waiting for installation* (03). An authorized user must then select the promotion for installation before the objects are actually installed into the application levels target libraries. To do this the authorized user can use the **Install** button. Clicking the Install button will display the ‘Set Install date and time’ dialog where the user can set the Scheduled Date and Scheduled Time and click OK to install the RFP or Cancel to cancel the process.

18 RFP Installation History

Historical information is kept in MDCMS for every object that is promoted. If the promotion of an object results in the change or deletion of previously existing source, the previous version of the source will be archived. Installation History stores between 1 and an infinite number of generations of the source for each object at each Application Level. If an object does not have, or wasn't compiled from, source code, then the object itself is compressed and archived in the IFS directory MDCMS/ARCHIVE/. The number of generations to store is set per Application Level. A different number of generations can be set for each *DATA or *DTAGRP attribute.

Completed RFPs can be viewed from the RFP Installation History listing, or from the RFP Listing, if the RFP listing isn't filtered to an active status.

The RFP Installation History listing is accessed by right clicking on the Repository connection in the ***MdRepositoryView*** and selecting ***RFP Installation History***. That action will bring up the ***MdInstallationHistoryView***. This view will present the history of installed or deleted objects and allows for the ability to compare, retrieve or rollback prior versions of objects.



T8-DEV - IH

Date of Installation	Time of Installation	RFP	Application	Level	Description	User	Installed by	Install Status	Test Status	RFP Commands	RFP Scripts
› 10.04.2017	11:02:16	10549	GRAF	10	svn clp source	MMORGAN	MMORGAN		Ongoing	false	false
› 31.03.2017	11:25:42	10556	GRAF	10	test compile, move and rollback of stamping	MMORGAN	MMORGAN		Accepted	false	false
⌄ 31.03.2017	11:23:58	10555	GRAF	10	test compile, move and rollback of stamping	MMORGAN	MMORGAN		Accepted	true	false
Object	MDCMS Attribute	Project	User	Acti...	Target Library	Target Sourc...	Target Sourcefile	Cmds	Scripts		
MDA010	CBLLE	GRAF01...	MMORGAN	Mod...	TEST8O_10	TEST8S_10	QCBLSRC	false	false		
MDA011	CBLLE	GRAF01...	MMORGAN	Mod...	TEST8O_10	TEST8S_10	QCBLSRC	false	false		

The history can be filtered based on many different fields.

Expand an RFP row to list all objects that were installed by the RFP.

18.1 Copy RFP

To copy an existing RFP in order to create a new RFP right click and select **Copy.** That action will display the ***Copy MDCMS RFP*** dialog with all fields pre-filled with the values from the selected RFP.

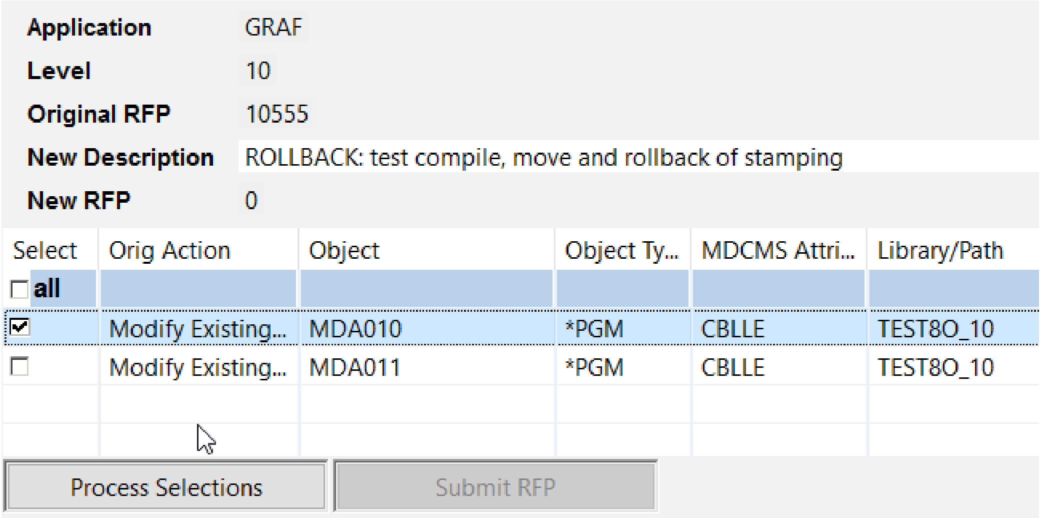
2 additional fields are presented:

Copy Object Requests – if selected, a list of all objects on the original RFP are listed and pre-selected with the same action (Modify, Recompile, Update or Delete) as before. Unselect any unneeded objects and click button Process Selections to add those objects to the new RFP. This is a quick way to repeat work for a large block of objects.

Copy RFP Commands/Scripts – if selected, all commands and scripts defined to run at the RFP level for the specific original RFP will be copied to the new RFP.

18.2 Rollback RFP

Use the Rollback option to reverse the installation of one or more objects in the original RFP.



Application	GRAF
Level	10
Original RFP	10555
New Description	ROLLBACK: test compile, move and rollback of stamping
New RFP	0

Select	Orig Action	Object	Object Ty...	MDCMS Attri...	Library/Path
□ **all**					
☑	Modify Existing...	MDA010	*PGM	CBLLE	TEST8O_10
□	Modify Existing...	MDA011	*PGM	CBLLE	TEST8O_10

