Munsys 14

MANAGEMENT CONSOLE USER MANUAL

Munsys® Management Console User Manual

Visit Open Spatial on the internet: http://www.openspatial.com

Contents

Chapter 1
Introducing the Munsys Management Console User Manual

Introduction

The Munsys Management Console is a standalone application that allows users with relevant privileges to manipulate objects and components within a Munsys schema in a structured and user-friendly way.

The Munsys Management Console can be activated by any authorized database user, but the functionality that will be available to a user is dependent on the roles that have been assigned to the user.

- The following components can be managed from the Munsys Management Console:
- The Munsys Schema – creating, dropping, exporting, importing and validating a schema (including importing demo data).
- Applications – used to manage Munsys application settings
- Locks – used to maintain record/object locking
- Query Categories – used to maintain query categories and query privileges
- Security – used to manage users and roles
- Privilege – used to assign roles to tables and views
- Tables/Views – maintenance of Munsys-related spatial tables, lookup tables and indexes
- Lineage – used primarily to implement triggers in the Oracle database to track and archive changes made to records in spatial tables
- Scheduled Tasks – Used to manage and optimize Scheduled Tasks

What's in this manual

The *Munsys Management Console User Manual* consists of the following chapters:

- Chapter 1 – Welcome to the Munsys Management Console User Manual gives an overview of this manual, and provides the typographical conventions used throughout the Munsys documentation set.
- Chapter 2 – Getting acquainted with the Munsys Management Console gives an overview of the concepts of the application.
- Chapter 3 – Managing the Munsys Schema discusses the various schema management options in the Munsys Management console.
- Chapter 4 – Import Demo Data shows the administrator how to add modify or delete Munsys demo data within the Munsys Management Console.
- Chapter 5 – Working With Munsys Applications shows the administrator how to add modify or delete Application settings within the Munsys Management Console.
- Chapter 6 – Record locking describes record locking management in the Munsys Management Console.
- Chapter 7 – Query Categories describes the concept of query categories and query category management in the Munsys Management Console.
- Chapter 8 – Security in the Munsys Management Console describes the creation and maintenance of database users and roles.
- Chapter 9 – Privilege describes the maintenance of all Munsys spatial and lookup tables, as well as indexing.
- Chapter 10 – Working With tables and indexes describes the maintenance of all Munsys spatial, lookup and attribute tables as well as indexing.
- Chapter 11 – Munsys Lineage describes the creation and maintenance of lineage.
- Chapter 12 – Scheduled Tasks describes the creation and maintenance of database scheduled tasks.

Conventions in this manual

The following table lists the typographical conventions used in this manual.

Text element	Example
Keys you press on the keyboard	CTRL, ENTER, DEL
Screen buttons	Click Close.
Folder paths	C:\Program Files\Open Spatial
Menu paths	choose **File** > **Validate Schema**.
Hypertext links to more information	http://www.openspatial.com
Dialog box/screen names	The New Query Category dialog box
Application functions	The Validate Schema function

Table 1 Munsys typographical conventions

Finding the information you need

To get help on

- general issues, select Help from the Munsys Management Console menu bar.
- an operation in progress, click the Help button on the dialog box.
- the latest support options, visit http://www.openspatial.com

Chapter 2
Getting acquainted with the Munsys Management Console

Introduction

This chapter provides an introduction to the Munsys Management Console interface, as well as the way that the Console is structured. You will also be shown how to start the Munsys Management console and connect to the database.

Munsys Management Console structure

The Munsys Management Console consists of a window divided into two panes: the Tree Pane and the Content Pane.

The left pane (the tree pane) shows the main components supported by the MMC.

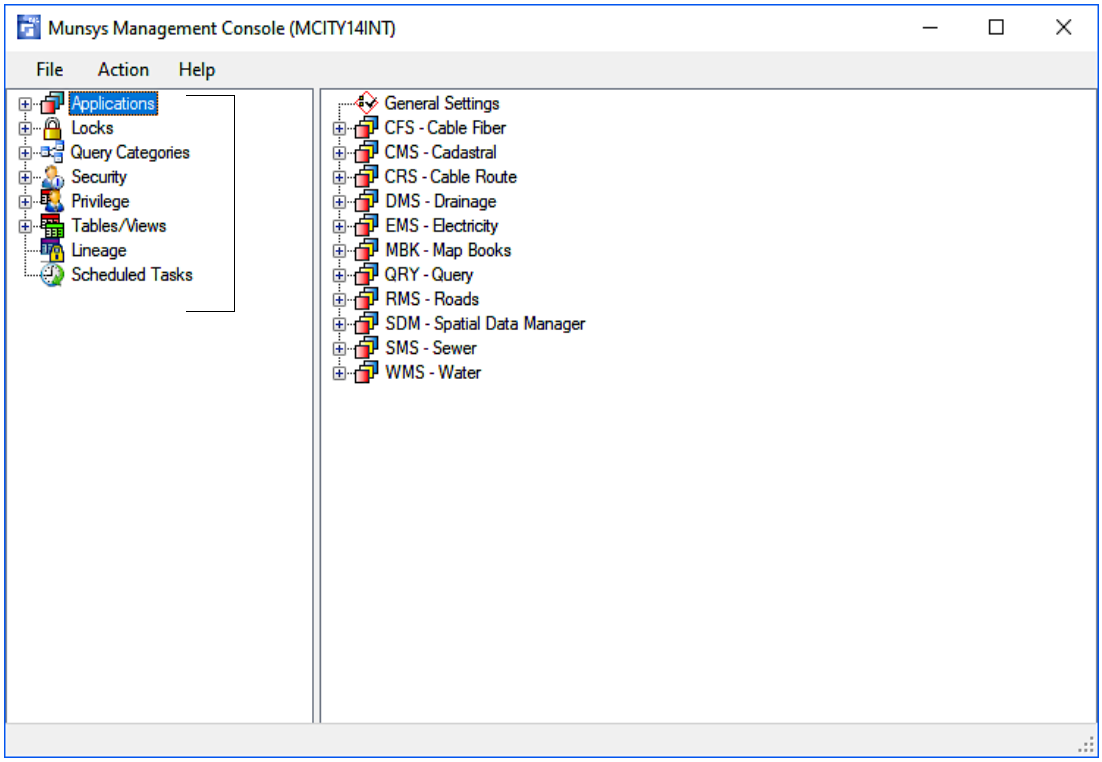


Figure 1 Munsys Management Console: Tree Pane

The Munsys Management Console content pane contains the following components:

- Applications
- Locks
- Query Categories
- Security
- Privilege
- Tables/Views
- Lineage
- Scheduled tasks

The right pane (the content pane) contains the content of the function that has been selected in the tree pane.

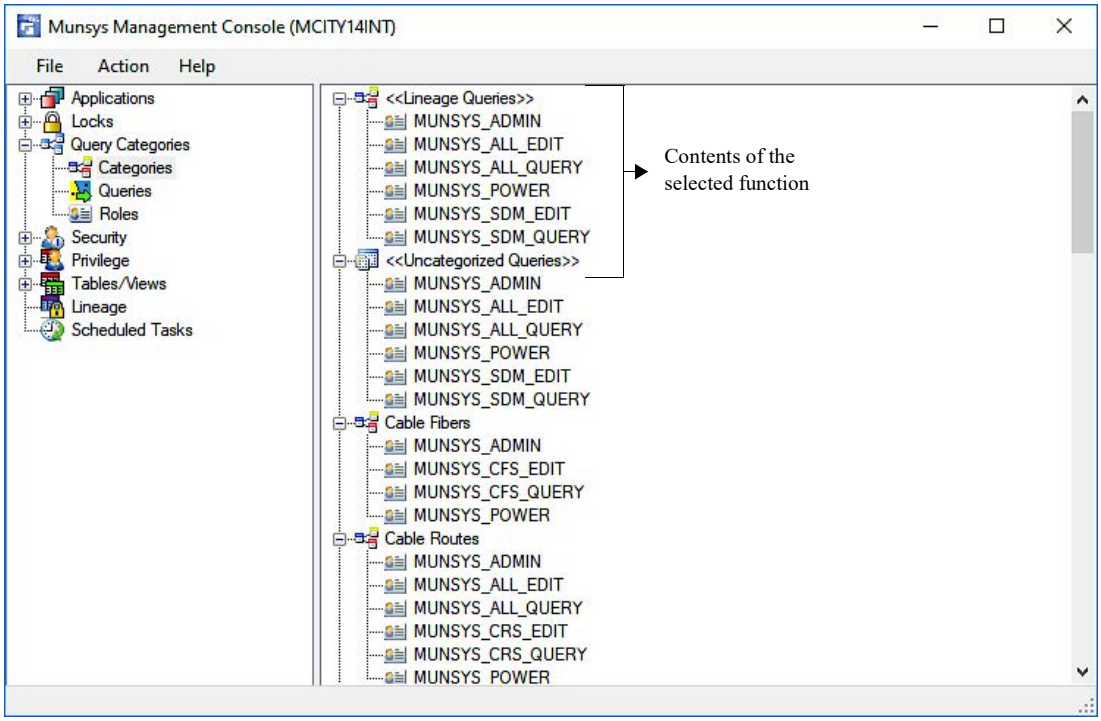


Figure 2 Munsys Management Console: Content Pane

- Both the content pane and the tree pane allow context-sensitive menus to be activated on selected items.

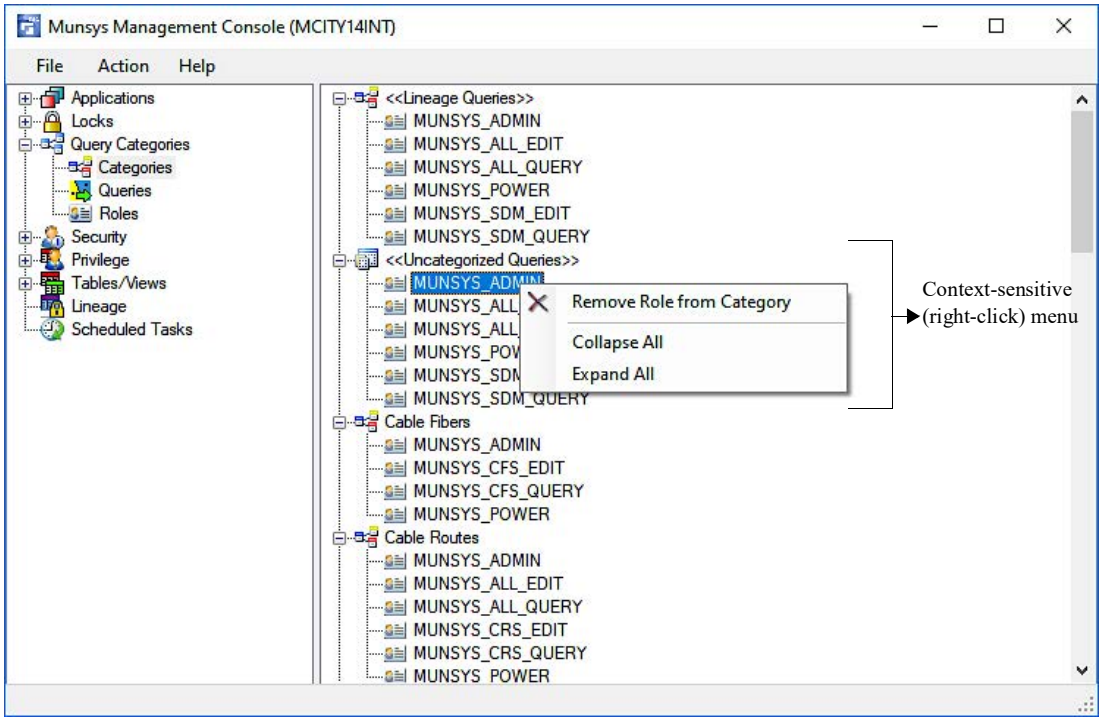


Figure 3 Munsys Management Console: Context-sensitive menu

- The same items that appear on the context-sensitive menu will also appear on the main menu under the Action menu. The Action menu is therefore not static, but varies according to the context-sensitive items related to the function that has been selected.

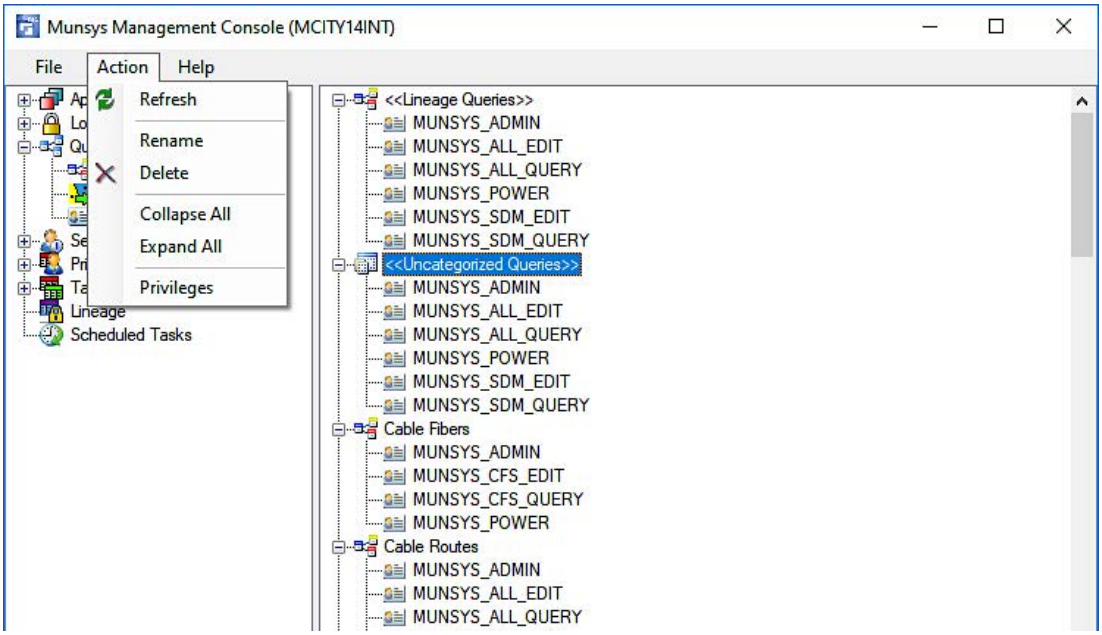


Figure 4 Munsys Management Console: Action menu

- The File menu contains the functionality that is used to maintain and manipulate the database and schema. Some functionality may be disabled depending on your login authentication.

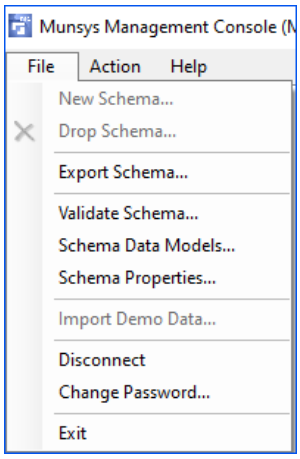


Figure 5 Munsys Management Console: File menu

Logging in to the Munsys Management Console

To start the Munsys Management Console, do the following:

1. Choose **Start > Programs > Open Spatial > Munsys 14.2 > Munsys Management Console 14.2**
2. or Double-click the **Munsys Management Console 14.2** icon on the Windows desktop.

The Connect to Database dialog box is displayed.

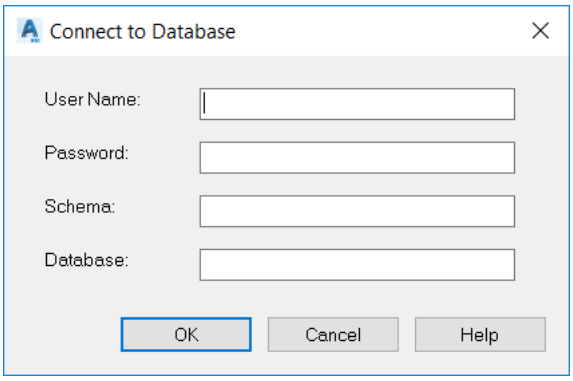


Figure 6 The Connect to Database dialog box

3 Enter the following information:

- User Name and Password (users who are working with an Oracle 11g version database or higher should note that the user name and password are case sensitive)
- Schema
- Database

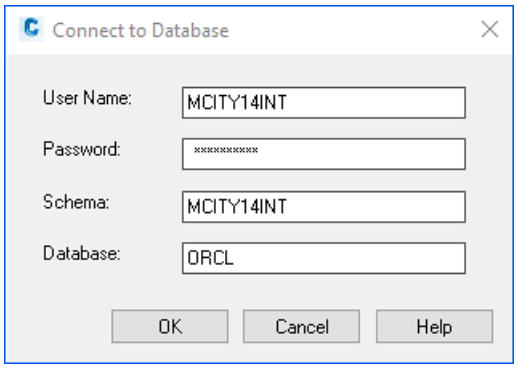


Figure 7 The Connect to Database dialog box

Tip Use the **Disconnect** menu item on the **File** menu to change schema, database and user information if you are already logged in to Munsys Management Console and you want to log in with different credentials.

4 If you log in to the Munsys Management Console without specifying a schema name, only limited functionality will be available on the **File** menu, and none of the other functionality will be available. This is normally the case when you want to create a new schema.

Note The opening position of the Munsys Management Console application window is recorded in the registry settings. The screen position, height and width parameters, as well as the with of the content pane, are applied when opening the application.

Changing your password

1 You can change the password used to connect to the Munsys Management Console with the **File > Change Password** menu item.

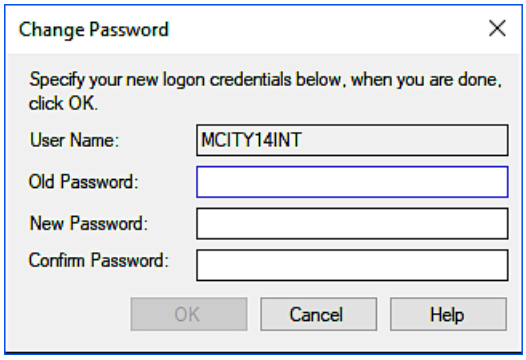


Figure 8 The Change Password dialog box

2 Change the password as required, and then click **OK**.

Chapter 3
Schema management

Introduction

Using the Munsys Management Console, the database administrator can perform the following schema management functions:

- create a new schema
- drop an existing schema
- export a schema
- validate a schema
- install available data models in a schema
- change properties of a schema

Creating a new Munsys Schema

To be able to create a new schema, you will need to have the following roles assigned:

- Munsys role: MUNSYS_ADMIN
- Database privileges: CREATE USER, CREATE TABLESPACE, GRANT ANY ROLE, GRANT ANY PRIVILEGE

Note You will need to log in as **SYSTEM** with **DBA** privileges.

A Munsys schema is created by following five steps:

1. Specifying a schema name and password. The schema name is the owner or default user that is used to connect and manipulate the contents of the schema.
2. Specifying the database extents and coordinate tolerances. The database extents provide an area of coverage for the data that is expected to be captured in the schema. Coordinate tolerances are set to enforce expected data accuracy.
3. Specifying regional information: The locale, units, AutoCAD Map coordinate system and Oracle coordinate system (SRID). Regional information provides the system with information as to which part of the world the data represents. Locality provides information that is used when the data models are created, as various locations have different data expectations. The AutoCAD Map coordinate system code identifies the relevant coordinate projection system to use when referring to the data in the schema in an AutoCAD Map environment. The code represents the coordinate system that is associated with the geometry stored throughout the schema. The Oracle SRID is used when new or modified objects are posted to the database.
4. Installing the relevant data models.
5. Creating the necessary Tablespaces in the database.