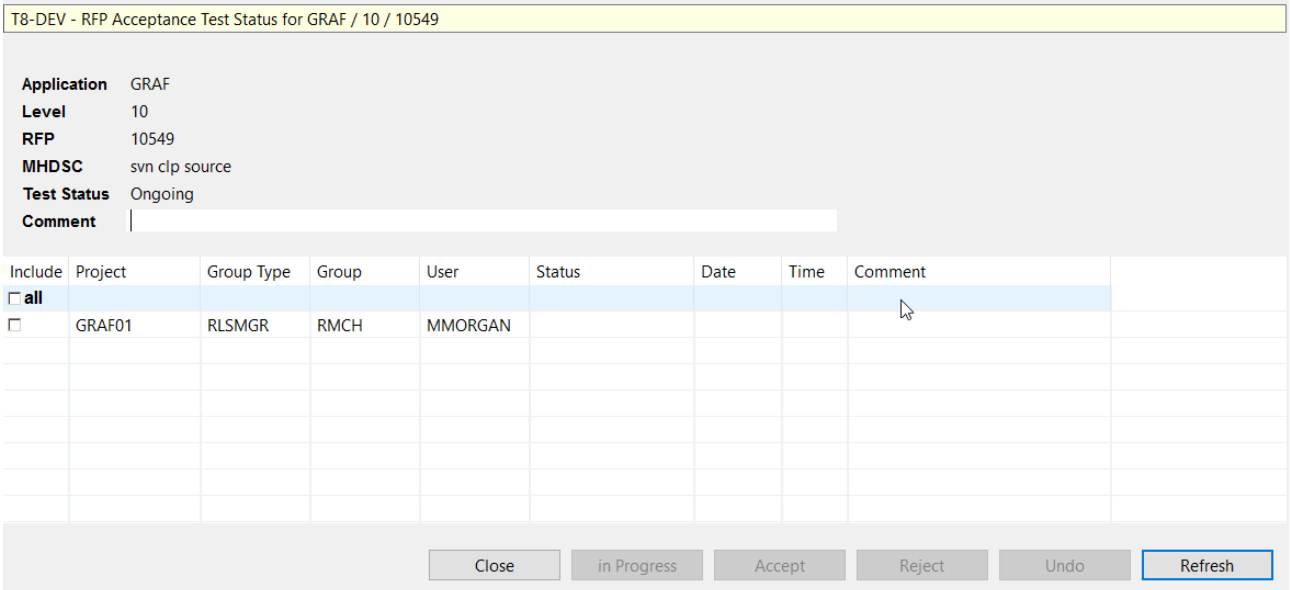
A dialog will appear where one or more of the objects can be selected. Once all necessary objects are selected, click Process Selections to generate a new RFP with the description in the Header. The new RFP can then be submitted immediately by clicking the Submit RFP button or can be kept in the RFP listing to be submitted at a later time. A rollback RFP does the following:

- deletes new objects
- adds deleted objects
- repeats recompiles or updates
- reverts to prior version for modifications

18.3 Acceptance Test Status

If the Application Level of the RFP requires MDWorkflow acceptance before the RFP can continue to the next level of the migration path, option Acceptance Test Status can be used to accept, reject or view the status of the RFP.

You can also left-click on the column Test Status for a row to open the Acceptance Test Status view.



T8-DEV - RFP Acceptance Test Status for GRAF / 10 / 10549

Application	GRAF
Level	10
RFP	10549
MHDSC	svn clp source
Test Status	Ongoing
Comment	

Include	Project	Group Type	Group	User	Status	Date	Time	Comment
□ **all**								
□	GRAF01	RLSMGR	RMCH	MMORGAN				

Test Status
Ongoing – Testing in progress and results are not yet complete
Accepted (Provisional) – Test Users have accepted the results of the test but confirmation has not yet occurred
Accepted – Test acceptance has been confirmed. The next step in the Workflow process may occur.
Accepted (Provisional) – Test Users have accepted the results of the test but confirmation has not yet occurred
Accepted – Test acceptance has been confirmed. The next step in the Workflow process may occur.
Rejected (Provisional) – Test Users have rejected the results of the test but confirmation has not yet occurred
Rejected – Test rejection has been confirmed. The next step in the Workflow process will not occur.

Project List Fields

Check Box	The check box will be displayed for a row if: a) the RFP Test Status has not yet been confirmed and b) the current user is a member of the defined Group and c) either the defined user is blank so that anyone in the Group can edit the Status for the Project, or, the defined user is the current user
Project	The Project ID that is impacted by the RFP. The ID may be clicked to view more details about the Project
Group Type	The Group Type that is responsible for the acceptance of an installed RFP into the specified level.
Group	The Group that is assigned to the Group Type for this Project in order to test the results of the RFP. The Group may be clicked to view more information about the Group
User	If a value exists, then only this user may edit the status. If blank, then any user in the group can edit the status.
Status	The status for that Project. The status for all projects must be Accepted before the RFP Test Status of Accepted can be confirmed. Only the status for 1 project must be Rejected before the RFP Test Status of Rejected can be confirmed.
Date	The date that the current status was set by the user
Time	The time that the current status was set by the user
Comment (in header)	A comment that will be applied to every checked row when a button is clicked. A comment is mandatory for rejected rows

Buttons

In Progress	Set every checked Project to status In Progress, signifying that you are in the process of testing for those Projects.
Accept	Set every checked Project to status Accepted, signifying that you have accepted the testing results for those Projects.
Reject	Set every checked Project to status Rejected, signifying that you have rejected the testing results for those Projects.
Undo	Undo the current status for the checked Projects.
Confirm Acceptance	Once all project rows have been accepted, the Confirm Acceptance button can be clicked by an authorized user to permanently set the RFP to accepted and to allow the next step in the Workflow process to occur for the RFP.
Confirm Rejection	Once 1 or more project rows have been rejected, the Confirm Rejection button can be clicked by an authorized user to permanently set the RFP to rejected and to remove any pending next steps from the Workflow process for the RFP.

18.4 Objects in Installed RFP

Clicking the expand icon › that appears to the left of the date in the Date of Installation column will expand the RFP entry and display the Objects associated with the RFP.

Clicking on any object for an RFP will bring up the ***IHObject*** view.

The ***IHObject*** view will display all details for the selected object, including Project information.

The following options are available:

View Archived Source
If source is archived for a given object in the installed RFP, this option will be enabled. When selected, the source code is pulled out of the archive and displayed in the editor defined for the given file type in display mode.

Get Archived Source
If source is archived for a given object in the installed RFP, this option will be enabled. When selected, the archived source member can be copied to a source file in a non-managed library or the archived IFS file can be copied to a non-managed IFS folder.

Request Archived Source
If source is archived for a given object in the installed RFP, this option will be enabled. When selected, a new object request can be created and the archived source will be placed in the developer's library to work with and migrate from. A dialog will appear to confirm/change the object, RFP and project information for creating the object request.

Compare
If source is archived for a given object in the installed RFP, this option will be enabled. When selected, the Compare prompt is provided to compare the Archived Source with 1 or 2 other versions of source. See the Section Compare Object Source for more details.

Compare with Current Version
If source is archived for a given object in the installed RFP, this option will be enabled. When selected, the Compare prompt is provided to compare the Archived Source with the current version of the source in the target level or level chain. See the Section Compare Object Source for more details.

Object Commands
If Commands were executed for a given object in the installed RFP, this option will be enabled. When selected, the list of commands will be displayed.

Object Scripts
If Scripts were executed for a given object in the installed RFP, this option will be enabled. When selected, the list of scripts will be displayed.

Custom Data Transformations
If a table was installed with custom column values for the copy of data from the old version of the table to the new version of the table using MDTransform, this option will be enabled. When selected, the list of custom column values will be displayed.

MDXREF
Open the MDXREF Objects view filtered by the object or SQL name, application and level

19 RFP Send Listing

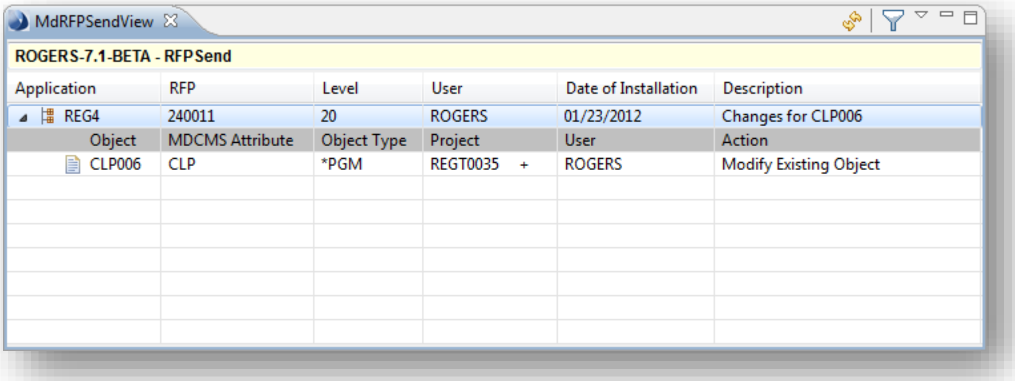
It is possible to control your business applications across your IBMi systems or logical partitions with MDCMS, as long as a licensed copy of MDCMS exists on each of the systems or logical partitions involved.

The first step for specifying what is to be sent to those additional systems is to define the OS/400 locations and distribution levels that will be used for RFP distribution. Refer to those sections in the manual for details on creating and maintaining the required target locations and levels.

When a promotion is installed into an Application level, the RFP with all of its installed objects, commands and scripts will appear in the list as long as:

1) the application/level has at least 1 defined distribution level
2) the flag, **Place RFP in Send Promotion List**, is set to Y at the RFP level

To view the RFP Send Listing, select option **RFP Send Listing** from the repository list.



ROGERS-7.1-BETA - RFPSend

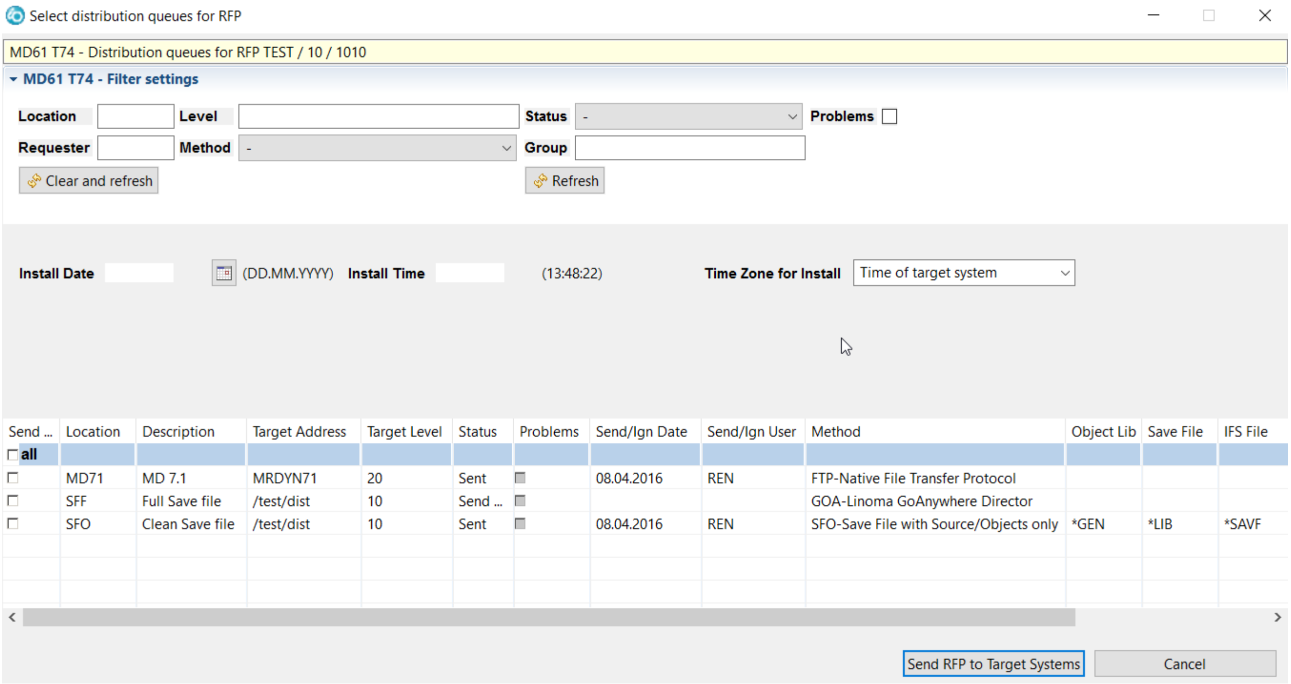
Application	RFP	Level	User	Date of Installation	Description
REG4	240011	20	ROGERS	01/23/2012	Changes for CLP006
Object	MDCMS Attribute	Object Type	Project	User	Action
CLP006	CLP	*PGM	REGT0035 +	ROGERS	Modify Existing Object

The ***MdRFPSendView*** will contain any RFP's that are currently open for distribution to other systems or logical partitions.

To view or reopen RFPs that are closed in the Send Listing, use repository option RFP Send History.

19.1 Send RFP to Target Systems

To send an RFP to the designated target systems or logical partitions right click on the RFP and select option **Send RFP to Target Systems**, or left click on the icon for a row in the list, or left-click on the Send Status column in Installation History. This action will display the Select distribution queues for RFP dialog.