To create a new Munsys Schema

1 Choose **File** > **New Schema**...

The Create New Schema – Step 1 of 5 dialog box is displayed:

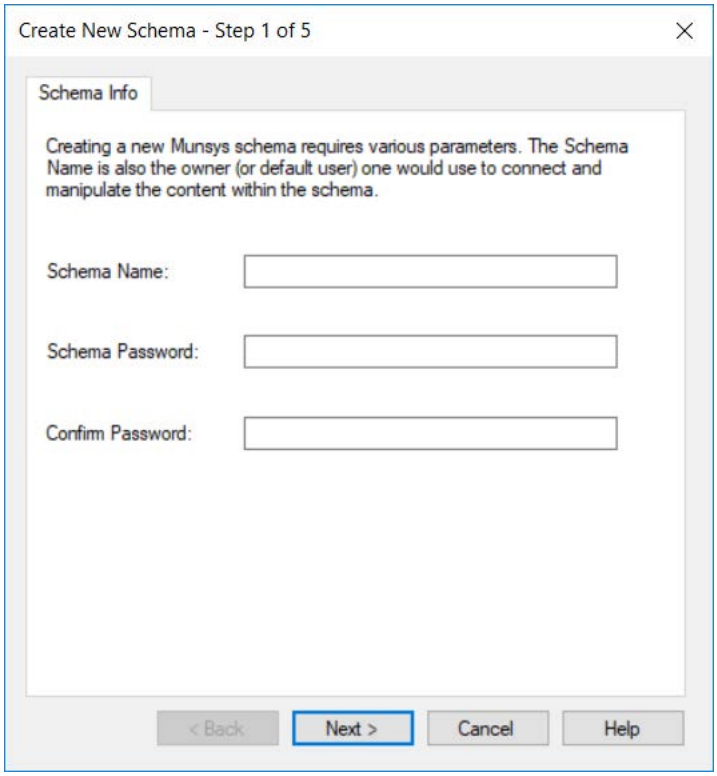


Figure 1 Create new Schema – Step 1 of 5: Schema Name

2 Enter a name for the schema with a maximum of 30 (thirty) characters.

3 Enter and confirm a password for the schema. The maximum length for the password is 30 (thirty) characters.

4 Click **Next** to continue to the **second** step of the schema creation process. (The Next button remains unavailable until all the required fields have been completed.)

In the second step of the schema creation process, the database extents and tolerances are specified:

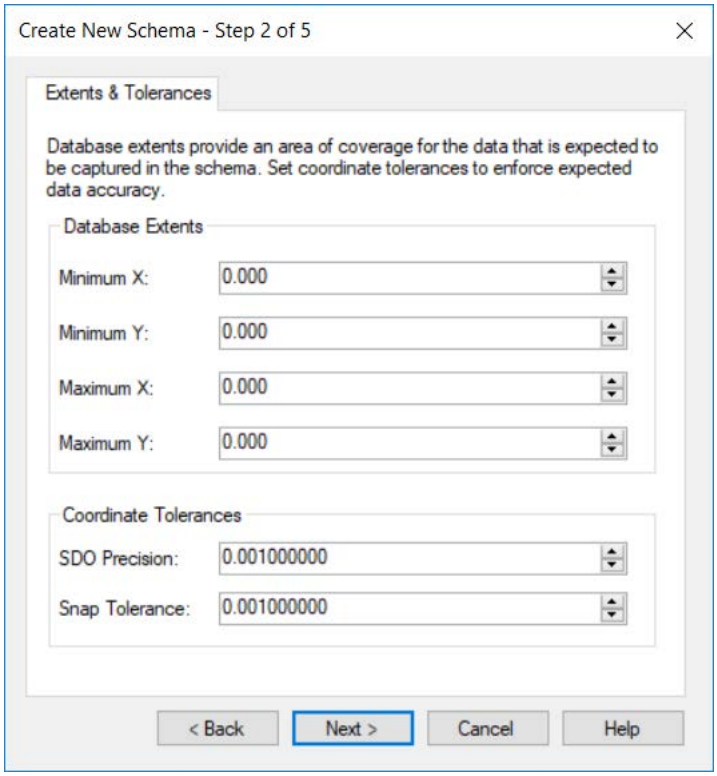


Figure 2 Create New Schema – Step 2 of 5: Extents and Tolerances

5 In the **Database Extents** group, specify the minimum and maximum X and Y values.

6 Next, specify the **SDO Precision** and **Snap Tolerance** values.

7 Click **Next** to continue to the **third** step of the schema creation process.

In the third step of the schema creation process, the regional info for the schema is specified. The regional info provides the system with information about which part of the world the data represents. The locality part of the regional info is used when the data models are created, as different locations have different data expectations.

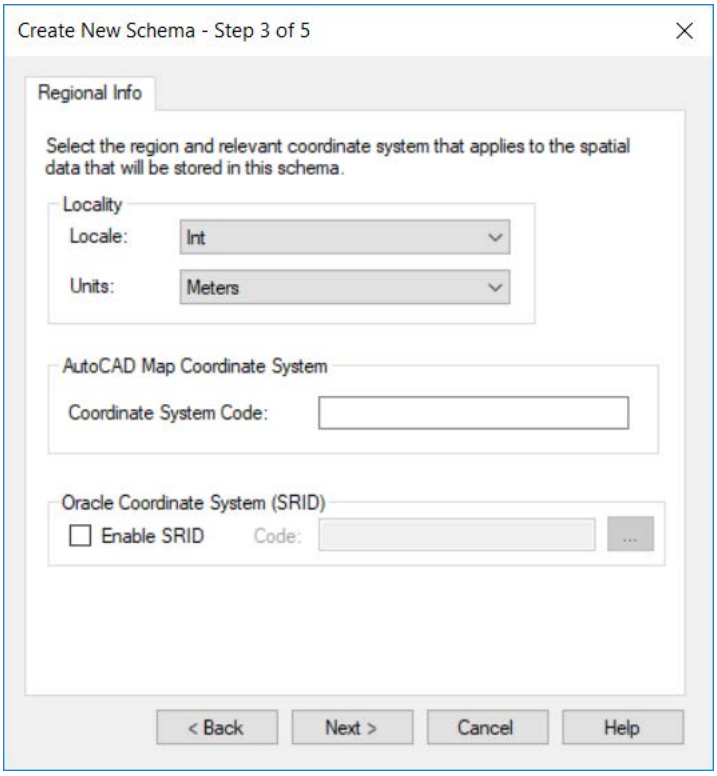


Figure 3 Create New Schema – Step 3 of 5: Regional Information

8 In the **Locality** group, select the **Locale** and **Units**. The default values are **United States** and **Feet**.

9 Next, enter the **AutoCAD Map Coordinate System Code**. This is the code that identifies the relevant coordinate projection system to use when referring to the data in the schema in an AutoCAD Map environment. This code represents which coordinate system is associated with the geometry stored throughout the schema. The coordinate system is especially important in an environment where multiple projections systems are used, so that if necessary data may be transformed between various systems.

10 Select the **Enable SRID** check box if you want to you want to see the existing Oracle SRIDs in the database, and then enter the appropriate code. Click the button next to the **Code** field to display the existing SRIDs in the Oracle database.

Note The Oracle SRID is not currently used by the Munsys system; however, other systems require an SRID stored as part of the geometry. The SRID value is used when posting new objects or updating others to the database from the AutoCAD environment, where every geometry record includes the SRID as one of the elements. The SRID is obtained from the Oracle list.

11 Click **Next** to continue to the **fourth** step of the schema creation process, in which the data models for the new schema are specified:

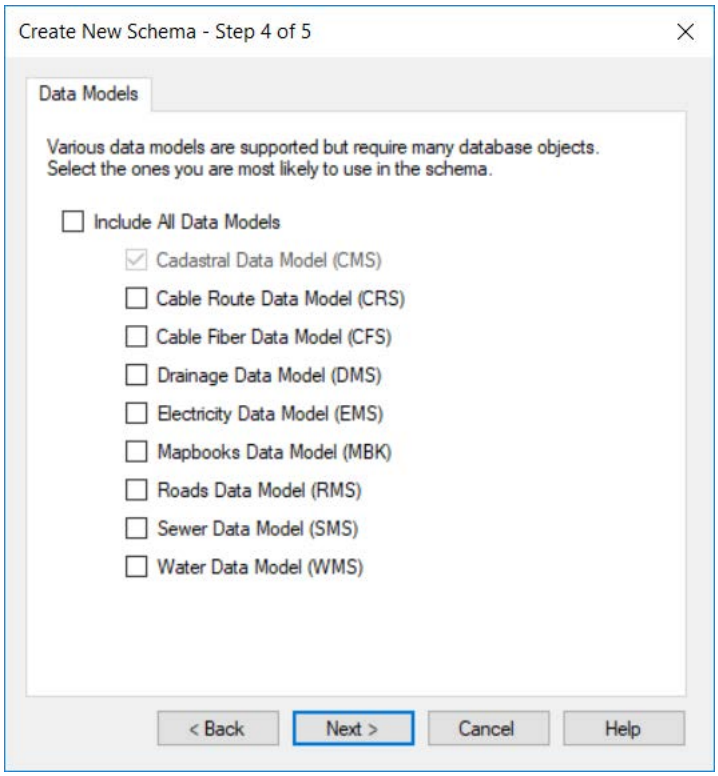


Figure 4 Create New Schema – Step 4 of 5: Data Models

12 Select the data models that you want to install. To install all the available data models, select the **Include All Data Models** option. The **Cadastral** data model is installed by default.

13 Click **Next** to continue to the **fifth** and last step of the schema creation process, in which the necessary tablespaces for the new schema are created:

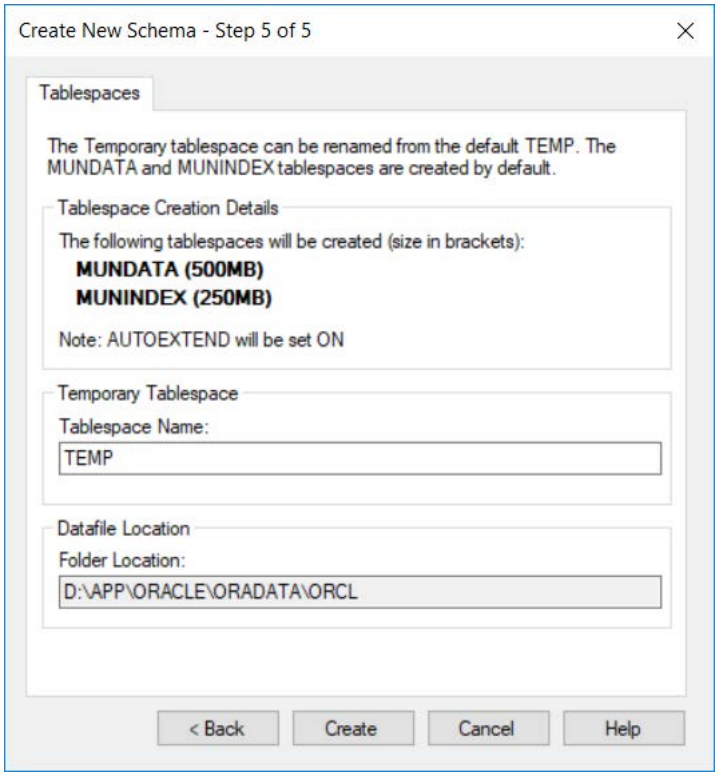


Figure 5 Create New Schema – Step 5 of 5: Data Models

The Tablespace Creation Details information area shows the tablespaces that will be created. The respective tablespace sizes are displayed next to the tablespace names.

14 Enter a name for the temporary tablespace (the default temporary tablespace name is TEMP).

15 The **Datafile Location** field displays the location of the data files that will be used by the schema. The location cannot be changed.

16 Click **Create** to create the new schema.

The Schema Create Progress dialog box is displayed.

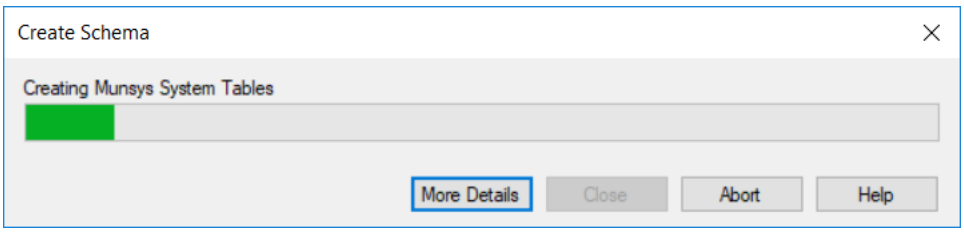


Figure 6 The Schema Create Progress dialog box

Tip Click the **More Details** button to expand the dialog box and display details of the schema creation process. Click the **Abort** button if you want to abort the schema creation process at any stage. If the schema creation process is aborted before it has been completed, the items created up to the point of the abort will be left in the schema.

When the schema has been created, the following dialog box is displayed:

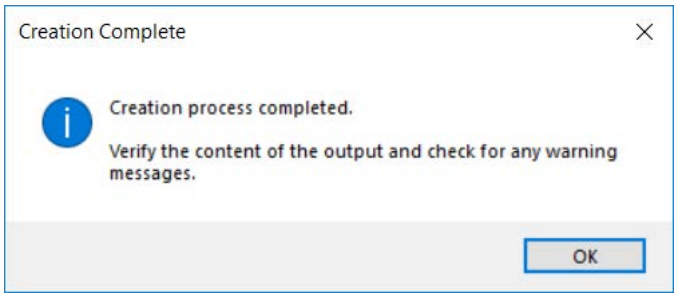


Figure 7 Schema Creation Complete

Dropping an existing schema

In order to drop a schema from the database, you will need to have the MUNSYS_ADMIN role assigned, as well as the DROP USER and DROP TABLESPACE database privileges. A list of all the schemas in the database is displayed on the Drop Schema dialog box, from where you can select the schema that you want to drop. If you are logged on as a schema owner, the name of that schema will not be displayed in the list of schemas that can be dropped, as an owner cannot drop their own schema.

To drop a schema from the database, do the following:

1 Choose **File** > **Drop Schema**.

The Drop Schema dialog box is displayed.

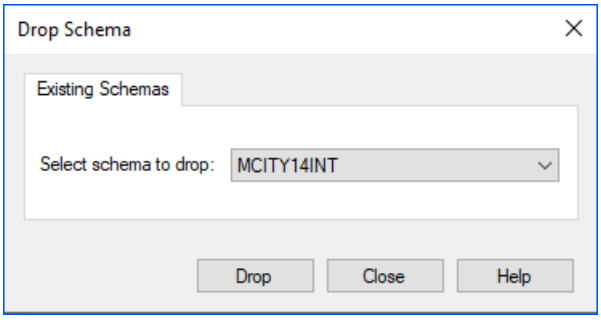


Figure 8 The Drop Schema dialog box

2 Select the schema that you want to drop, and then click the **Drop** button.

The following message is displayed:

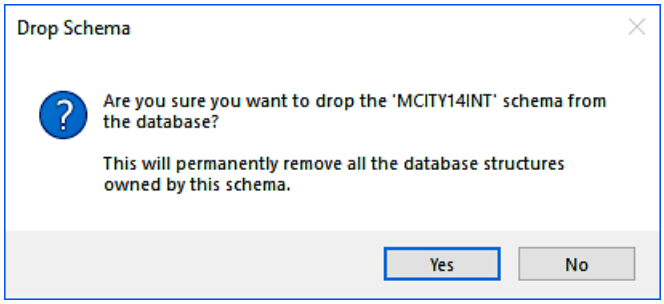


Figure 9 Drop Schema confirmation message

3 Click **Yes** to drop the schema from the database.

The following confirmation message is displayed when the schema has been dropped successfully:

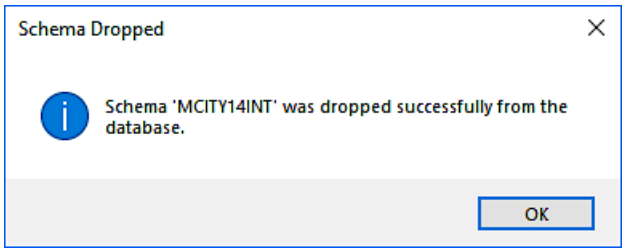


Figure 10 Schema Dropped

Exporting a schema

This function is used to export the contents of a schema to a dump file. To be able to export a schema, you will need to have the MUNSYS_ADMIN role assigned, as well as the EXP_FULL_DATABASE database privilege.

To export a schema, do the following:

1. Choose **File** > **Export Schema**...
2. The **Export Schema** dialog box is displayed.

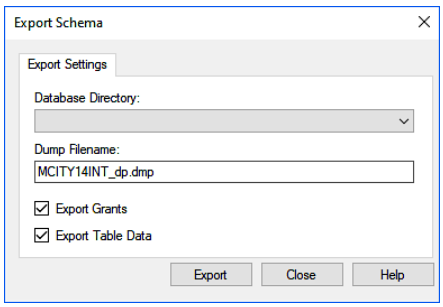


Figure 11 The Export Schema dialog box

3. Enter a name for the dump file that you are going to export the schema to, and the click **Browse...** to specify the destination folder for the file.

   The Dump File field is populated with the destination information of the schema.
4. Select or clear the following check boxes to specify export options:
   - **Export Grants** – select this option to include the grants when exporting the schema
   - **Export Table Data** – select this option to include the table contents when exporting the schema
   - **Compress Extents** – select this option to compress the data in the schema when it is exported
5. Click the **Export** button to export the schema to the file that you specified.

The Export Schema: Progress Log dialog box is displayed, showing details of the export process.

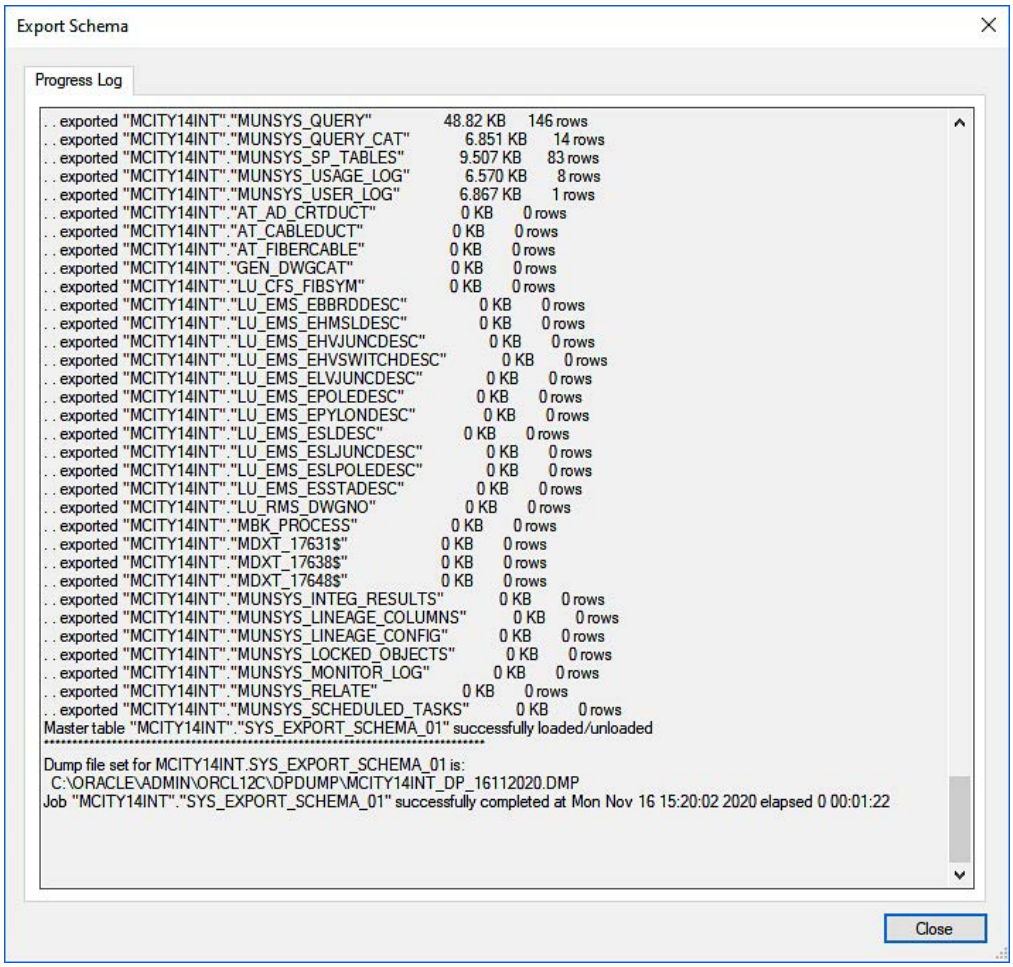


Figure 12 Export Schema: Progress Log

6 Click **Close** when the schema export has been completed successfully.

Validating a schema

The schema validation function validates the Munsys system model tables against rules defined to ensure the integrity of Munsys system tables. You need to have the **MUNSYS_ADMIN** role assigned and be logged in as the schema owner to be able to validate a schema. The results of the validation are shown on the Validation Results dialog box.