Select distribution queues for RFP

MD61 T74 - Distribution queues for RFP TEST / 10 / 1010

MD61 T74 - Filter settings

Location:
Level:
Status: -
Problems: □
Requester:
Method: -
Group:

Install Date: (DD.MM.YYYY)
Install Time: (13:48:22)
Time Zone for Install: Time of target system

Send ...	Location	Description	Target Address	Target Level	Status	Problems	Send/Ign Date	Send/Ign User	Method	Object Lib	Save File	IFS File
□ **all**												
□	MD71	MD 7.1	MRDYN71	20	Sent	□	08.04.2016	REN	FTP-Native File Transfer Protocol			
□	SFF	Full Save file	/test/dist	10	Send ...	□			GOA-Linoma GoAnywhere Director			
□	SFO	Clean Save file	/test/dist	10	Sent	□	08.04.2016	REN	SFO-Save File with Source/Objects only	*GEN	*LIB	*SAVF

The target levels can be filtered by values entered in the filter settings section of the view. See the MDCMS User Manual for more information about the filter and list fields.

Install Date
The date that the install should take place on the target systems.

If the target level is set to automatic, it will be received, submitted and approved for installation immediately on the target system. Then a 2nd batch job will be submitted to batch that will be scheduled for the date/time provided.

If the target level is set to manual, then a user will need to carry out the manual steps, but the install date/time will default to the value provided in this screen.

If blank, the installation will be scheduled for the same day that the submission completes on the target system.

Install Time
The time that the install should take place on the target systems.

If blank, the installation will begin as soon as installation approval is complete on the target system.

TimeZone
Indication of which location the entered Install Date/Time refers to.

Local System – schedule the date/time based on the time zone of the local (sending) system. If the target system is in a different time zone, the scheduled date/time will be adjusted accordingly. So, if the RFP is scheduled to install at 9pm EST because the local system is in New York, it will be installed at 6pm PST if the target system is in Los Angeles.

Target System – schedule the date/time based on the time zone of the target system. The scheduled date/time used on the SBMJOB command will match the value entered exactly. So, if the RFP is sent to 3 different locations and scheduled for 9pm, it will install at 9pm of the time zone of each of those systems.

Send
Select 1 or more Locations to send the RFP to, or click all to select all locations in the list. Once selected, press button Send RFP to Target Systems to submit the send job.

Additionally, for Transmit Type SFO (export Objects to a Save File without MDCMS information), the following fields can be edited for each row:

Name of Temporary Library saved to the Save File
Name of the Save File
Name of the IFS File, if the Save File is copied to IFS

Options
RFP Send History – open the RFP Send History view filtered by the RFP and target location
FTP Log – view the FTP log of the most recent attempt to send via FTP to the selected location for this RFP

19.2 Add RFP to Send Listing

A new package of objects can be added to the RFP Send Listing by right-clicking in the view and selecting option Add. The resulting dialog allows you to create an empty RFP to send, which can be populated manually with objects later, or it can automatically contain objects and commands based on installation history.

Add MDCMS RFPSend Parameters:

Application
The local application code of the application to send from

Level
The local application level of the promotion level to send from

Description
The description to apply to the new RFP.
If filtering to a specific RFP number, the description can be left blank to automatically use the description from the installed RFP. If splitting into multiple RFPs, the description will be applied + a suffix appended stating the part of the RFP, such as DB or non-DB Part 3.

Empty RFP
True – A new send RFP will be created without any objects in it.
False – The new send RFP will be populated with all objects in installation history that match the values of the filter fields.

Minimum/Maximum Install Date
The range of dates for installation history to include in the RFP. A blank minimum means no minimum. A blank maximum means no maximum.

Minimum/Maximum RFP Number
The range of installed RFP numbers to include in the RFP. A blank minimum means no minimum. A blank maximum means no maximum.

Project
Only include objects that were installed for project matching the name pattern in the filter.

Task
Filter the object requests to those that were installed for a specific task number

Subtask
Filter the object requests to those that were installed for a specific subtask number

Object Requester
Only include objects that were requested by a user matching the name pattern in the filter.

Object Library
Only include objects that installed into a library matching the name pattern in the filter.

MDCMS Attribute
Only include objects that were requested for an attribute matching the name pattern in the filter.

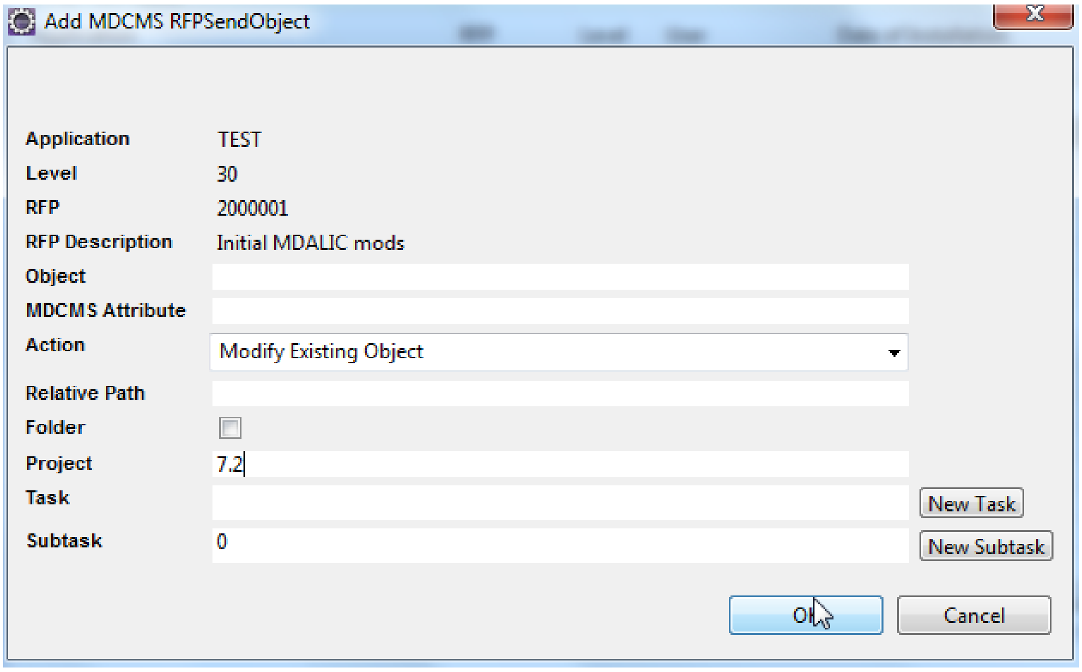
Include RFP Cmd/Scr
True – any commands that were defined for specific installed RFPs, that contain objects included in the send RFP, will be included in the send RFP.
False- no commands defined for specific RFPs will be added to the send RFP

Split DB and Non-DB
If multiple send RFPs should be created. 1 containing any database objects (Physical files, Logical files, data areas and SQL elements) and at least 1 containing non-database objects.