To validate a schema, do the following:

1 Choose **File** > **Validate Schema**...

The Validation Results dialog box is displayed. This dialog box consists of the following three tabs:

- **Summary** – this tab lists the main category of the tests that were performed, and whether the entire category failed or passed the schema validation test:

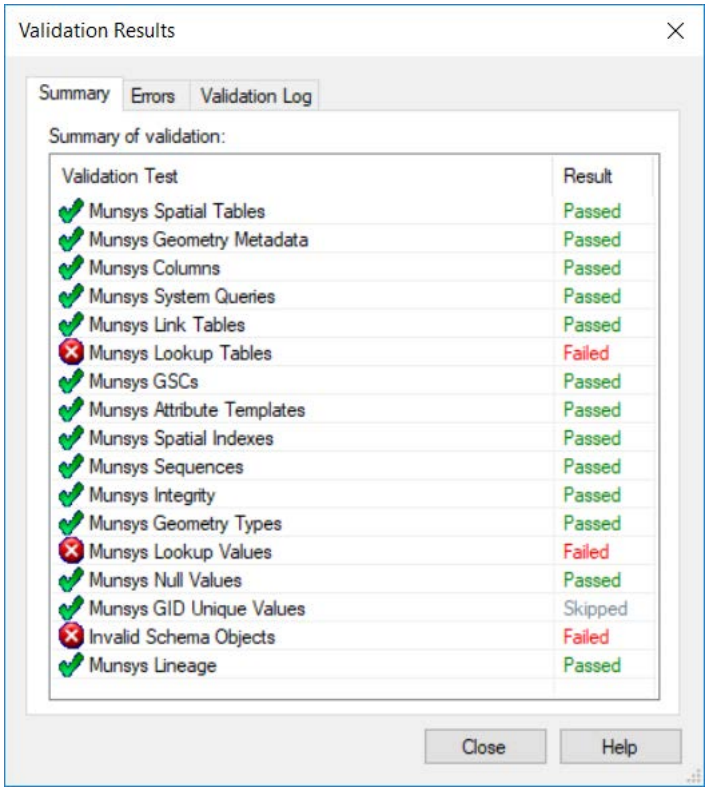


Figure 13 Validation Results dialog box: Summary tab

- **Errors** – this tab contains a list of all the errors that were encountered during the schema validation. The results are grouped under the validation category that was performed, as seen in the following example:

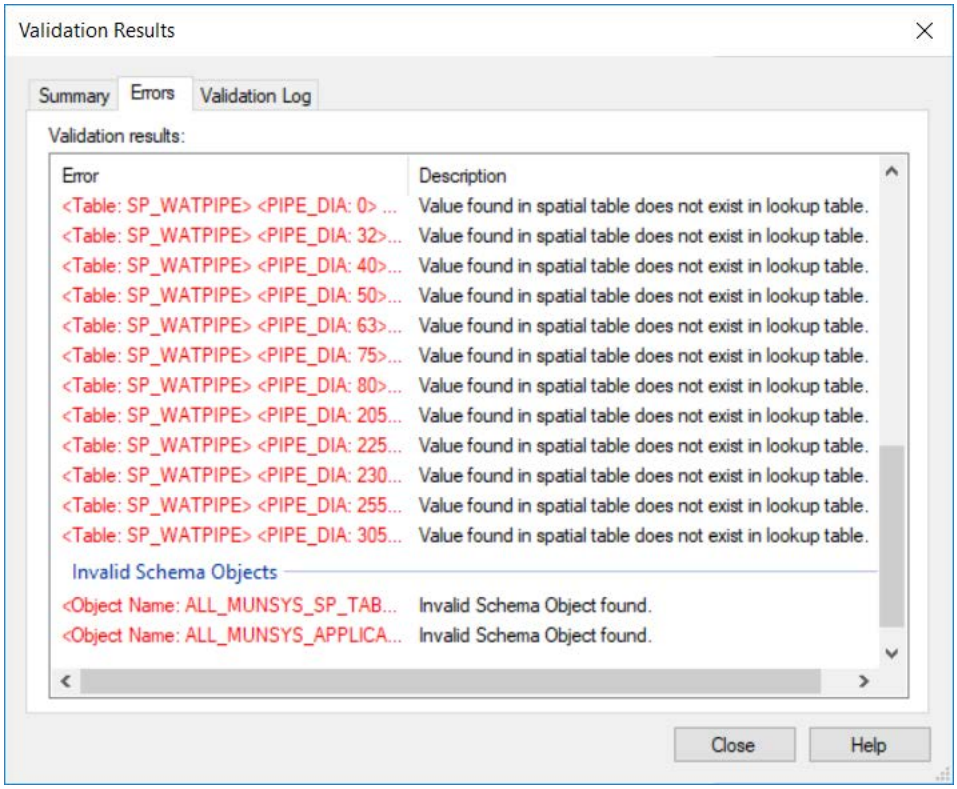


Figure 14 Validation Results dialog box: Errors tab

- **Validation Log** – this tab contains a list of the tests that were performed, as well as a detailed account of the outcome. The validation log can be saved as a text file.

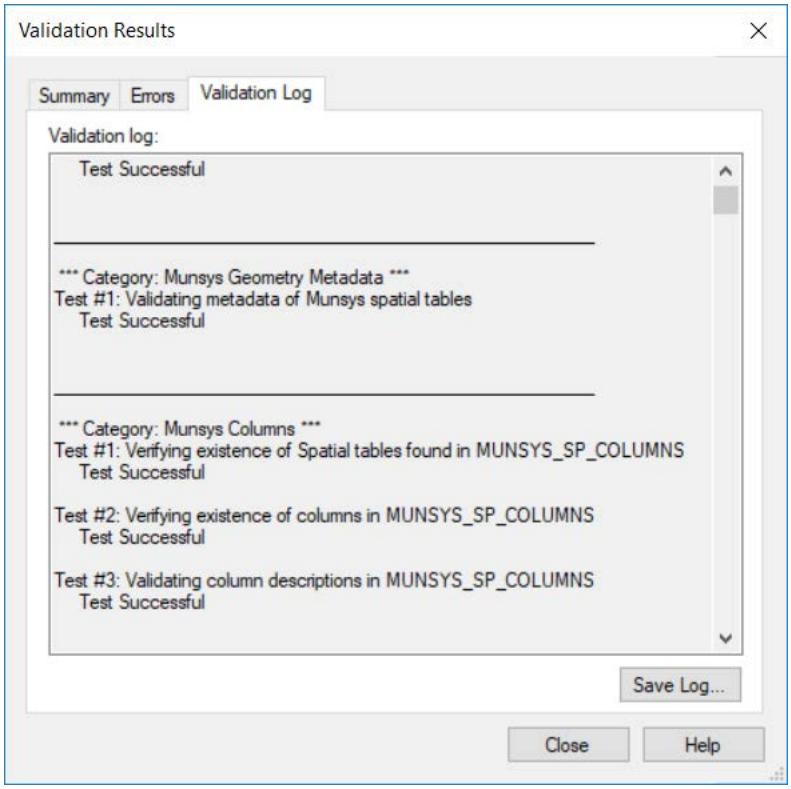


Figure 15 Validation Results dialog box: Validation Log tab

2 Click the **Save Log...** button to save the validation results as a text file, and then click **Close** to exit the **Validation Results** dialog box.

Installing additional data models in a schema

With this function, you can install additional data models in a Munsys schema. You will need the following roles/privileges assigned to be able to execute this function:

- Munsys: MUNSYS_ADMIN
- Database Privileges: CREATE ANY TABLE, ANY VIEW, ANY SEQUENCE and CREATE TABLE, CREATE VIEW and CREATE SEQUENCE

To install additional data models, do the following:

1 Choose **File** > **Schema Data Models**...

The **Data Models** dialog box is displayed, showing a list of the data models that are available for installation.

2 Select the data models that you want to install, and then click **Install**.

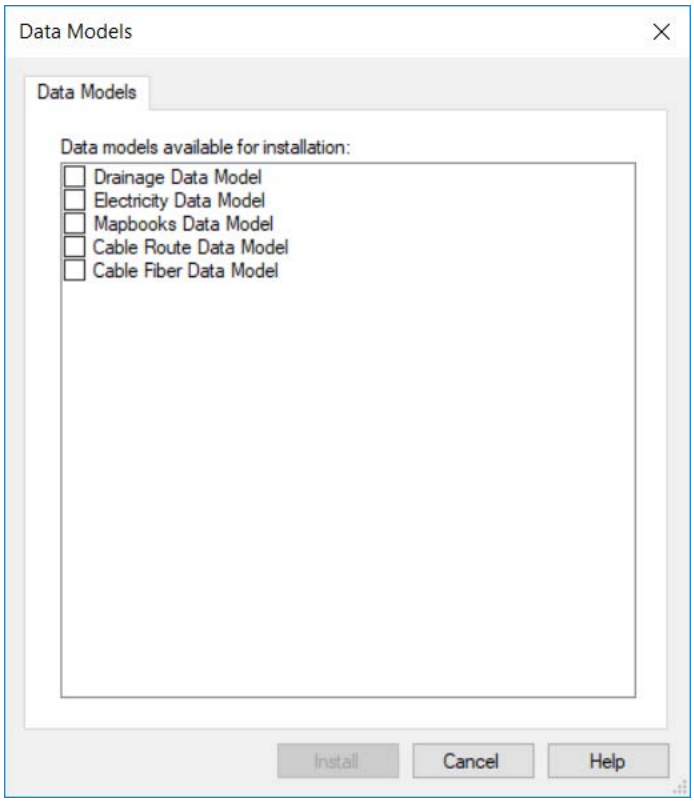


Figure 16 The Data Models dialog box

The Schema Create Progress dialog box is displayed, and the schema is updated with the new data models.

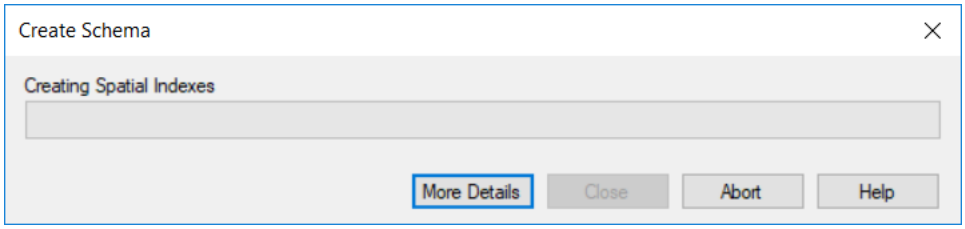


Figure 17 Importing Data Models dialog box

You can do the following on this dialog box:

- Click the **More Details** button to expand the dialog box and show details of the data model installation process.
- Click **Abort** to abort the installation of the data models if required.

  Once the data models have been installed successfully, the following message is displayed:

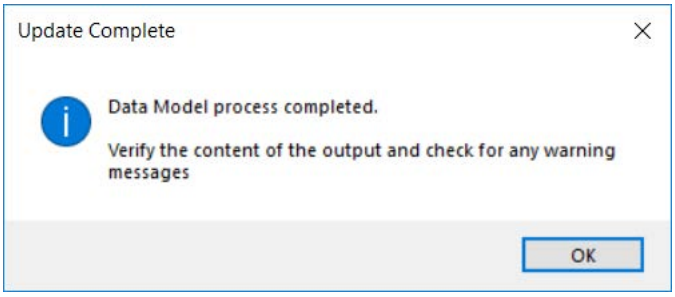


Figure 18 Update Complete: data models installed

Changing the properties of a Munsys schema

With this function, you can change basic properties of a Munsys schema, which are generally stored in the **MUNSYS_DB_SETTINGS** table. You will need the **MUNSYS_ADMIN** role assigned to be able to change schema properties.

To change the properties of a schema, do the following:

1 Choose **File** > **Schema Properties**...

2 The **Schema Properties** dialog box is displayed. This dialog box consists of three tabs: **General**, **Schema** and **Lineage**.

The General Tab

1 On the **General** tab, you can change database settings such as the **Snap Tolerance**, **SDO Precision** and the **Database Coordinate System**. Once changes have been made, the **OK** button becomes available.

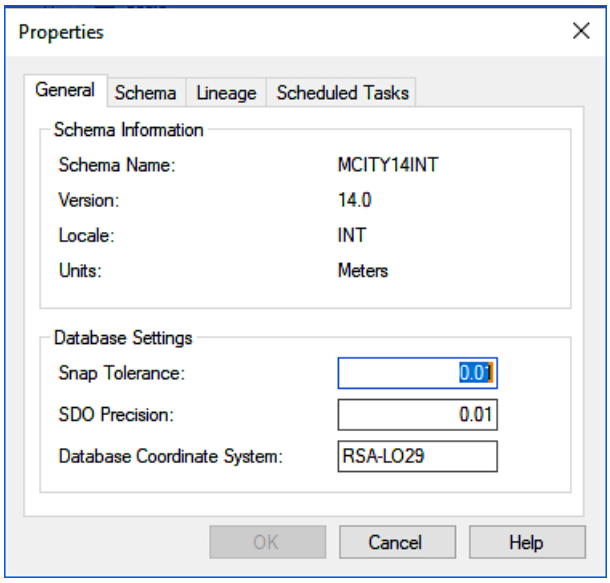


Figure 19 Schema Properties: General

The Schema Tab

1 On the **Schema** tab, you can change the **Database Extents** and the **Oracle SRID**.

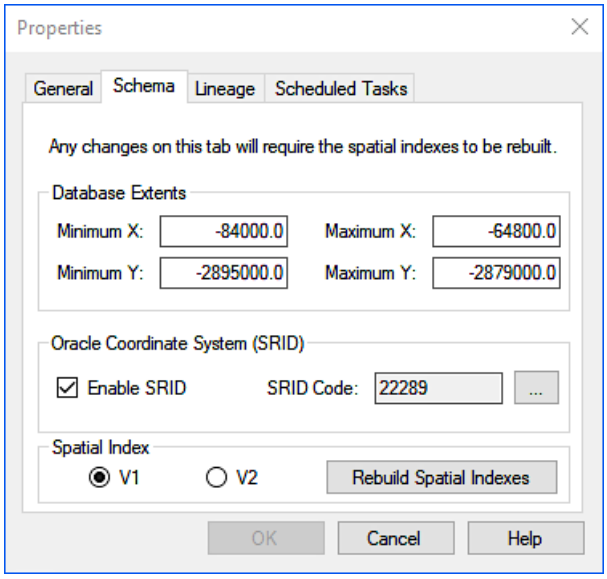


Figure 20 Schema Properties: Schema

Note Changes that are made on the **Schema** tab (the database extents and the Oracle coordinate system) have a direct effect on spatial indexing, and therefore the indexes will have to be rebuilt if you make any changes on this tab. The **OK** button remains unavailable until you have rebuilt the spatial indexes.

Oracle has made significant improvements to the partitioning methods which allows for enhanced performances and allows for spatial indexes to be system-managed.

Spatial Indexes V1 and V2

The main benefit of using V2 Spatial Indexing is because of the simplified spatial index management. Effective with Oracle Release 12.2 and all subsequent versions, spatial indexes can be system-managed by specifying INDEXTYPE=MDSYS.SPATIAL_INDEX_V2 at index creation. You are strongly encouraged to use the V2 index type for all new spatial indexes you create, regardless of whether the spatial table or the spatial index is partitioned, and you may also want to use it if you decide to re-create legacy spatial indexes.

The old INDEXTYPE=MDSYS.SPATIAL_INDEX (V1) is still available for use, It may provide slightly better index creation performance, especially with small data sets and no partitioning involved. You might also want to use the old type if you need to drop a legacy spatial index and then want to re-create it in exactly the same form as it was before.

We recommend that you consult the Oracle documentation before deciding which method of spatial indexing is more suited to your requirements.

https://docs.oracle.com/en/database/oracle/oracle-database/18/spatl/indexing-querying-spatial-data.html#GUID-6BBF58C4-10D0-4993-8DF2-60C3157412D7

Rebuilding Spatial Indexes after updating Database Extents

1 If you have only made changes to any of the Database Extents X and Y parameters on the Schema tab, you are required to **Rebuild Spatial Indexes**, however, the SRID will not be updated.

The following confirmation message is displayed:

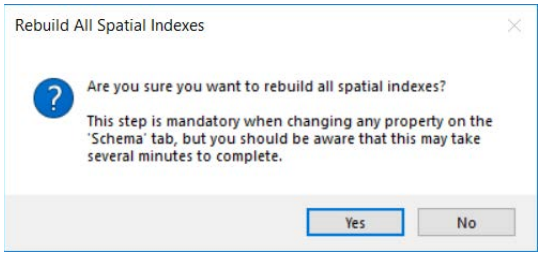


Figure 21 Rebuild All Spatial Indexes

2 Click **Yes** to rebuild the spatial indexes.

The Spatial Index Rebuild Progress dialog box is displayed.

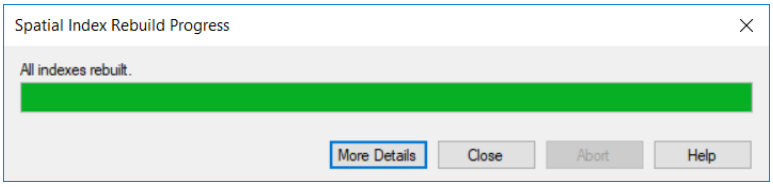


Figure 22 Spatial Index Rebuild Progress

3 Select the **More Details** button to display more information.

Note the SRID update was skipped.

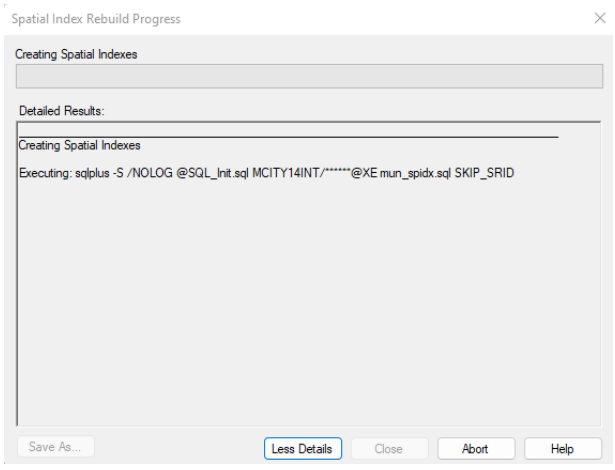


Figure 23 More Details when Rebuilding Spatial Index

4 The buttons on the dialog box allow you to save the results as a text file, display details of the index rebuilding process or abort the process.

When the indexes have been rebuilt successfully a confirmation message is displayed:

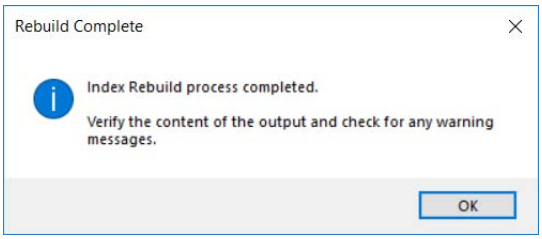


Figure 24 Rebuild complete

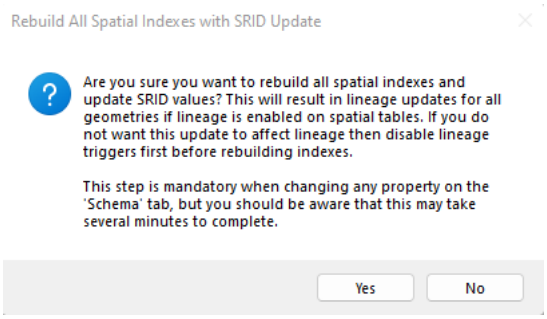
5 Click **OK** to return to the **Properties** dialog box.

Once you have made all the necessary changes to the schema, click **OK**.

Rebuilding Spatial Indexes after updating SRID

1 If you made changes to the **Use SRID** check box, or updated the **SRID Code**, you are also required to **Rebuild Spatial Indexes**, and will result in Lineage updates for all geometries if Lineage is enabled. You will need to disable the Lineage Triggers if you do not want the Lineage updates for all geometries.

The following confirmation message is displayed:



2 Click **Yes** to rebuild the spatial indexes.

The Spatial Index Rebuild Progress dialog box is displayed.

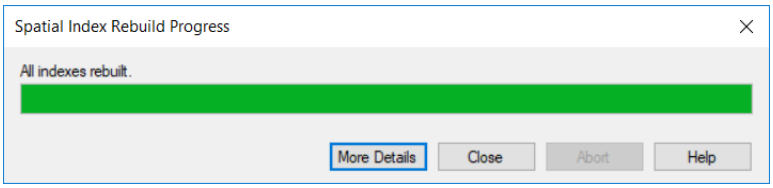


Figure 25 Spatial Index Rebuild Progress

3 Select the **More Details** button to display more information.

Note the SRID update was run.

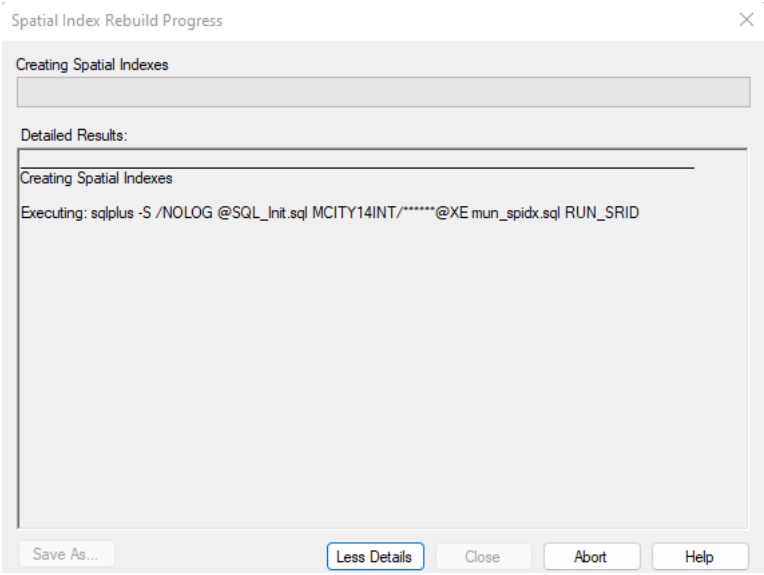


Figure 26 More Details when Rebuilding Spatial Index

4 The buttons on the dialog box allow you to save the results as a text file, display details of the index rebuilding process or abort the process.

When the indexes have been rebuilt successfully a confirmation message is displayed:

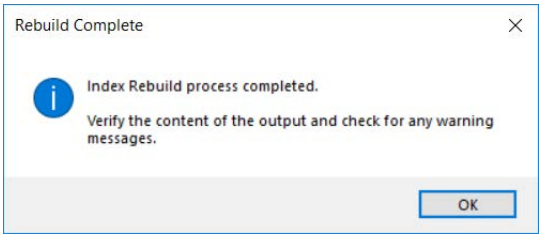


Figure 27 Rebuild complete

5 Click **OK** to return to the **Properties** dialog box.

Once you have made all the necessary changes to the schema, click **OK**.

The Lineage Tab

1 On the **Lineage** tab, you can enable/ disable Lineage functionality and select Lineage editing behavior for the logged in schema.

Filling out the **Lineage** tab in the schema **Properties** dialog box is the is the first essential step in setting up Munsys Lineage.

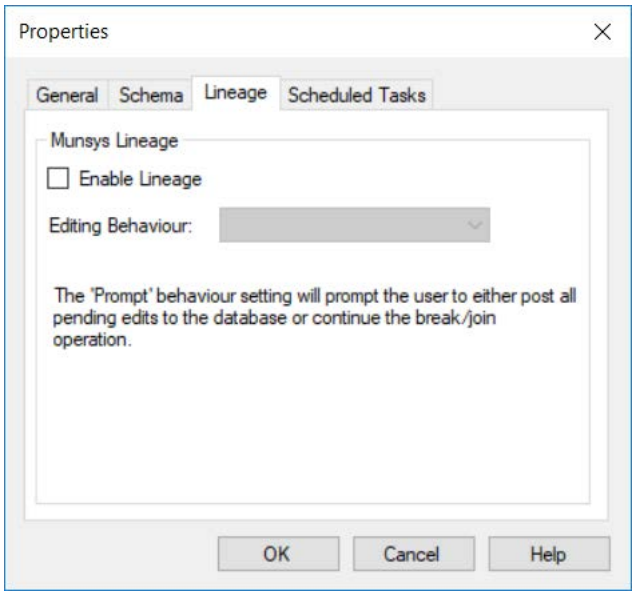


Figure 28 Schema Properties: Lineage

2 Select **Enable Lineage**, and click **OK**.

3 Select the **Editing Behavior** of Lineage

Editing behavior affects how Munsys Lineage will respond to multiple break or join operations on line objects. In the available drop down, the following options can be selected:

- **Continue Editing:** This setting will continue the break/join operation without posting any pending edits to the database. Note that, if more than one split is made and changes are not posted to the database in between splits, the splits will all be linked to the same Lineage parent ID of the line that was broken.
- **Automatic Posting:** This setting will post all pending edits to the database and then continue the break/join operation.
- **Prompt:** This setting will prompt the user to either **Post** all pending edits to the database or **Continue** the break/join operation. Note that, if the same line object is broken more than once and changes ARE posted to the database in between breaks, each line segment will be linked to the Lineage parent of the last line that was broken.

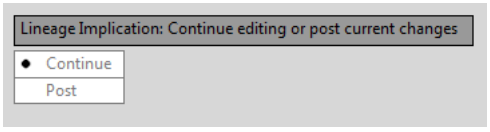


Figure 29 Lineage Implication prompt message displayed in Munsys Applications.

4 Click **OK**, or continue making changes in the schema **Properties** dialog box.

5 To continue setting up Lineage refer to the chapter **Munsys Lineage.**

The Scheduled Tasks Tab

1 On the **Scheduled Tasks** tab, you can enable/ disable the Scheduled Tasks functionality and select Scheduled Tasks process time to run Scheduled Tasks on the schema.

Filling out the **Scheduled Task** tab in the schema **Properties** dialog box is the is the first essential step in setting up Munsys Scheduled Tasks.

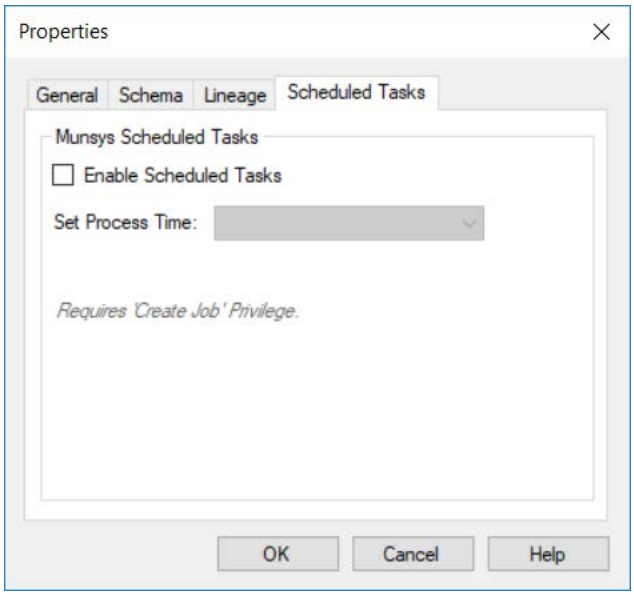


Figure 30 Schema Properties: Scheduled Tasks

Note You will need the **'Create Job'** Privilege to enable Munsys **Scheduled Tasks**.

2 Select **Enable Scheduled Tasks**.

3 Select the preferred Process Time when you want the Scheduled Task to run, and click **OK**.

4 To continue setting up Scheduled Tasks refer to the chapter **Munsys Scheduled Tasks.**

Chapter 4
Import Demo Data

Introduction

The demo data included in Munsys contains more than 14000 land parcels with associated data complementary to each application. The data contained within the demo data has been specifically captured to represent typical practical requirements. New users can make use of the demo data for training and to familiarise themselves with the Munsys applications. Existing users can experiment with new features and enhancements. The demo data will assist in testing/training or presenting Munsys to users and clients without actually having to capture data. Be aware that the demo data contains errors for testing purposes.

The Munsys Demo Data is installed along with the Munsys Management Console. There are two demo schemas for the United Sates (MCITY14US) and International (MCITY14INT) locales.

The MCITY14 schema has spatial and attribute data that can be used for training purposes to familiarise the users with the concepts of Munsys, or it can be used to test functionality before running on the live database.

Munsys Demo Data contains data for all the Munsys applications. Once the database administrator has imported the demo data, users will be able to retrieve the data from the database.

Importing Munsys Demo Data

Be aware that the demo data schema will automatically be created in the import demo data process. To recreate an existing demo data schema, the schema will first need to be dropped before the import can begin.

1 Open the **Munsys Management Console**

2 Complete the 'Connect to the Database' dialog that appears.

The database administrator must login as **SYSTEM** with **DBA** privileges and specify the database to which the demo data will be imported. Leave the **Schema** field blank. Click **OK**.

3 On the menu, select **File > Import Demo Data.**

The Import Demo Data dialog box will appear.

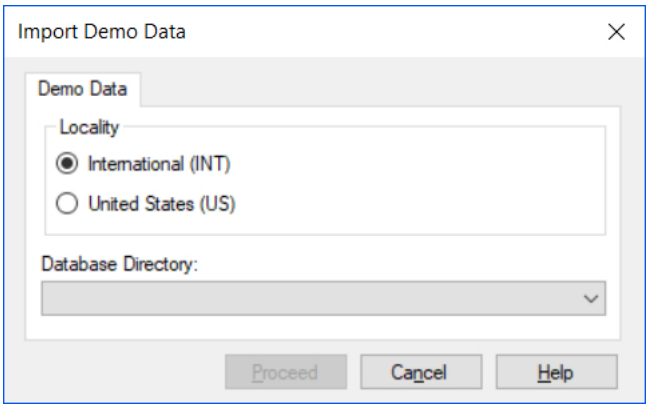


Figure 1 Import Demo Data dialog box

4 In the **Import Demo Data** dialog box, select the appropriate **Locality: International (INT)** or **United States (US)**.

5 Select an **Database Directory** to import the Demo Data.

- By default oracle should have a directory already called **DATA_PUMP_DIR**. To discover where this directory is located on the server, you will need to run the following sql statement on the database that will be hosting the Munsys demo data:

SELECT DIRECTORY_NAME,DIRECTORY_PATH FROM DBA_DIRECTORIES WHERE DIRECTORY_NAME='DATA_PUMP_DIR';

- To create a new directory you will need to run an sql statement similar to the one below:

CREATE DIRECTORY "NAME OFTHE DATA DIRECTORY" AS '/usr/apps/data-files';

This path must exist and be accessible on the Oracle server. You will need to copy the data pump files located on the Munsys install directory:
(E.g. Copy the files **MCITY14INT_DP.dmp** and **MCITY14US_DP.dmp** from C:\Program Files\Open Spatial\MunConsole14.2\DumpFiles) to the directory path on the Oracle server.

- If you log on to the Oracle database with a user other than the one that will be used to create the Munsys schema you will need to grant permissions to the directory using an sql statement similar to the one below:

GRANT READ,WRITE ON DIRECTORY "NAME OF THE DATA DIRECTORY" TO "THE USER NAME";

- To see the new directory, **repeat steps 1-4** and you should see the new directory in the 'Database Directory' drop down of the **Import Demo Data** dialog.

6 Click **Proceed**.

7 The **Import Demo Data Progress** dialog will appear:

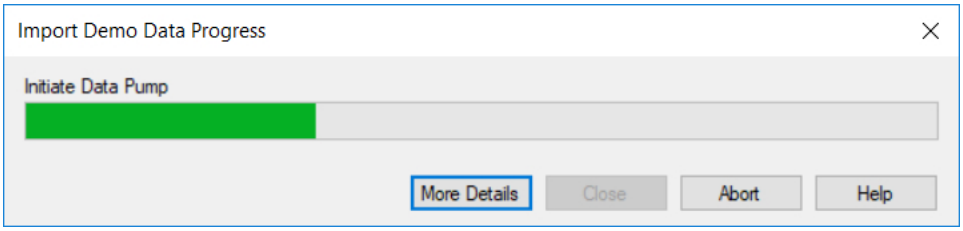


Figure 2 Import Demo Data Progress

The required tablespaces and demo data schema are created. Munsys imports the Oracle dump file that populates the schema with the demo data.

8 When the Import Demo Data Progress is done a confirmation message will be displayed. Click **OK** to go back to the **Import Demo Data Progress** dialog.

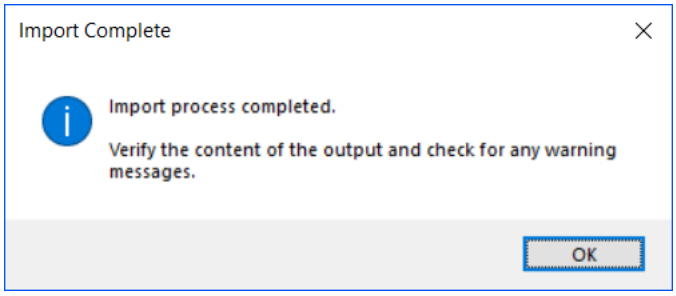


Figure 3 Import Complete

When the import process is completed, verify the content of the output and check for any warning messages in the More Details tab of the **Import Demo Data Progress** dialog.

Log on defaults

During the import process of the demo data, the user will be notified of the schema name that will be created. This information will be required to log on to Munsys Applications and Munsys Management Console. The user logs on to Munsys using the following default log on detail:

US locale

Username	MCITY14US
Password	MCITY14US
Schema	MCITY14US
Database	Database name configured with Oracle client

International locale

Username	MCITY14INT
Password	MCITY14INT
Schema	MCITY14INT
Database	Database name configured with Oracle client

Once a schema is created for the demo data, the database administrator should change the password from its default value. The database administrator should use Munsys Management Console to create users and grant the required roles.

Note Passwords are case sensitive when connecting to an Oracle database.

Chapter 5
Working with Munsys Application Settings

Introduction

Each Munsys application has its own default capture settings that are set by the database administrator in the Munsys Management Console. In the Munsys Management Console, the Applications tree contains a list of all the available Munsys applications and their respective application settings.

This chapter provides information about application setting management in the Munsys Management Console. The following functions can be performed from the Applications tree:

- Creating a new Application Setting
- Modifying an existing Application Setting
- Deleting an Application Setting
- Adding a new Integrity Setting
- Modifying an new Integrity Setting
- Deleting an existing Integrity Setting
- Creating a new Network Check Notification
- Editing an existing Network Check Notification
- Deleting an existing Network Check Notification
- Creating a new Network Node Integrity
- Editing an existing Network Node Integrity
- Deleting an existing Network Node Integrity
- Creating a new Object Attribute Integrity
- Editing an existing Object Attribute Integrity
- Deleting an existing Object Attribute Integrity

The Applications tree: overview

The Applications main tree consists of list of all the Munsys application with their respective application settings.

Using the Action menu or context-sensitive (right-click) menu that is activated when an item is selected, settings for the various applications are created and maintained.

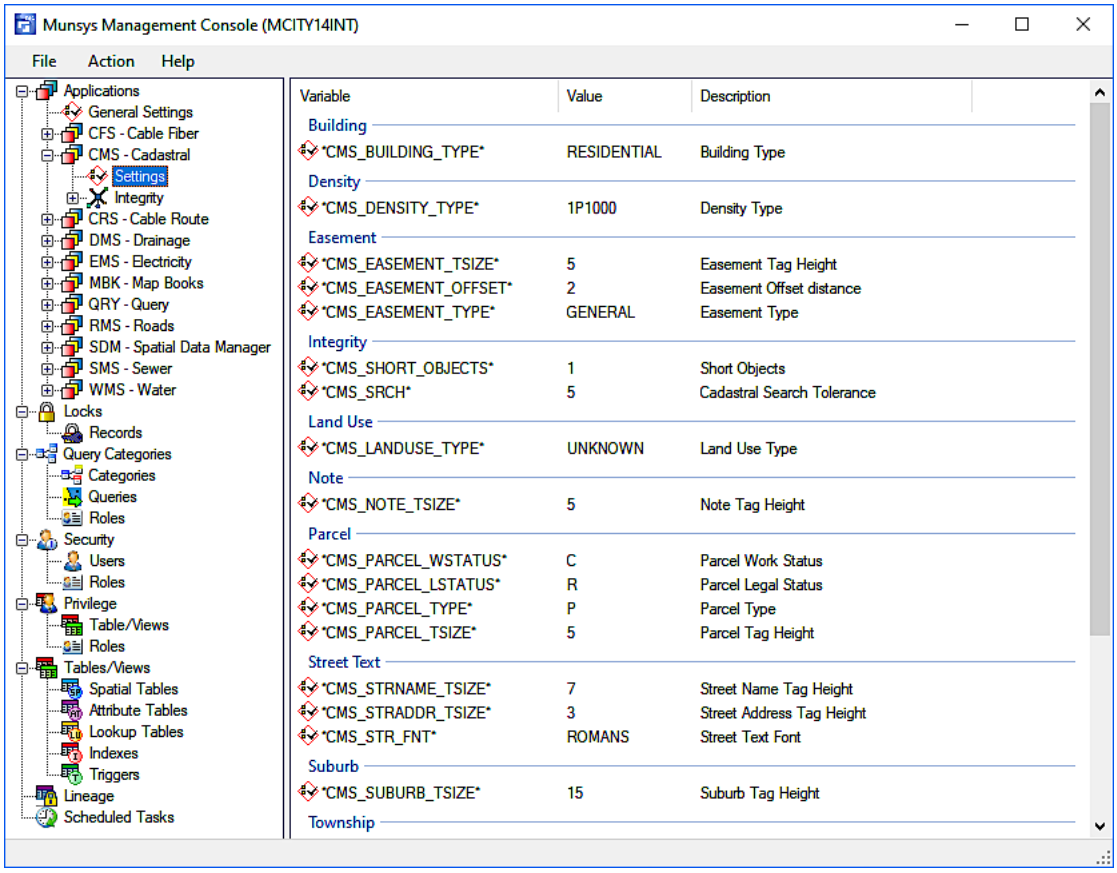


Figure 1 Munsys Management Console: Applications tree

Creating a new application setting

Each Munsys application has its own default settings that are used when data is captured, for example the snap tolerance, tag and symbol scale, node types, etc. Application settings are stored in the MUNSYS_AP_SETTINGS table. When a new setting is created, it inherits the category from the current application and a new record is inserted in the MUNSYS_AP_SETTINGS table. You will need the MUNSYS_POWER or MUNSYS_ADMIN roles assigned to be able to create a new application setting.

To create a new application setting, do the following:

1 Select the **Settings** item below the application that you want to create the setting for in the **Applications** tree, and then select the **New Setting...** option on the **Action** menu or the **context-sensitive (right-click)** menu.

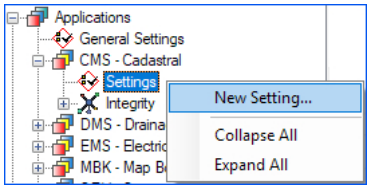


Figure 2 Selecting Application Settings

The New Application Setting dialog box is displayed.