Non-DB Objs per RFP
The maximum number of non-database objects per send RFP.
For example, if the filtered values would mean there are 500 programs, and the number of non-DB Objects per RFP is 100, then there would be 5 RFPs created to contain them (plus 1 for DB objects, if there are any).
Leave this field blank to place all non-DB objects on a single RFP.

19.3 Add Object to Send RFP

When it is necessary to add additional objects to the RFP that will be sent to a target system or logical partition, right click on RFP and select ⊞ New Object Request. That action will bring up the ***Add MDCMS RFPSendObject*** dialog.



Application	TEST
Level	30
RFP	2000001
RFP Description	Initial MDALIC mods
Object	
MDCMS Attribute	
Action	Modify Existing Object
Relative Path	
Folder	☐
Project	7.2
Task	
Subtask	0

Add Object Parameters:

Object	the object name - enter a portion of the name for Content-Assist to list objects starting with that text
MDCMS Attribute	the attribute indicating the type and location of the object to be requested. If left blank, MDCMS will attempt to discern the attribute based on MDXREF and MDCMS history
Action	DDL to decide to Modify, Recompile, Update or Delete the object
Relative Path	The relative path, if requesting an IFS object, that resides in a subfolder of the folder defined for the attribute. Content-Assist may be used multiple times to drill down to the appropriate folder
Folder	Whether or not an IFS object is a folder
Project	the Project, or reason, for the request - enter a portion of the name for Content-Assist to list Projects starting with that text
Task	a Task number within the Project, if necessary
Subtask	a Subtask within the Project Task, if necessary

19.4 Merge Multiple RFPs into One RFP

Any RFPs for the same application level that are in status 00 OR 01 may be merged together. To do so, select each of the rows of RFPs to be merged and then right-click on one of the selected rows and select option Merge.

A confirmation window is then displayed where some of the selected RFPs can be unchecked and the description of the result RFP can be edited. Click Merge to carry out the merge process.

19.5 Send RFP Commands

To view or modify the commands that are defined for the specific RFP, right click on the RFP and select **RFP Commands**. That action will display the ***MdRFPCommandView***. See section RFP Commands for more details.

19.6 Send RFP Scripts

To view or modify the scripts that are defined for the specific RFP, right click on the RFP and select **RFP Scripts**. That action will display the ***MdRFPScriptView***. See section RFP Scripts for more details.

19.7 Objects in Send RFP

To review the details of any object included on the RFP, simply expand the RFP contents using the ▷ arrow to the left of the Application name in the Application column and then left click an object row.

Use right-click for an object row for the following options:

Add – add an object to the RFP

Copy – add an object to the RFP with the parameter values initially filled with the values from the selected row

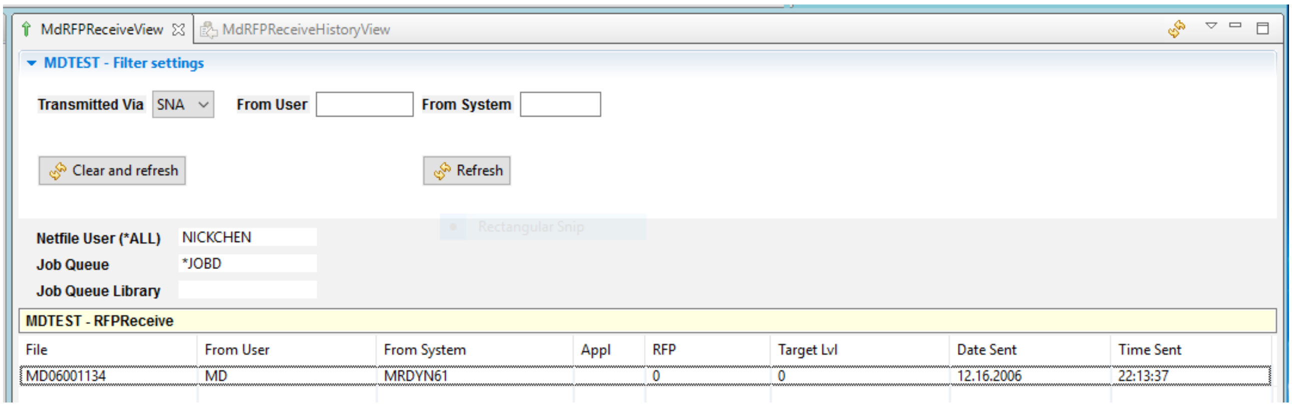
Delete – remove the Object from the RFP to be Sent

Object Commands – view and modify the list of commands that are to be sent with the specific Object. See section Attribute Commands for more information about defining commands.

20 RFP Receive Listing

If the automatic receipt of Promotions is not used (see MDOpen Settings Levels and MDCMS API's MDRCVIFS or MDRCVSNA), then MDCMS promotions sent from remote systems are received manually using ***RFP Receive Listing*** from the repository list.

The ***MdRFPReceiveView*** below will contain any RFP's that are currently open for receipt from other systems or logical partitions.



Transmitted Via: SNA
From User:
From System:

Netfile User (*ALL): NICKCHEN
Job Queue: *JOBD
Job Queue Library:

MDTEST - RFPReceive

File	From User	From System	Appl	RFP	Target Lvl	Date Sent	Time Sent
MD06001134	MD	MRDYN61		0	0	12.16.2006	22:13:37

Promotion packages may be received here via SNA or FTP (or anything else besides SNA). MDCMS remembers which method was used the last time that a promotion was received. It is best to select the Transmitted via (if incorrect) before entering the other information because the screen fields available change based on Transmitted via. The example above had SNA selected so the Netfile User field is available for entry.

To view RFPs that are received (not showing) in the Receive Listing, use repository list option RFP Receive History.