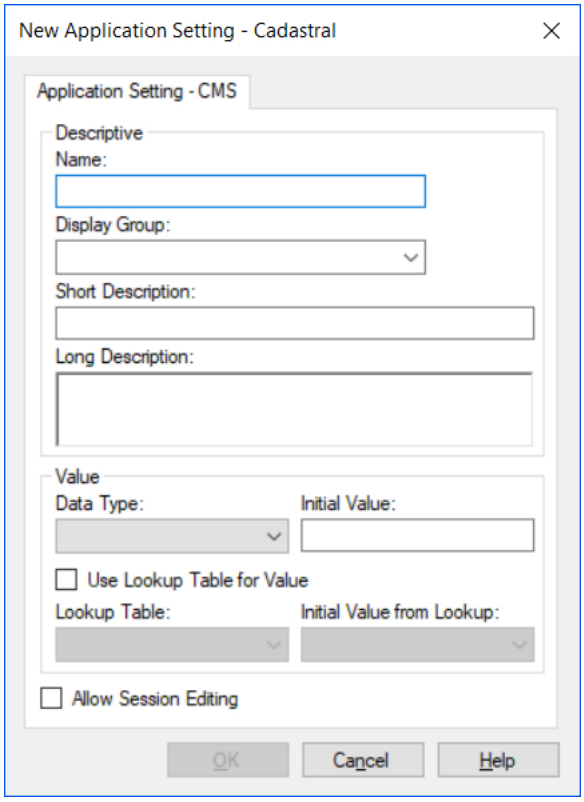


Figure 3 The New Application Setting dialog box

1 Enter the following **Descriptive** information:

- Name – enter a descriptive alpha-numeric name (containing no spaces) of no longer than 30 characters
- Display group – from the list, select the group within the set of application settings that this setting will belong to. You can select an existing group, or specify a new group for the new setting.
- Short description – enter a short, descriptive setting that give the user an indication of what the setting will be used for (maximum length 40 characters)
- Long description – enter a longer description for the setting (maximum length 250 characters)

2 Enter the following **Value** information:

- Data type – select the data type of the value that will be stored.
- Initial value – select one of the following values:
  - Character : allows for string type values
  - Integer : whole numbers
  - Float : decimal numbers
  - Boolean : only allows the values T (true) or F (false)
  - Angle : stores a decimal number, but uses AutoCAD formatting to display the angle value

3 Select the **Use lookup table for value** check box if you want the value to be selected from a lookup table. The **LCODE** value from the lookup table will be stored as the Initial Value once it has been selected.

- If the **Use Lookup table for Value** option has been selected, use this list to select the appropriate table.

4 Select the appropriate initial value from the list of lookup tables.

5 Select the **Allow Session Editing** option to allow the user to select this from the same lookup table specified when the user gets to change a value in the Munsys application. The lookup table has the fixed structure of **LCODE**, **b** – where **LCODE** is the value that is stored in the value of the setting, but **LVALUE** is the description that is displayed in the drop-down entry.

6 Click **OK** to create the new application setting.

Modifying an existing application setting

You will need the MUNSYS_POWER or MUNSYS_ADMIN roles assigned to be able to change the properties of an application setting. Application settings are changed using the Modify Application Setting dialog box.

To modify an existing application setting, do the following:

1 Select the application setting that you want to modify, and then select the **Properties**... option on the **Action** menu or the **context-sensitive** (right-click) menu.

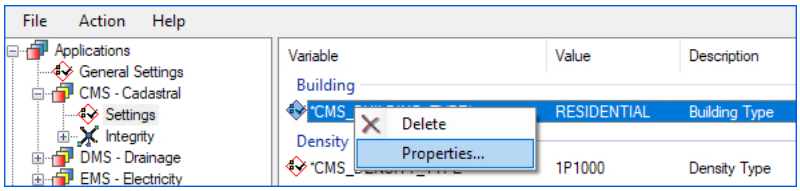


Figure 4 Selecting Application Setting to modify

The Modify Application Setting dialog box is displayed.

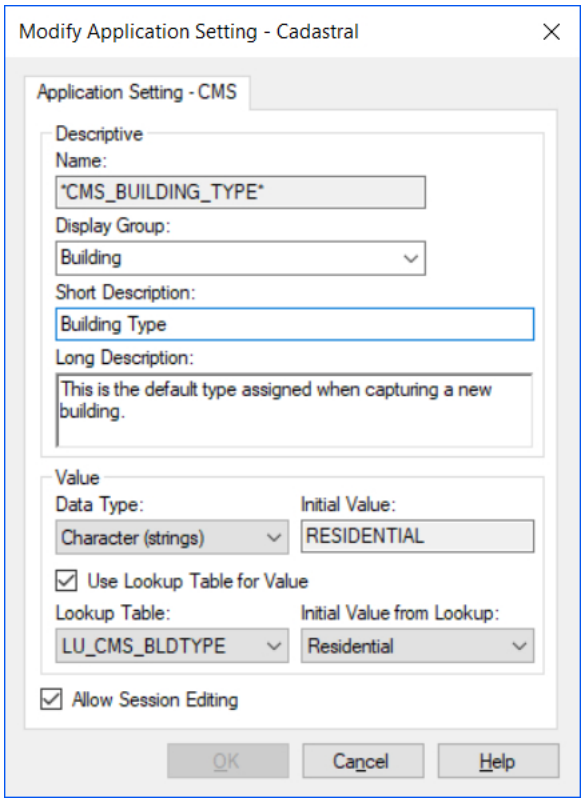


Figure 5 Modify Application Setting dialog box

2 Change the properties as required, and then click **OK**.

Deleting an application setting

You can delete an application setting from a Munsys application. Each Munsys application has its own default settings that are used when data is captured, for example the snap tolerance, tag and symbol scale, node types, etc. Application settings are stored in the MUNSYS_AP_SETTINGS table.

You will need the MUNSYS_POWER or MUNSYS_ADMIN roles assigned to be able to delete an application setting.

To delete an application setting, do the following:

1 Select the application setting that you want to delete, and then select the **Delete** option on the **Action** menu or the **context-sensitive** (right-click) menu.

The following message is displayed:

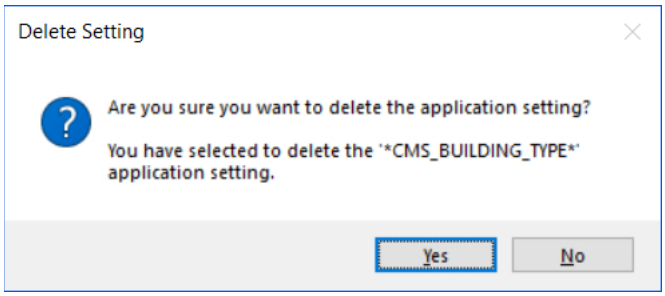


Figure 6 Delete application setting

2 Click **Yes** to delete the application setting from the application.

Integrity Validations

The Munsys Integrity Validation Tools and checks are extremely powerful and allows administrators to manage the integrity of their organization's spatial data to a high degree of sophistication with relative ease.

The Integrity Validation Tools can be divided into three groups of tools:

- Network Check Notifications
- Network Node Integrity
- Object Attribute Integrity

Note For more information on Munsys compound validations please refer to the Munsys Administrators Reference Guide (Munsys14_2_AdminReferenceGuide.pdf) or the Munsys Concepts User Manual (Munsys14_2_ConceptsUserManual.pdf) for more details.

Network Check Notifications

The Network Check Notification Integrity tools are available only for Applications that have Network-able Feature objects. These include Drainage, Electricity, Roads, Sewer and Water. Since the networks in each of these have their own characteristics, the Network Check Notification Integrity tools are application specific.These configurations can be used to monitor changes in the topological relationships of the network.These serve as notifications to rerun network integrity validations.These settings must be in line with the Network integrity test routines for each of the applications.

When any of the items that have been configured for notification are changed, the user will be notified to rerun the network integrity through which the network relationships are then validated.

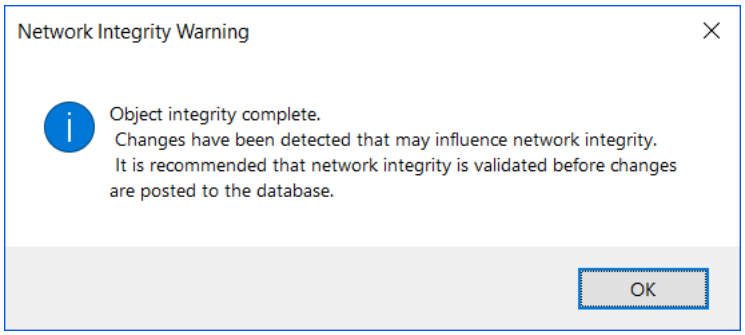


Figure 7 Network Integrity Warning

Creating a new Network Check Notification

Each Munsys application has its own Network Check Notification settings that are used when the network is validated.Network Check Notification settings are stored in the MUNSYS_INTEG_OBJECTS table. When a new notification is created, it inherits the category from the current application and a new record is inserted in the MUNSYS_INTEG_OBJECTS table.

You will need the MUNSYS_POWER or MUNSYS_ADMIN roles assigned to be able to create a new network check notification setting.

1 Select **Network Check Notification** in the tree under the application that you want to create the Notification for.

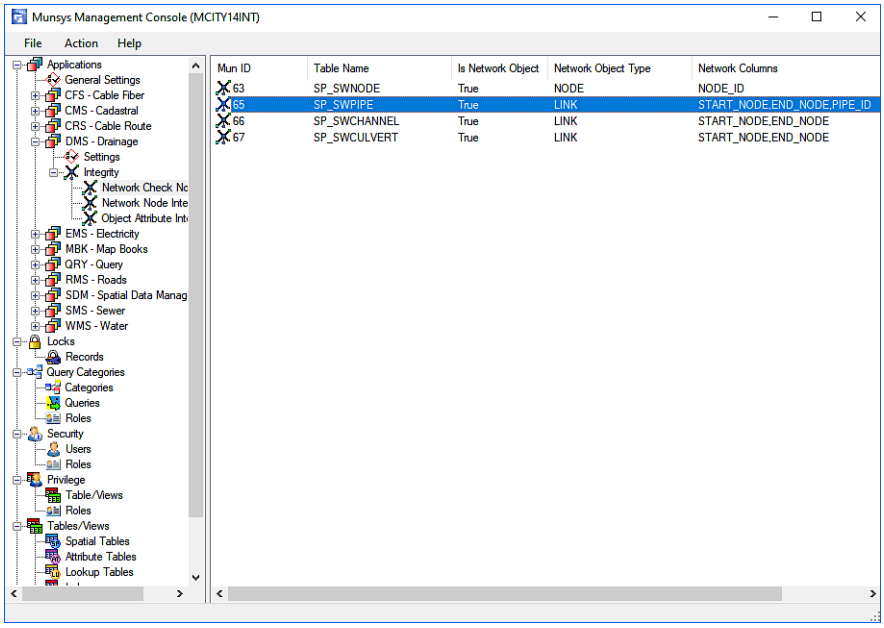


Figure 8 Munsys Management Console: Network Check Notification

2 Select the **New Network Check Notification…** Option on the **Action** menu or the context-sensitive (right-click) menu.

The following dialog is displayed:

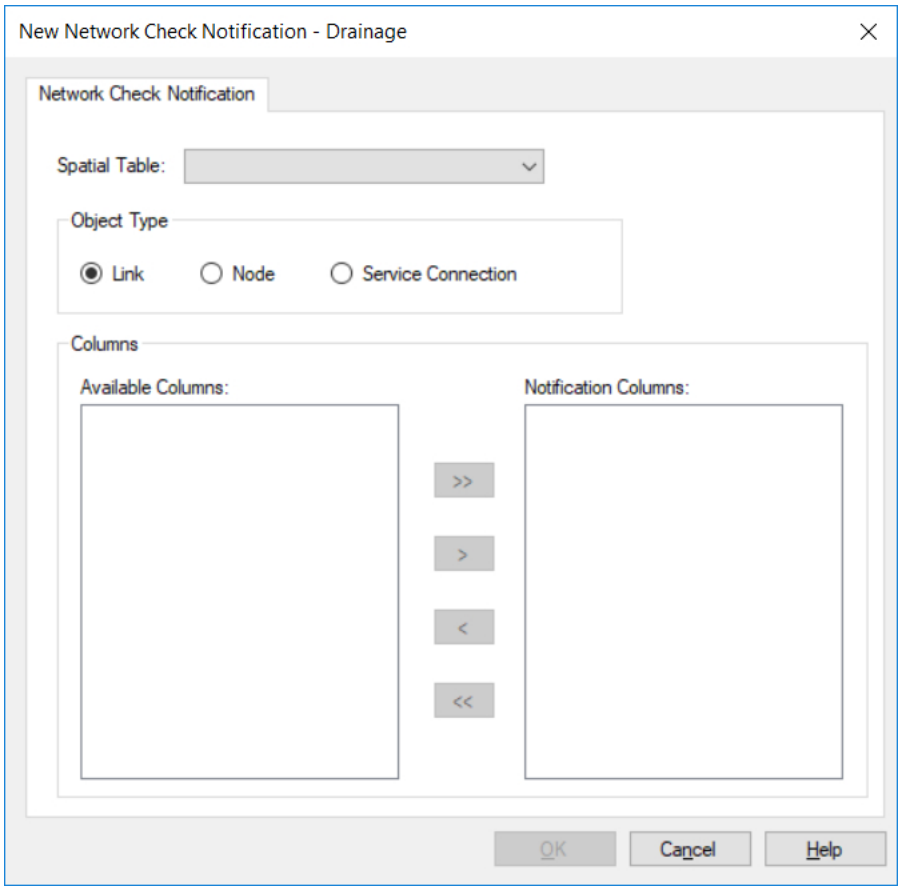


Figure 9 New Network Check Notification

3 Select the Spatial Table for which you want to add a Notification.Note that only those objects are listed that are a part of the Application under which you have made the selection. Once selected, you will notice that the columns from the selected table are now listed in the Available Columns list.

4 Next select what role the selected table plays within the network.You have three options:

- Link – these must be linear objects that act as topological links (polylines) within the network.
- Node – these must be point type objects that act as topological nodes within the network.
- Service Connection – these must be linear type objects that act as topological connections between the network and the service delivery endpoint, usually a polygon type entity such as a property.

5 Next, select the columns from the Available Columns list that must be used to notify users to rerun Network Integrity.

6 Click **OK** to save your changes to the database.

Editing an existing Network Check Notification

You will need the MUNSYS_POWER or MUNSYS_ADMIN roles assigned to be able to edit an existing network check notification setting.

To edit an existing network check notification, do the following:

1 Select **Network Check Notification** in the tree under the application where you want to edit an existing notification.

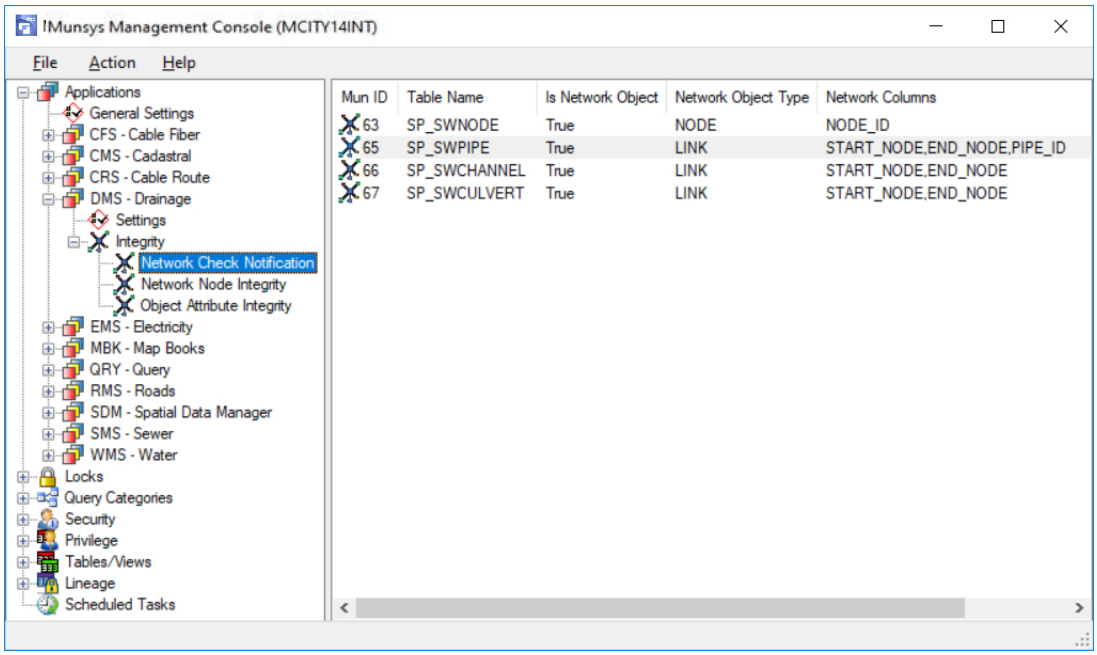


Figure 10 Select Network Check Notification

2 Select the Existing Notification record in the right hand notifications lists.

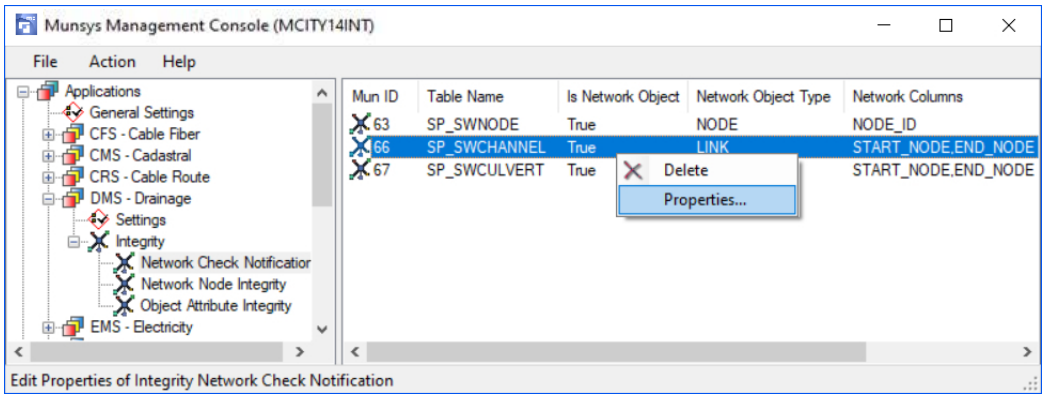


Figure 11 Selecting Network Check Notification Properties

3 Select the **Properties** Option on the **Action** menu or the **context-sensitive** (right-click) menu. The following dialogue is displayed:

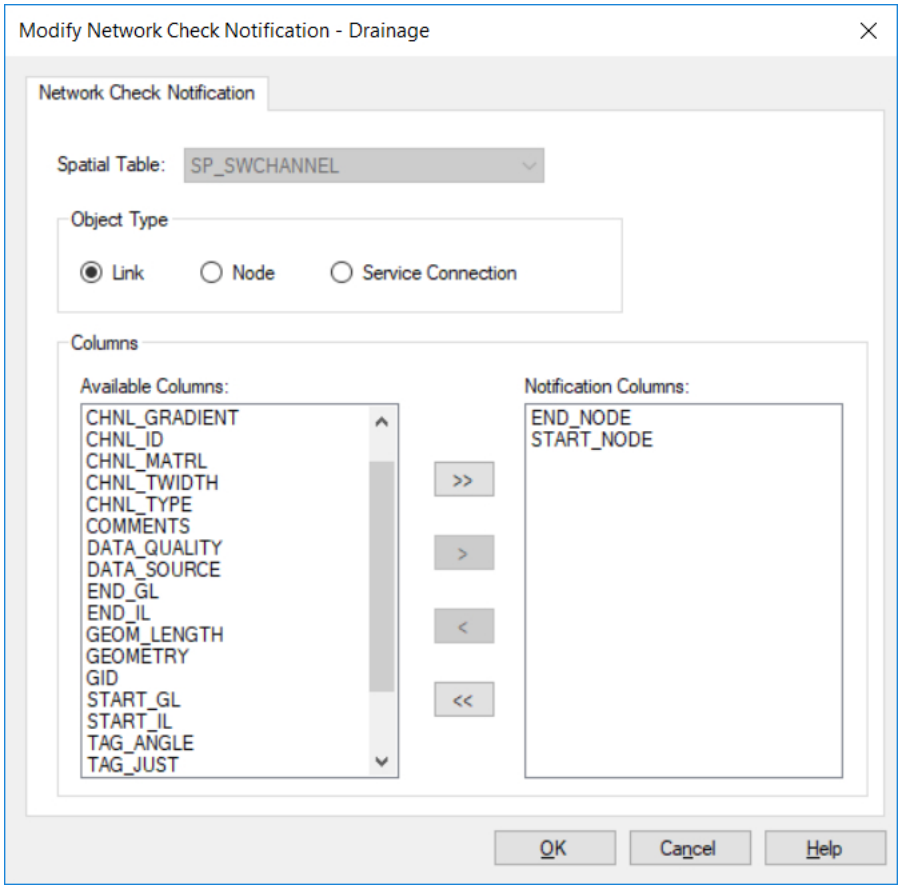


Figure 12 Modify Network Check Notification

4 Edit the Notification Properties as needed.

5 Click **OK** to post the changes to the database.

Deleting an existing Network Check Notification

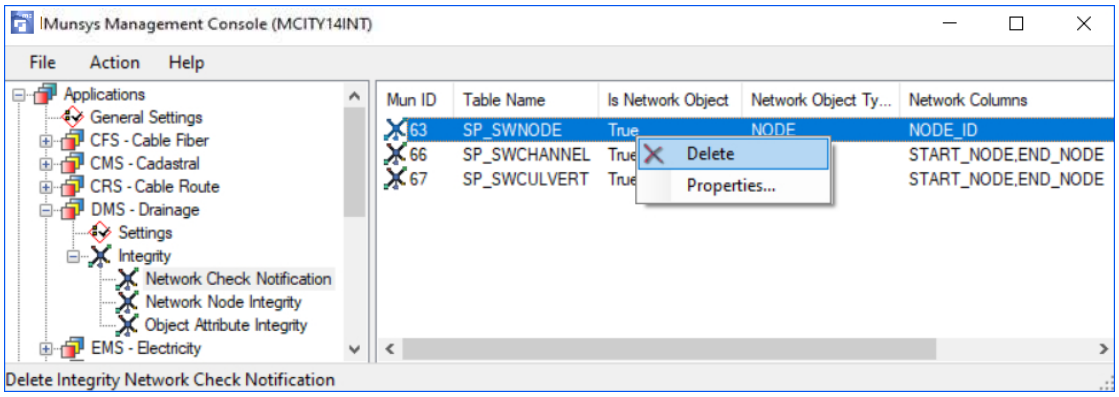
You will need the MUNSYS_POWER or MUNSYS_ADMIN roles assigned to be able to delete network check notification setting.

To delete an application setting, do the following:

1 Select **Network Check Notification** in the tree under the application where you want to edit an existing notification.

2 Select an Existing Notification record in the right-hand notifications lists.

3 Select the **Delete** Option from the **Action** menu or the **context-sensitive** (right-click) menu.



The following message is displayed:

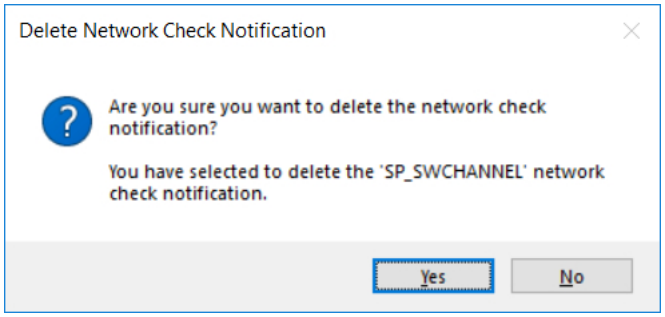


Figure 13 Delete Network Check Notification record

4 Click **Yes** to delete the Network Check Notification record.

Network Node Integrity

The Network Node Integrity tools are available only for Applications that have Network-able Feature objects.These include Drainage, Roads, Sewer and Water.Since the networks in each of these have their own characteristics, the Network Node Integrity tools are application specific.These configurations can be used to configure different Network characteristics.

The Network Node Integrity tools can be used to determine how specific node types should behave in the network.Two rule types are supported:

- The INTERMEDIATE rule allows a node to be inserted between the start and end nodes of a link object.
- The LINKLIMITS rule specifies a minimum and a maximum number of links allowable for the node type.

Creating a new Network Node Integrity Record

Each Munsys application has its own Network Node Integrity settings that are used when the network is validated.Network Node Integrity settings are stored in the MUNSYS_INTEG_NODES table.You will need the MUNSYS_POWER or MUNSYS_ADMIN roles assigned to be able to create a new network node integrity setting.

To create a new network node integrity setting, do the following:

1 Select **Network Node Integrity** in the tree under the application that you want to create the Node Integrity record for.

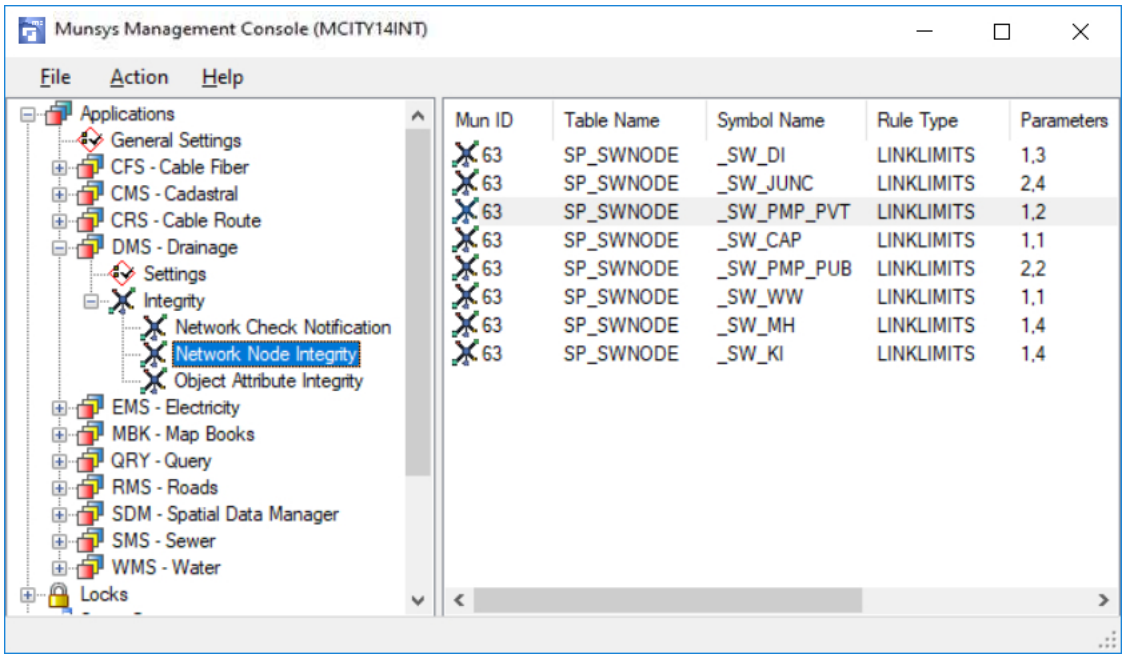


Figure 14 Munsys Management Console: Network Node Integrity

2 Select the **New Network Node Integrity…** Option on the **Action** menu or the **context-sensitive** (right-click) menu.

The following dialog is displayed:

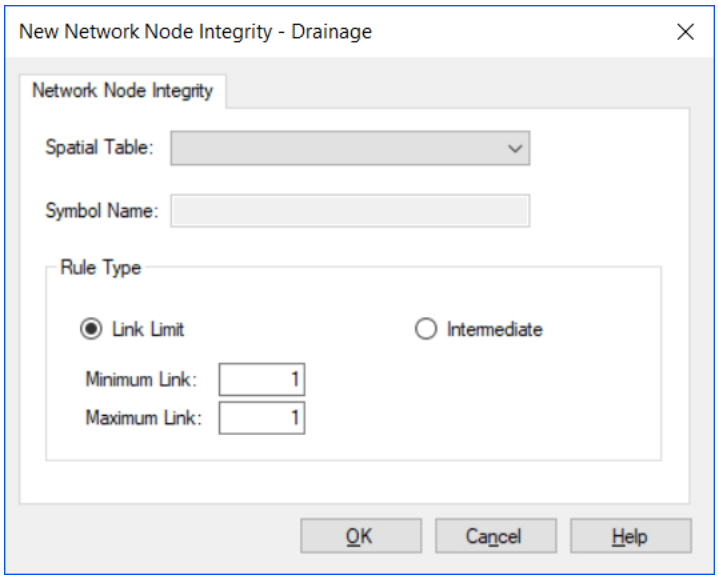


Figure 15 New Network Node Integrity

3 Select the Node table for which you want to add Network Node Integrity.

4 Enter the name of the Node Type (Munsys SYM_NAME) for which node integrity needs to be added.

5 Select the Rule type for the Node Integrity.There are two options:

- Link Limit – This option allows you to specify the range of links that are allowed for one node of the given type.The Minimum Link setting will test for a minimum number of links while the Maximum Link setting will test for a maximum number of connected links. Examples:
  - A reducer in a water network must have an incoming AND an outgoing link but no more.The minimum and maximum must therefore both be set to “2”.
  - An Endcap in most networks must always be limited to one link only. The minimum and maximum must therefore both be set to “1”.
  - A pre-cast storm water manhole must have at least ONE incoming and one outgoing link, but it could have multiple incoming links, depending on its design. The minimum should therefore be set to “2” while the maximum should be in line with its specifications but would usually be three or more…
- Intermediate – This node type does not break a pipe but is merely snapped onto the pipe.Example:
  - some organizations require that nodes such as Hydrants do not break pipes but sit intermediately between the start and end nodes.

6 Click **OK** to update the database and save your new settings.

Editing an existing Network Node Integrity

You will need the MUNSYS_POWER or MUNSYS_ADMIN roles assigned to be able to edit an existing network node integrity setting.

To edit an existing network node integrity setting, do the following:

1 Select **Network Node Integrity** in the tree under the application where you want to edit an existing Node Integrity record.

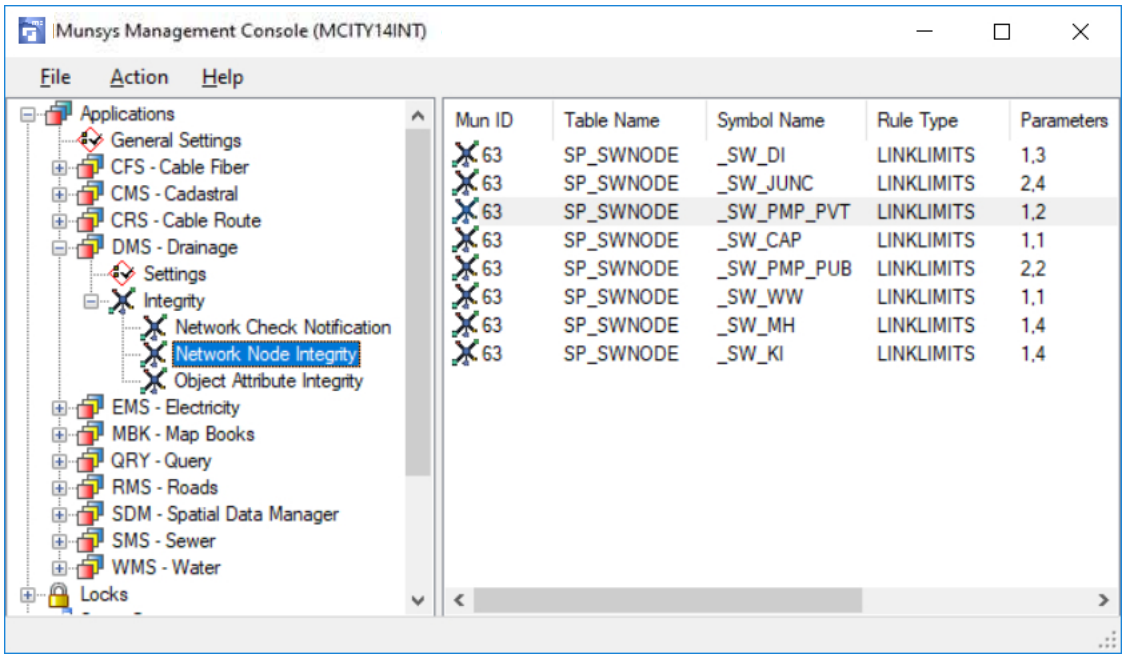


Figure 16 Select Network Node Integrity

2 Select the existing network node integrity record in the right-hand panel, and then select the **Properties** option on the **Action** menu or the **context-sensitive** (right-click) menu.

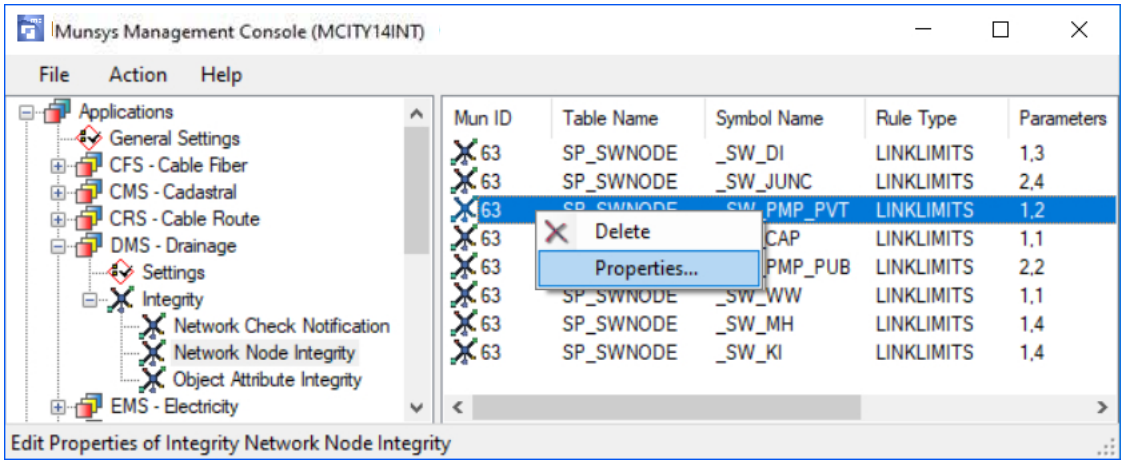


Figure 17 Selecting Network Node Integrity Properties

The following dialog is displayed:

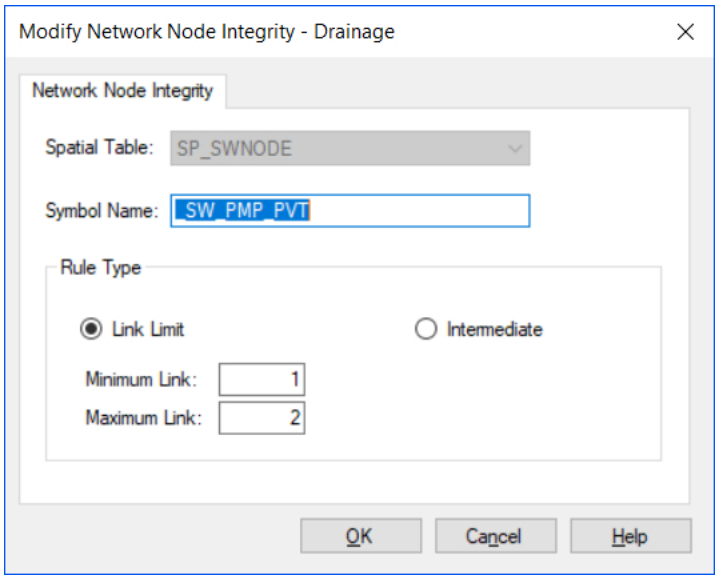


Figure 18 Modify Network Node Integrity

3 Make any changes to the values on the dialog as needed.

4 Click **OK** to update the existing Network Node Integrity record.

Delete an existing network Node Integrity

You will need the MUNSYS_POWER or MUNSYS_ADMIN roles assigned to be able to edit an existing network node integrity setting.

To edit an existing network node integrity setting, do the following:

1 Select **Network Node Integrity** in the tree under the application where you want to delete an existing Node Integrity record from.

2 Select the existing network node integrity record in the right-hand panel, and then select the **Delete** option on the **Action** menu or the **context-sensitive** (right-click) menu.

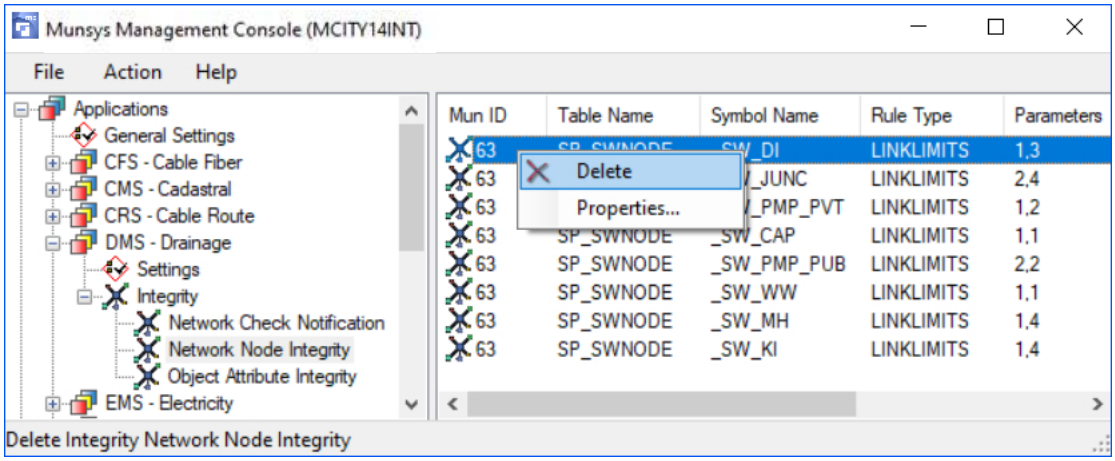


Figure 19 Select Delete Network Node Integrity

The following message is displayed:

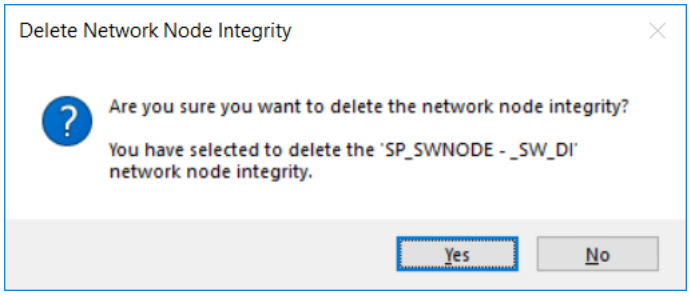


Figure 20 Delete Network Node Integrity Notification

3 Click **Yes** to delete the existing Network Node Integrity

Object Attribute Integrity

The Object Attribute Integrity Rules refer to integrity rules allows administrators to set up a range of integrity rules for any of the attributes of a data object.

Note NOTE: Multiple rules can be set for a specific attribute on any table.

When multiple checks are specified, the Rule order will determine in what order rules will be run to validate the Object's Integrity.The following rule types can be set:

- Compare Check – This rule will allow users to check the relationship between two attribute fields of the same table.
- Not Null – This forces attributes to have values.
- Range – Valid for NUMERICAL attributes only – the attribute must fall within a range defined as a Minimum and Maximum range: [Not] greater than or equal to x and less than or equal to y.
- Substring – This rule can be applied multiple times for the same attribute and allows validation of parts or substrings of the attribute value: The substring function returns a portion of string, beginning at a specified position in the string
- Value Check – This rule allows users to check the value of an attribute.

Creating an new Object Attribute Integrity

It is possible to set up Object Attribute Integrity rules for each Munsys object class. Object Attribute Integrity rules are stored in the MUNSYS_INTEG_ATTR table. You will need the MUNSYS_POWER or MUNSYS_ADMIN roles assigned to be able to create new object attribute integrity rules.

To create a new object attribute integrity rule, do the following:

1 Select **Object Attribute Integrity** in the tree under the application under which the object falls for which you want to create an object attribute integrity rule.

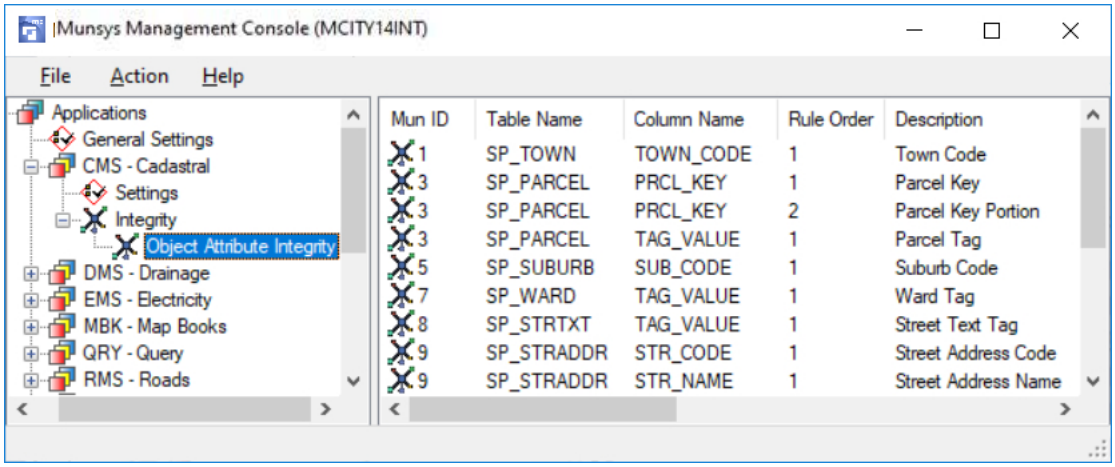


Figure 21 Munsys Management Console: Object Attribute Integrity

2 Select the **New Object Attribute Integrity…** option on the **Action** menu or the **context-sensitive** (right-click) menu.