Netfile User

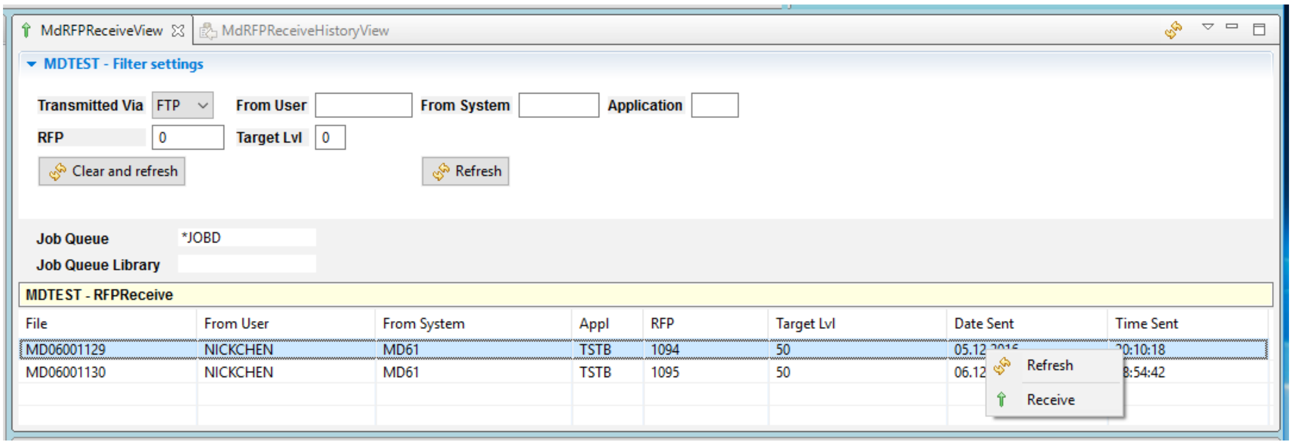
If the promotion package was sent via SNA, and a specific Netfile user was entered (default is QPGMR) then that same user id must be entered here to receive the promotion. Press the refresh button after entering the Netfile User to filter the list of RFPs.

Job Queue/Library

The actual receive job is submitted to batch. The subsystem job queue/library may be specified for the receive job.

20.1 RFP Receive Listing Options

Right clicking on an RFP received entry within the ***RFPReceiveView*** will display the following options:



Transmitted Via: FTP
From User:
From System:
Application:
RFP: 0
Target Lvl: 0

Job Queue: *JOBD
Job Queue Library:

MDTEST - RFPReceive

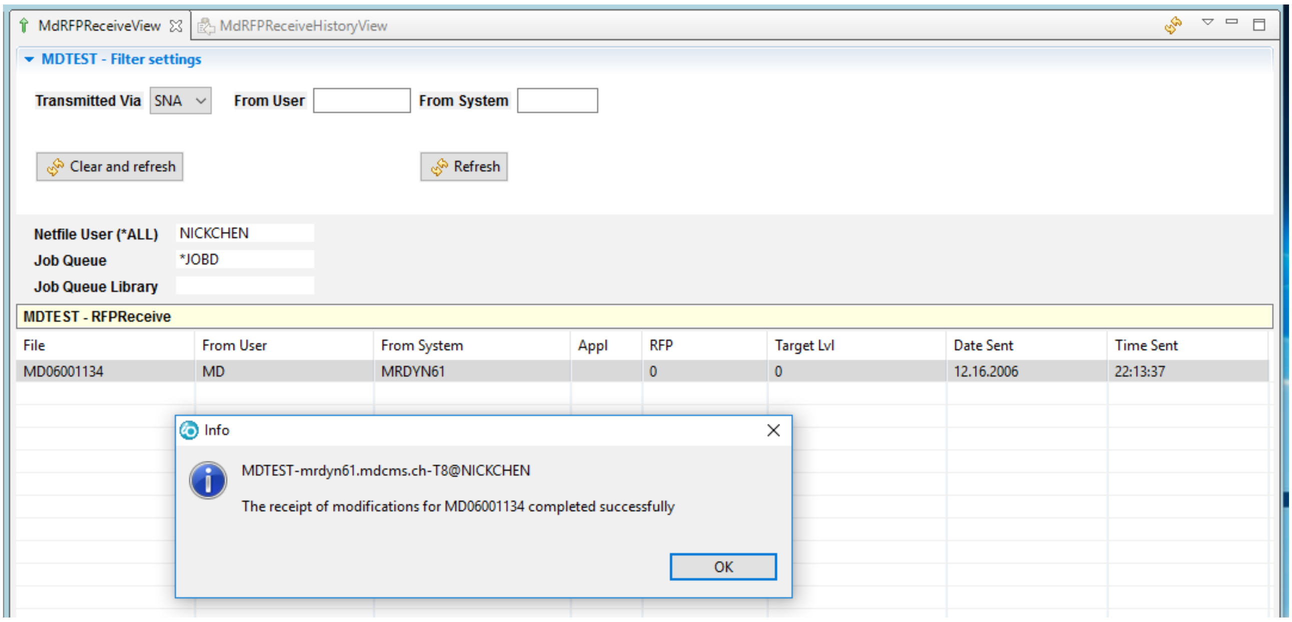
File	From User	From System	Appl	RFP	Target Lvl	Date Sent	Time Sent
MD06001129	NICKCHEN	MD61	TSTB	1094	50	05.12	0:10:18
MD06001130	NICKCHEN	MD61	TSTB	1095	50	06.12	8:54:42

If FTP is selected for Transmitted Via, the filter fields are available above. The Netfile user is not available.

Options:

Refresh	Refreshes the list of RFPs to receive
Receive	Submits a job to Receive the RFP

Choosing the Receive menu option will generate the following success window otherwise an error window will display:



Transmitted Via: SNA
From User:
From System:

Netfile User (*ALL): NICKCHEN
Job Queue: *JOBD
Job Queue Library:

MDTEST - RFPReceive

File	From User	From System	Appl	RFP	Target Lvl	Date Sent	Time Sent
MD06001134	MD	MRDYN61		0	0	12.16.2006	22:13:37

MDTEST-mrdyn61.mdcms.ch-T8@NICKCHEN

The receipt of modifications for MD06001134 completed successfully

The submitted job creates a temporary library with the same name as the file. All source and objects related to the promotion are placed in this temporary library. After the objects are installed, the temporary library is deleted.

New MDCMS request records are written for the source/objects and a new RFP number is generated for the received items. The Promotion will then be ready to be installed into the lowest level for the application, unless a higher target level was specified when the Promotion was sent.

If Auto-Submit for the Promotion Level is set to Y, then the compilation portion of the installation will begin as soon as the receipt has completed.