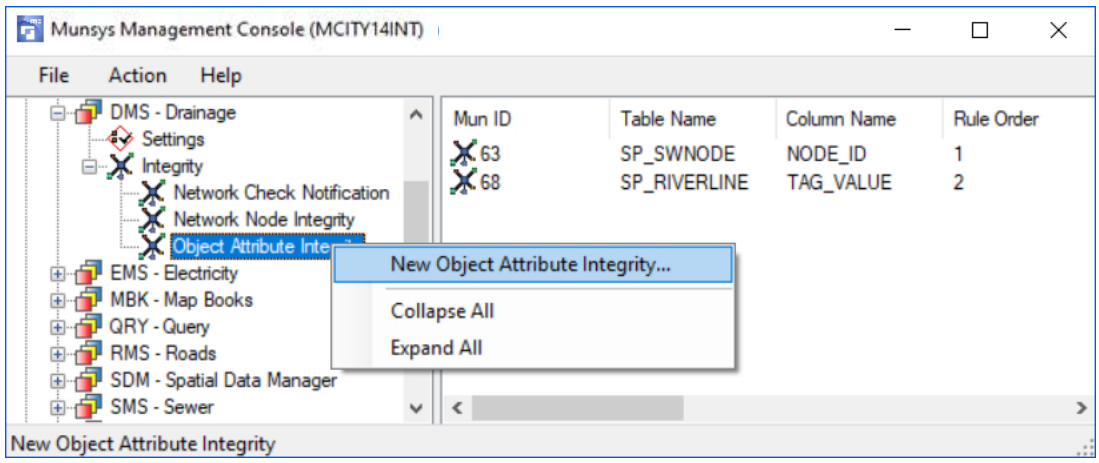


Figure 22 Create new Object Attribute Integrity

The following dialog is displayed:

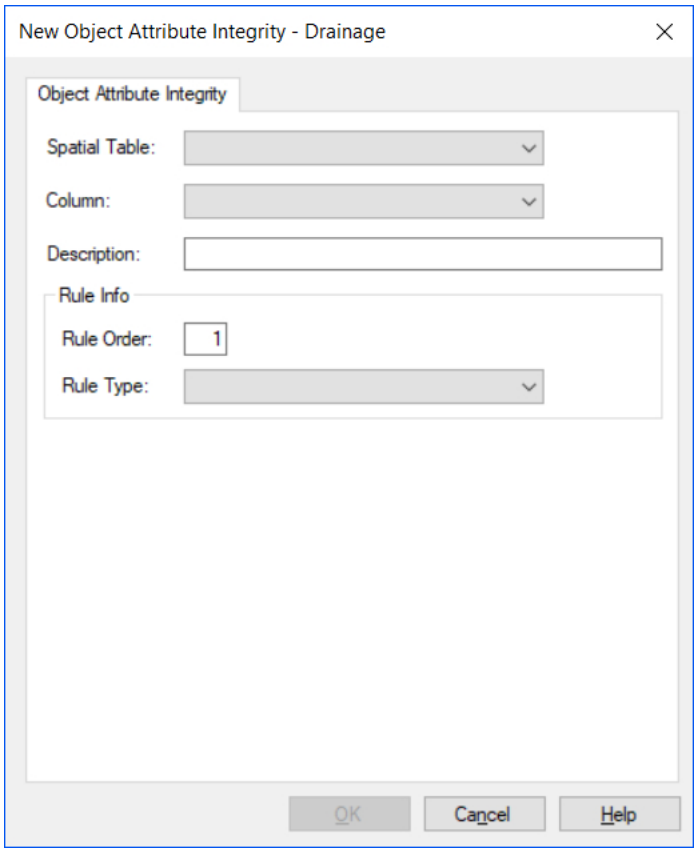


Figure 23 New Object Attribute Integrity

3 Select the Spatial table for which you want to add an object integrity rule.

4 Select the Column on which you want to add the rule.

5 Add a Description for the rule.It is sensible to add concise, but clear descriptions.

6 The system will automatically provide a Rule order number which is incremental based on existing rules.This value is not editable when first creating the rule.(To change the rule order, edit the rule afterwards).

7 Select a Rule type.Munsys Integrity offers five rule types, each with different parameters. These are:

- Compare Check – This rule will allow users to check the relationship between two attribute fields of the same table.Use this rule to perform Compound Validations.The Rule takes two parameters: An Operator and a Column name. Example - use this rule to ensure that the start invert level of a pipe is shallower (has a bigger value) than the end invert level.
- Not Null – This forces attributes to have values.

Note When setting a column to being Mandatory in Munsys Management Console, an Application Object Attribute Integrity rule with a rule type of NOT NULL is automatically created. When objects are checked using the Object Integrity checks within the Munsys Application, if a NOT NULL rule type is found, the Attribute Marker will record the following error:
[Column_name] may not be NULL, New Object - [Table Name]

- Range – Valid for NUMERICAL attributes only – the attribute must fall within a range defined as a Minimum and Maximum range.Example: use this rule to check that service connections are longer than 1 meter but shorter than 50 meters.And note that the dialogue states that these values are inclusive.
- Substring – This Rule allows users to validate substrings and can be applied multiple times for the same attribute.This is similar to “Mask” integrity rules available in other applications.Example: If you want to ensure that the Plan number for a property is valid, you may include two rules, the first stating that the first two characters must be alpha

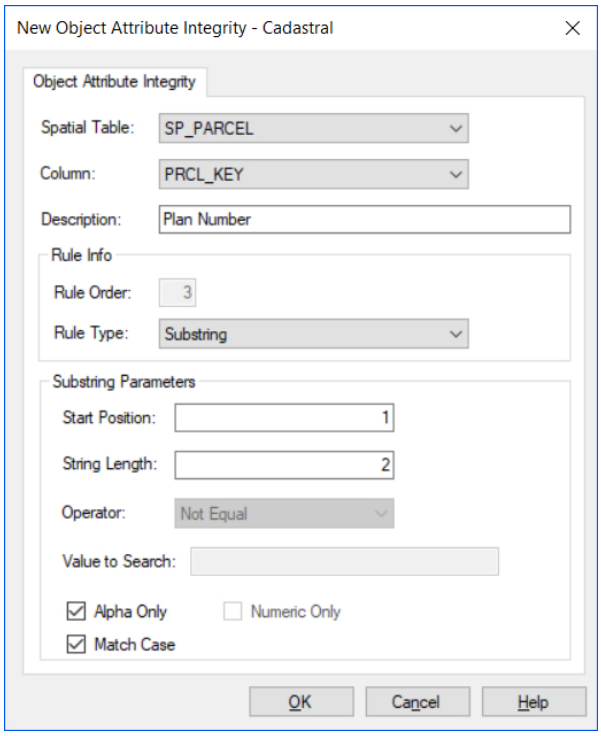


Figure 24 Substring Alpha Only rule

and the remainder of the string must be numeric

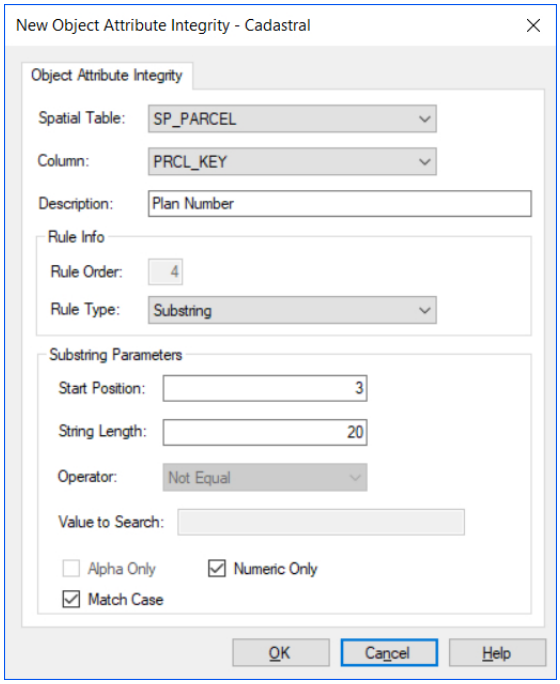


Figure 25 Substring Numeric Only rule

- Value Check – This rule allows users to check the value of an attribute against a single pre-set value using these operators.

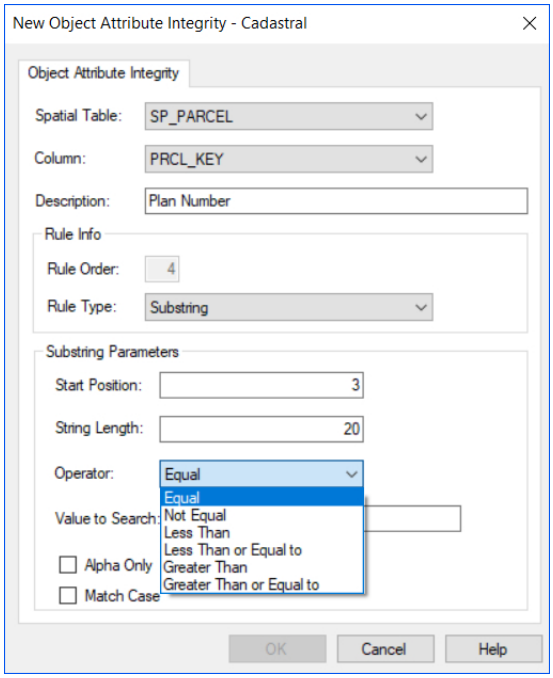


Figure 26 Substring Operator value selection

Edit an existing Object Attribute Integrity

You will need the MUNSYS_POWER or MUNSYS_ADMIN roles assigned to be able to edit an existing object integrity rule.

To edit an existing object integrity rule, do the following:

1 Select **Object Attribute Integrity** in the tree under the application under which the object falls, for which you want to edit an object attribute integrity rule.

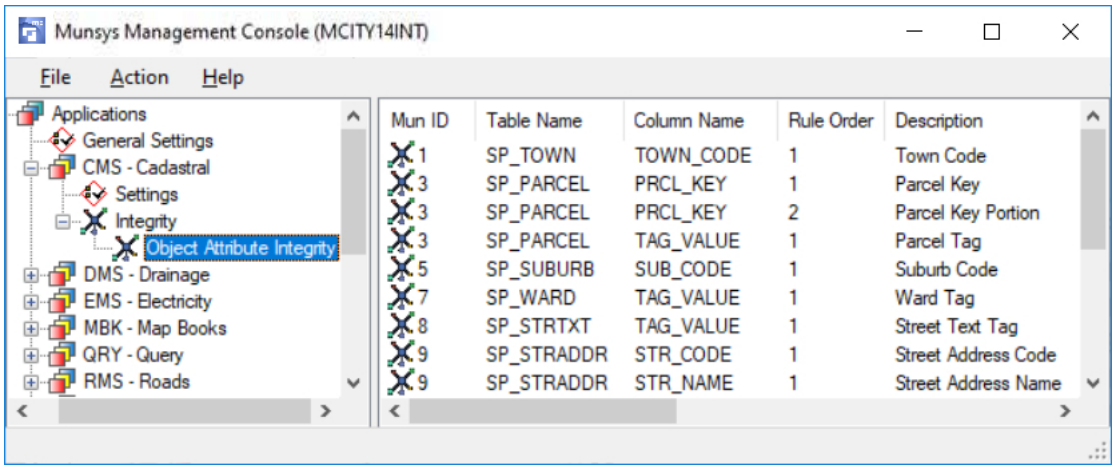


Figure 27 Select Object Attribute Integrity

2 Select the existing Object Attribute Integrity Rule in right-hand panel.

3 Select the **Properties** option on the **Action** menu or the **context-sensitive** (right-click) menu.

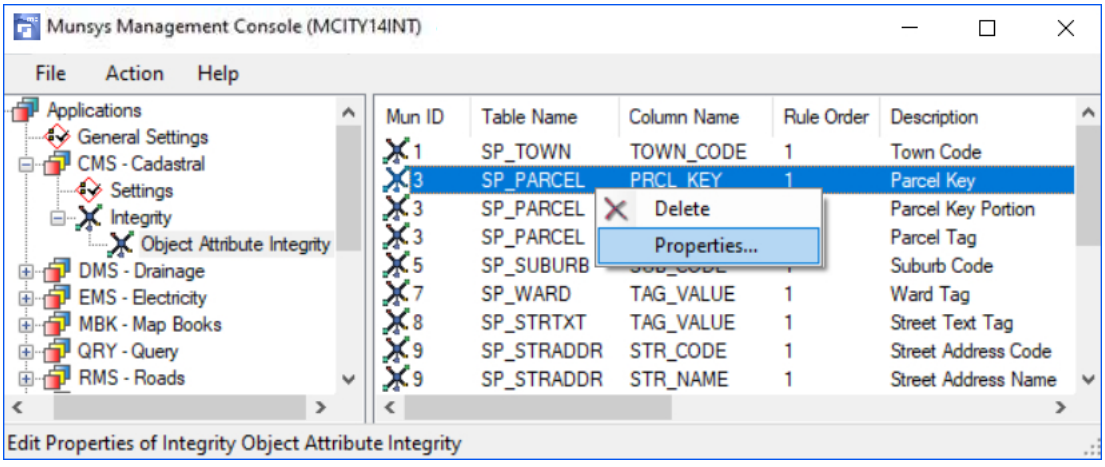


Figure 28 Selecting Object Attribute Integrity Properties

4 The following dialog will be displayed:

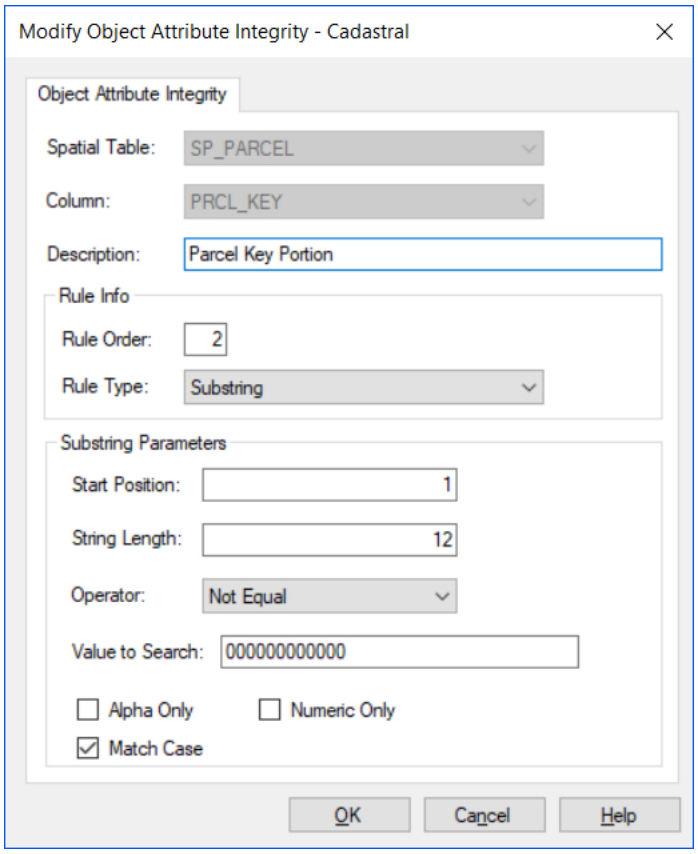


Figure 29 Modify Object Attribute Integrity

5 Edit the values on the dialogue to match the desired rule parameters.

Note Note if you want to reorder Rules, it is best to first change the rule order using a different number set as the system will not allow rules with the same order.

6 Once satisfied with the parameter settings, click **OK** to save your edits.

Delete an Existing Object Attribute Integrity

You will need the MUNSYS_POWER or MUNSYS_ADMIN roles assigned to be able to delete an existing object integrity rule.

To delete an existing object integrity rule, do the following:

1 Select **Object Attribute Integrity** in the tree under the application under which the object falls, for which you want to edit an object attribute integrity rule.

2 Select the existing Object Attribute Integrity Rule in right-hand panel.

3 Select the **Delete** option on the **Action** menu or the **context-sensitive** (right-click) menu.

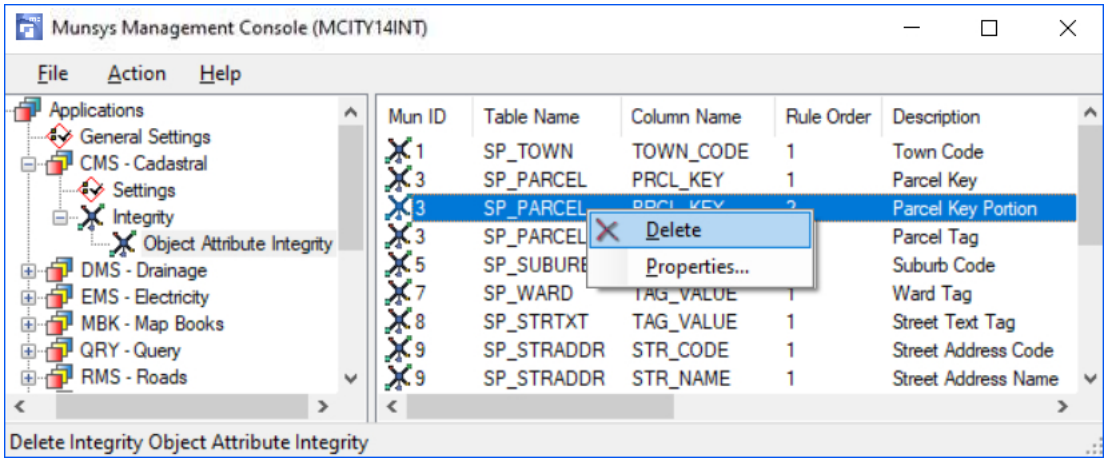


Figure 30 Select Delete Object Attribute Integrity

The following dialog will be displayed:

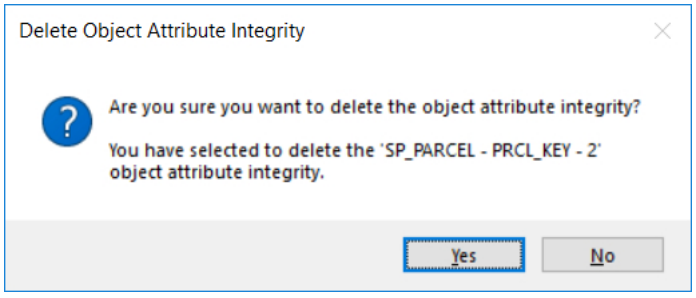


Figure 31 Delete Object Attribute Integrity Notification

4 Click **Yes** to delete the Object Attribute Integrity.

Chapter 6
Record locking

Introduction

Munsys record locking is activated by default, ensuring data integrity within the working environment and restricting the editing of a spatial object to one user at a time. In the Munsys Management Console, the following record locking issues are managed from the Locks tree:

- Locking behavior – enabling or disabling the record locking mechanism, and setting lock reminders for users then they log on or off.
- Lock reporting and overriding – displays the current status with regard to users and respective locked objects, and allows the database administrator to unlock objects currently locked by users.

Record locking in the Munsys schema can be disabled permanently or temporarily.

- Disable permanently – the option to clear all existing locks in the database is recommended before record locking is disabled permanently to ensure that no locking information is retained in the database.
- Disable temporarily – this can be done to allow users with locked objects to post changes to database, although these objects are not locked in the database. In this case, the administrator should verify existing locks before clearing all the locks from the database. The administrator should also ensure that users do not query any objects from the database while the lock administration is disabled. Clearing all the existing locks may cause problems when users try to post the changes to the database after the lock administration is enabled again. It is therefore recommended that the administrator notify all the users to post changes to the database before disabling the lock administration temporarily. If any objects are still locked and the administrator ensures that these locks can be cleared, the option to clear all the locks can be used.