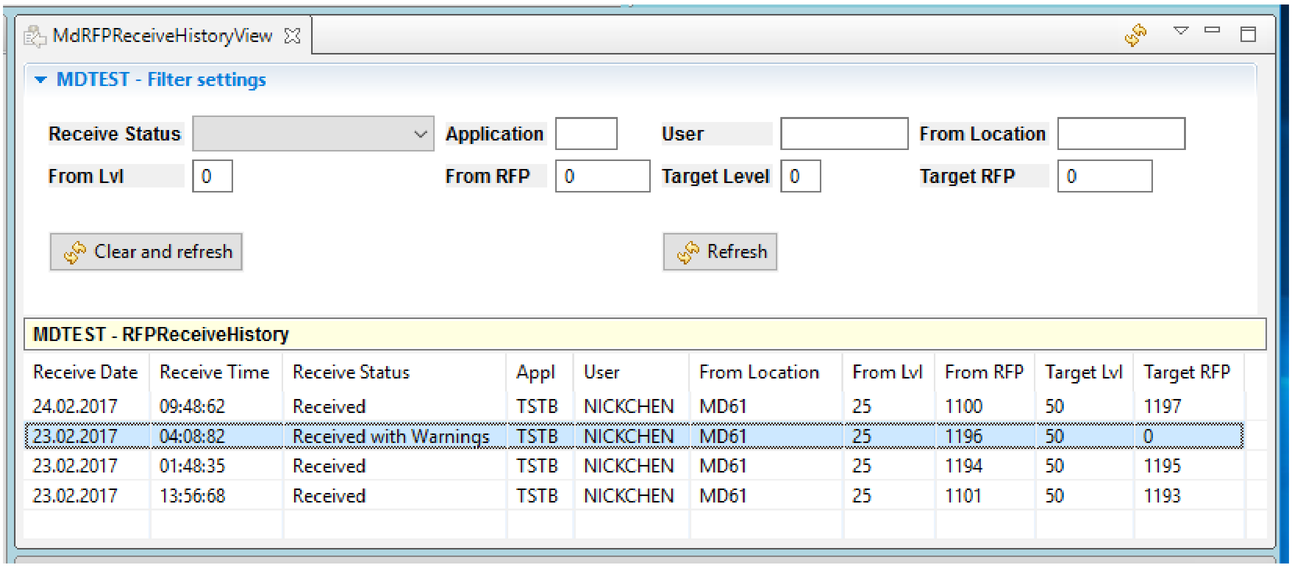
21 RFP Receive History

RFP Receive History can be selected from the repository listing view to view a log of all attempts to automatically or manually receive an RFP onto this system.

For each attempt, any warnings or exceptions can be viewed.

If an RFP was successfully sent, and should have been automatically received, but doesn't appear in the receive history view, then perform DSPMSG QSYSOPR to see the reason why the receive job couldn't be submitted.

The ***MdRFPReceiveHistoryView*** below will contain any RFP's that have been received or attempts to be received from other systems or logical partitions.



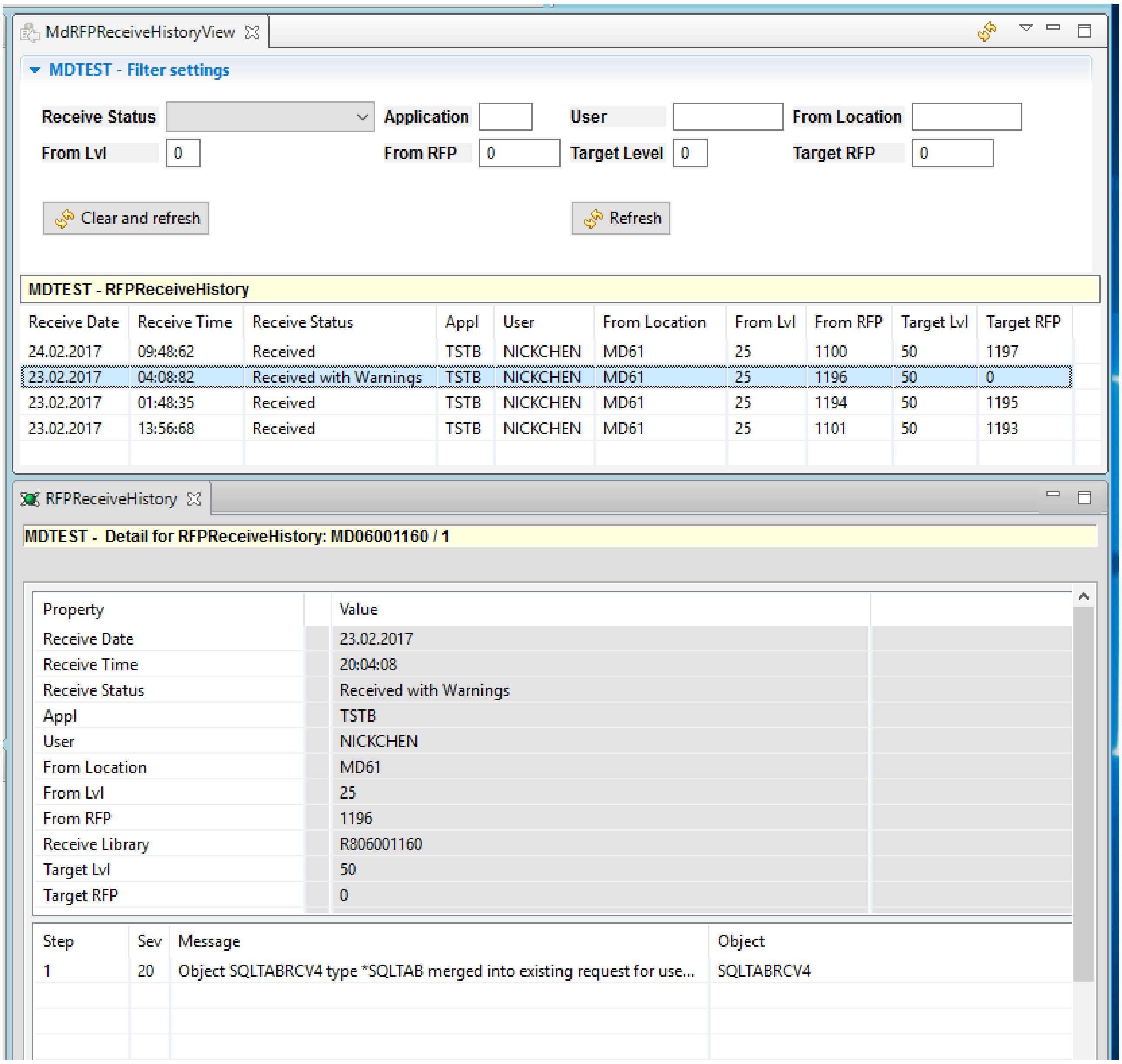
Receive Status		Application		User		From Location	
From Lvl	0	From RFP	0	Target Level	0	Target RFP	0

MDTEST - RFPReceiveHistory

Receive Date	Receive Time	Receive Status	Appl	User	From Location	From Lvl	From RFP	Target Lvl	Target RFP
24.02.2017	09:48:62	Received	TSTB	NICKCHEN	MD61	25	1100	50	1197
23.02.2017	04:08:82	Received with Warnings	TSTB	NICKCHEN	MD61	25	1196	50	0
23.02.2017	01:48:35	Received	TSTB	NICKCHEN	MD61	25	1194	50	1195
23.02.2017	13:56:68	Received	TSTB	NICKCHEN	MD61	25	1101	50	1193

21.1 RFP Receive History Detail

Left clicking on any receive history entry within the view will bring up the ***RFPReceiveHistory*** detail view. The ***RFPReceiveHistory*** detail view will present complete details for the selected RFP received and may also contain errors as shown below if the status is other than Received.



MdRFPReceiveHistoryView

MDTEST - Filter settings

Receive Status		Application		User		From Location	
From Lvl	0	From RFP	0	Target Level	0	Target RFP	0

MDTEST - RFPReceiveHistory

Receive Date	Receive Time	Receive Status	Appl	User	From Location	From Lvl	From RFP	Target Lvl	Target RFP
24.02.2017	09:48:62	Received	TSTB	NICKCHEN	MD61	25	1100	50	1197
23.02.2017	04:08:82	Received with Warnings	TSTB	NICKCHEN	MD61	25	1196	50	0
23.02.2017	01:48:35	Received	TSTB	NICKCHEN	MD61	25	1194	50	1195
23.02.2017	13:56:68	Received	TSTB	NICKCHEN	MD61	25	1101	50	1193

RFPReceiveHistory

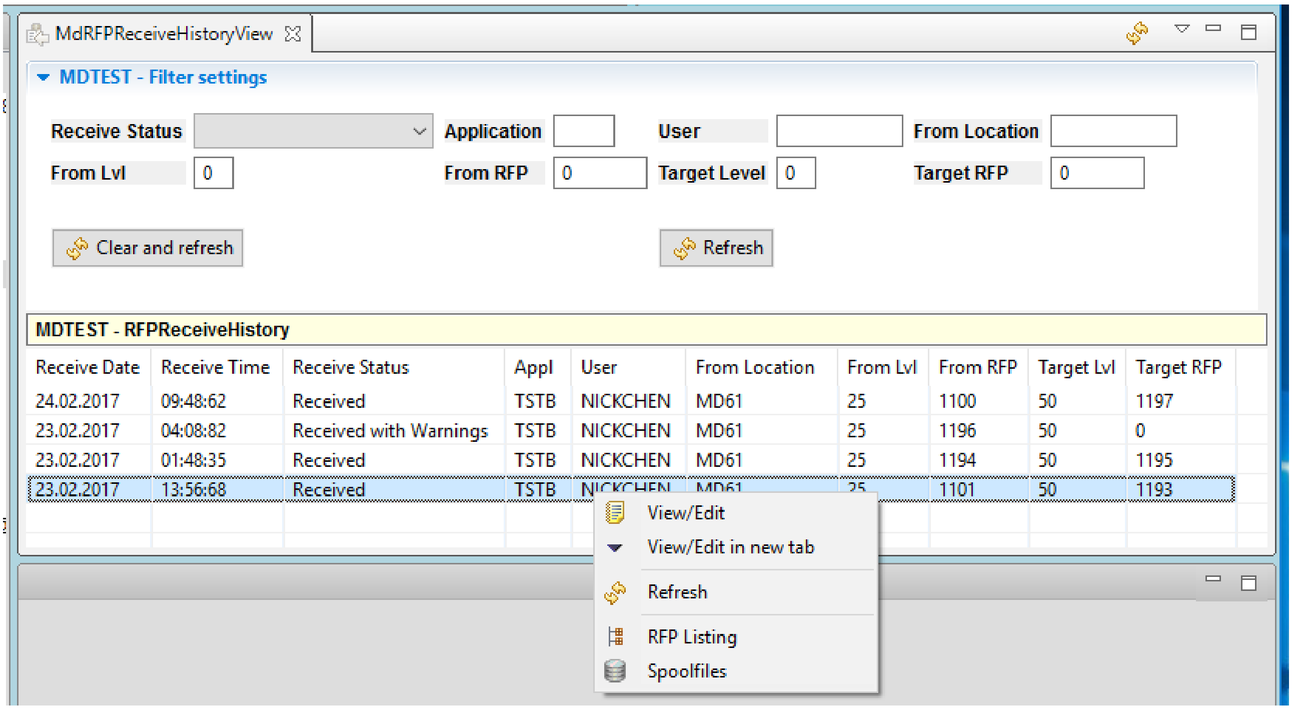
MDTEST - Detail for RFPReceiveHistory: MD06001160 / 1

Property	Value
Receive Date	23.02.2017
Receive Time	20:04:08
Receive Status	Received with Warnings
Appl	TSTB
User	NICKCHEN
From Location	MD61
From Lvl	25
From RFP	1196
Receive Library	R806001160
Target Lvl	50
Target RFP	0

Step	Sev	Message	Object
1	20	Object SQLTABRCV4 type *SQLTAB merged into existing request for use...	SQLTABRCV4

21.2 RFP Receive History Options

Right clicking on an RFP received entry within the ***RFPReceiveHistoryView*** will display the following options:



Receive Status		Application		User		From Location	
From Lvl	0	From RFP	0	Target Level	0	Target RFP	0

Receive Date	Receive Time	Receive Status	Appl	User	From Location	From Lvl	From RFP	Target Lvl	Target RFP
24.02.2017	09:48:62	Received	TSTB	NICKCHEN	MD61	25	1100	50	1197
23.02.2017	04:08:82	Received with Warnings	TSTB	NICKCHEN	MD61	25	1196	50	0
23.02.2017	01:48:35	Received	TSTB	NICKCHEN	MD61	25	1194	50	1195
23.02.2017	13:56:68	Received	TSTB	NICKCHEN	MD61	25	1101	50	1193

Options:

View/Edit	Same as left clicking to display the RFP receive history detail view
Refresh	Refreshes the RFP receive history view
RFP Listing	Brings up the RFP listing for the specific target RFP
Spoolfiles	Brings up any spool files for the receive job. Note: usually exist only on error