The Locks tree: overview

The Locks tree contains the Records branch, which displays a summary of the total number of records currently locked by all users in all tables.

- Clicking on Records in the Locks tree displays each user name that has locked objects, the spatial table where the objects reside and the number of objects locked for that table. You can drag the column headers to display in any order that you want.

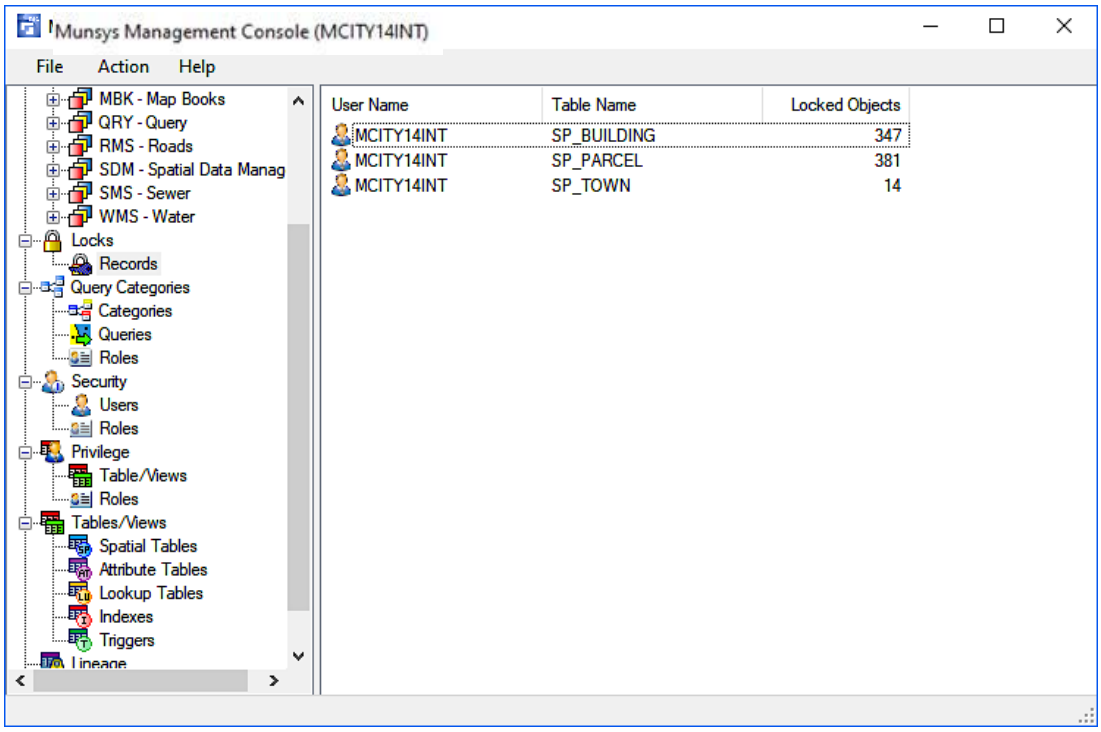


Figure 1 Locks tree: Records

Enabling or disabling record locking

Record locking is activated by default in any Munsys schema. From the Locks tree in the Munsys Management Console, you can change record locking properties by disabling or enabling record locking in a schema, or by changing the record locking status reminders for users.

To be able to work with record locking properties, you need to have the following roles/privileges assigned:

- Munsys role: MUNSYS_ADMIN
- Database privileges: SELECT, INSERT, UPDATE and DELETE on MUNSYS_DB_SETTINGS

To specify record lock settings

Right-click on Locks, and then select the Properties option on the context-sensitive menu, as seen below:

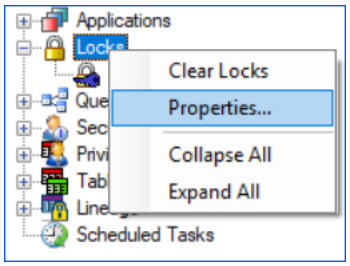


Figure 2 Record Locking: Properties

The Record Locking dialog box is displayed.

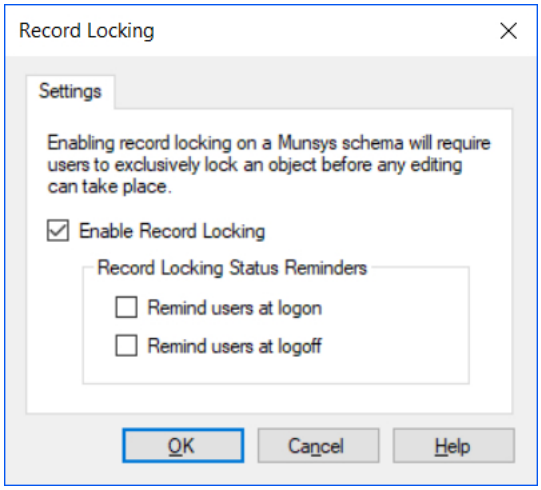


Figure 3 The Record Locking dialog box

1 Select or clear the **Enable Record Locking** check box to enable or disable record locking in the schema.

2 Set the record locking status reminders as required (this group is unavailable when record locking has been disabled):

- Remind users at logon reminds users that they still have objects locked when they log on to the schema.
- Remind users at logoff reminds users that they still have objects locked when they log off from the schema

3 Click **OK** to apply the settings that you have specified.

Clearing all the locks in the database

Using the Clear Locks menu item, the database administrator can clear all the locks in the database. Please note that unlocking all locked records could result in the loss of data.

To be able to clear all the locks in the database, you need to have the following roles/privileges assigned:

- Munsys role: MUNSYS_ADMIN
- Database privileges: SELECT, INSERT, UPDATE and DELETE on MUNSYS_LOCKED_OBJECT

To clear all the locks in the database

1 Right-click on **Locks**, and then select the **Clear Locks** option on the context-sensitive menu, as seen below:

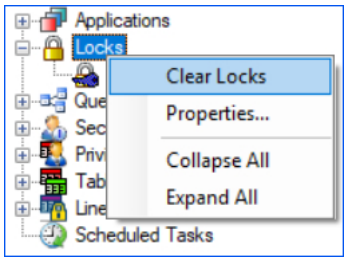


Figure 4 Record Locking: Clear Locks

The following confirmation message is displayed:

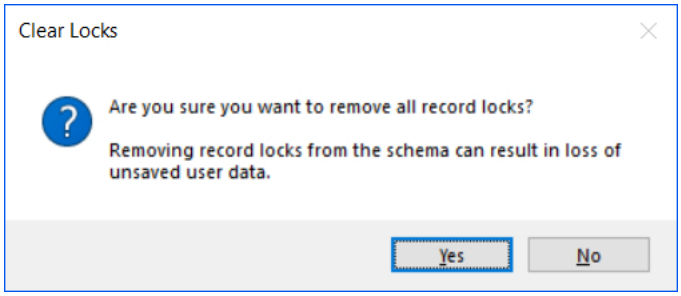


Figure 5 Clear Locks

2 Click **Yes** to clear all the locks in the database.

The locks are cleared, and the content pane refreshed and updated accordingly.

Working with locked records in the database

To be able to work with locked records in the database, you need to have the following roles/privileges assigned:

- Munsys role: MUNSYS_ADMIN
- Database privileges: SELECT, INSERT, UPDATE and DELETE on MUNSYS_LOCKED_OBJECT

The Records item in the Locks tree displays the number of objects that are currently locked for all users in all the spatial tables in the database. Selecting this item displays the following summary list in the content pane:

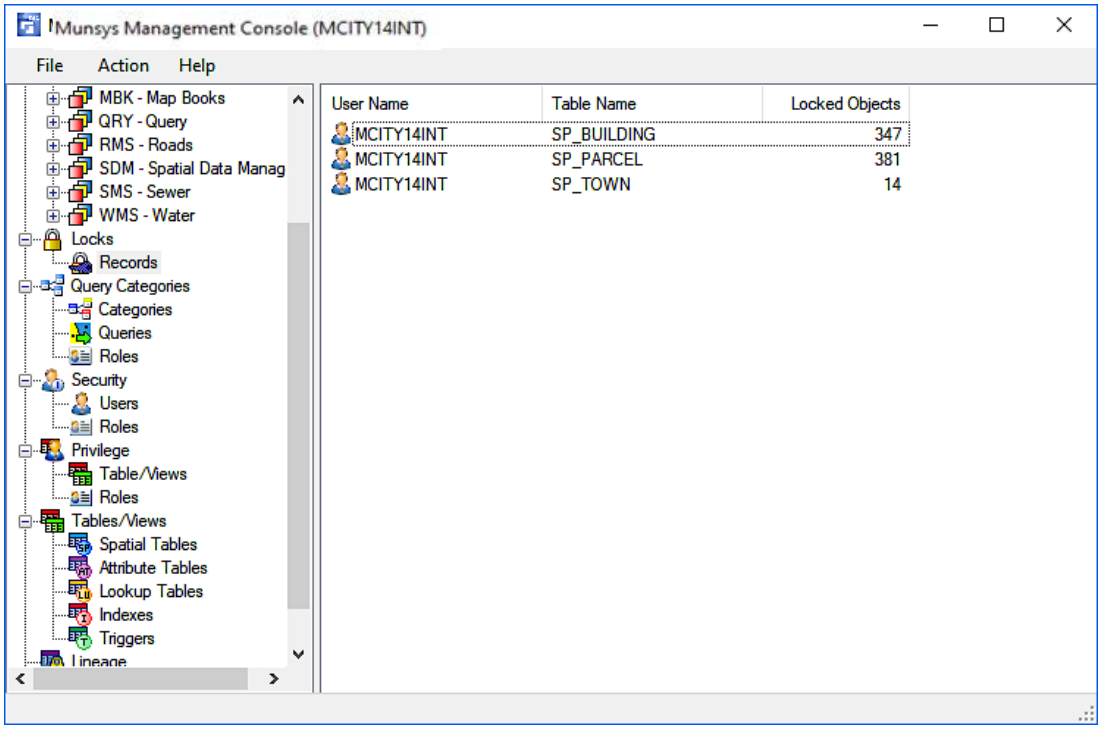


Figure 6 Record locking summary

Viewing locked objects

You can view details of the objects that have been locked by a user on the Record Locking Detail dialog box.

To access the Record Locking Detail dialog box, do the following:

1 Right-click on an item in the summary list, and then select the **Details** item on the context-sensitive menu:

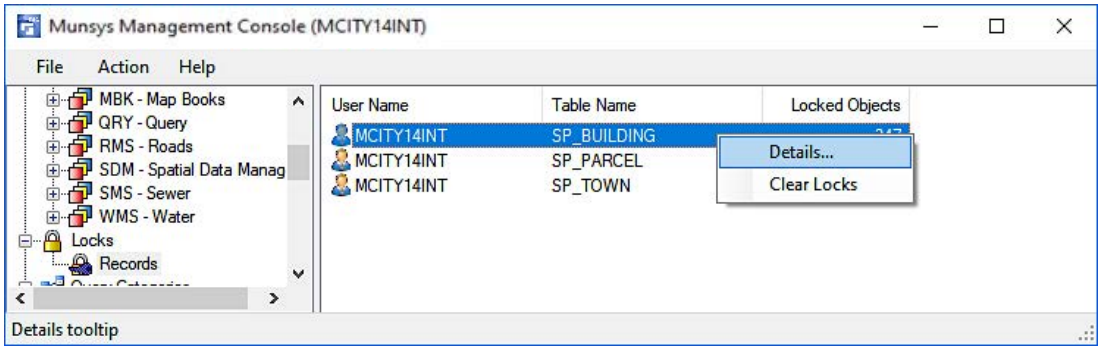


Figure 7 Selecting Locked Record Details

The Record Locking Detail dialog box is displayed:

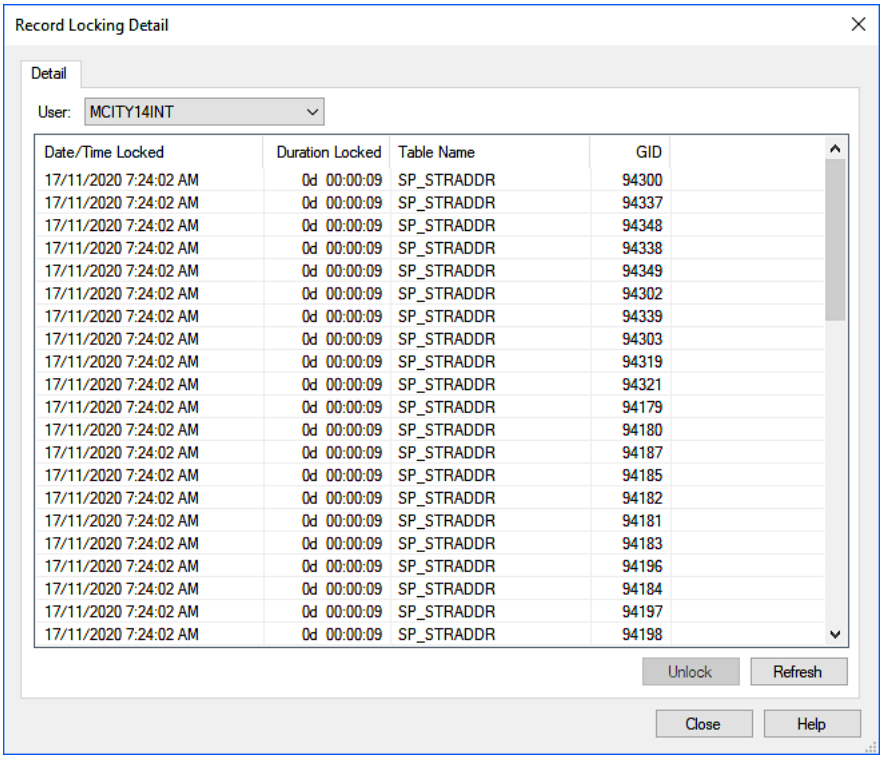


Figure 8 The Record Locking Detail dialog box

Unlocking locked objects

You can clear the objects that have been locked by a user by right-clicking on an item in the summary list, and then selecting the Clear Locks item on the context-sensitive menu, as seen below:

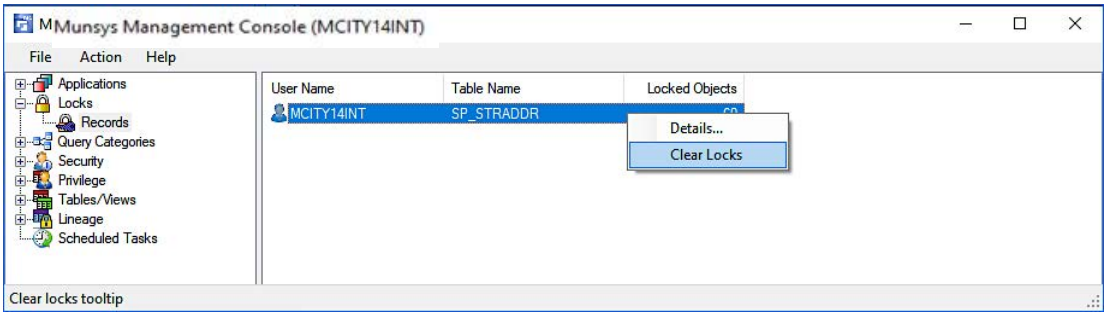


Figure 9 Selecting Clear Locks Option

You can also unlock locked objects from the Record Locking Detail dialog box:

1 Select the object(s) that you want to unlock, and then right-click on one or more selected records to display a context-sensitive menu that you can use to clear selected records for unlocking.

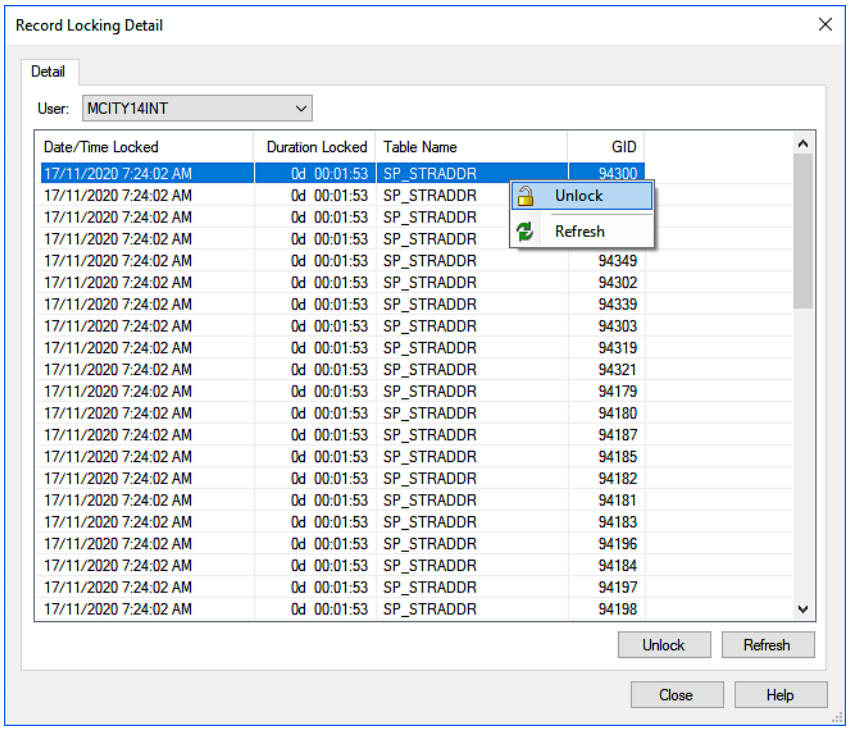


Figure 10 The Record Locking Detail dialog box: unlocking locked objects

2 Records can also be unlocked by using the **Unlock** button on the **Record Locking Detail** dialog box.

3 Click **Refresh** to refresh the dialog box once you have unlocked the locked records.

Chapter 7
Query Categories

Introduction

Munsys Query categories are maintained from the Query Categories tree in the Munsys Management Console (MMC). You need to have the MUNSYS_POWER or MUNSYS_ADMIN roles assigned to be able to manipulate query categories.

Munsys recognizes the following types of query categories:

- Uncategorized – by default any new query (not a user query) is seen as being uncategorized until it has been allocated to a category. Queries may be moved to or from the uncategorized group, but the category itself is a special case whereby the name may not be changed – only the content. The uncategorized group may also not be granted other roles – all queries in this group are seen to be granted the MUNSYS_ALL_QUERY role. As soon as a query is moved to a category or as a user query, it cannot also belong to the uncategorized group – the query no longer has this status.
- User – whenever a user creates a new query, it is automatically seen as a personal query, hence called a User query. Other users cannot see any other user's queries until they are shared in the pool of Uncategorized or Categorized queries.
- Custom Categories – each category is provided with a unique name. When a new category is created, it is automatically granted the MUNSYS_ALL_QUERY role. A category may be granted multiple roles.

Munsys provides default custom categories as part of a new Munsys schema. The power user can customize these and/or add personal categories. The following categories are created in a new schema, synonymous with the Munsys applications and the default roles will be granted to these categories:

- Fiber – MUNSYS_CFS_QUERY
- Cable Routes – MUNSYS_CRS_QUERY
- Cadastral – MUNSYS_CMS_QUERY
- Drainage – MUNSYS_DMS_QUERY
- Electricity – MUNSYS_EMS_QUERY
- Roads – MUNSYS_RMS_QUERY
- Sewer – MUNSYS_SMS_QUERY
- Water – MUNSYS_WMS_QUERY

The Query Categories tree: overview

The Query Categories main tree consists of three branches (nodes):

Note When an existing Munsys schema is upgraded to a Munsys 14 schema, all system queries are classified as uncategorized, and the power user/administrator will be required to move these into categories.

- Categories – allows the user to create and/or delete categories and provides a mechanism for the user to allocate roles to categories.
- Queries – the properties of the queries are not edited here, but the user may arrange queries into various categories. The visibility of queries within a category will be restricted by the roles granted to a user. However, in the MMC the power user has the right to see all queries and categories regardless of roles granted, having the responsibility to arrange the queries and query categories.
- Roles – contains the roles that exist in the database and have been granted to the various query categories.

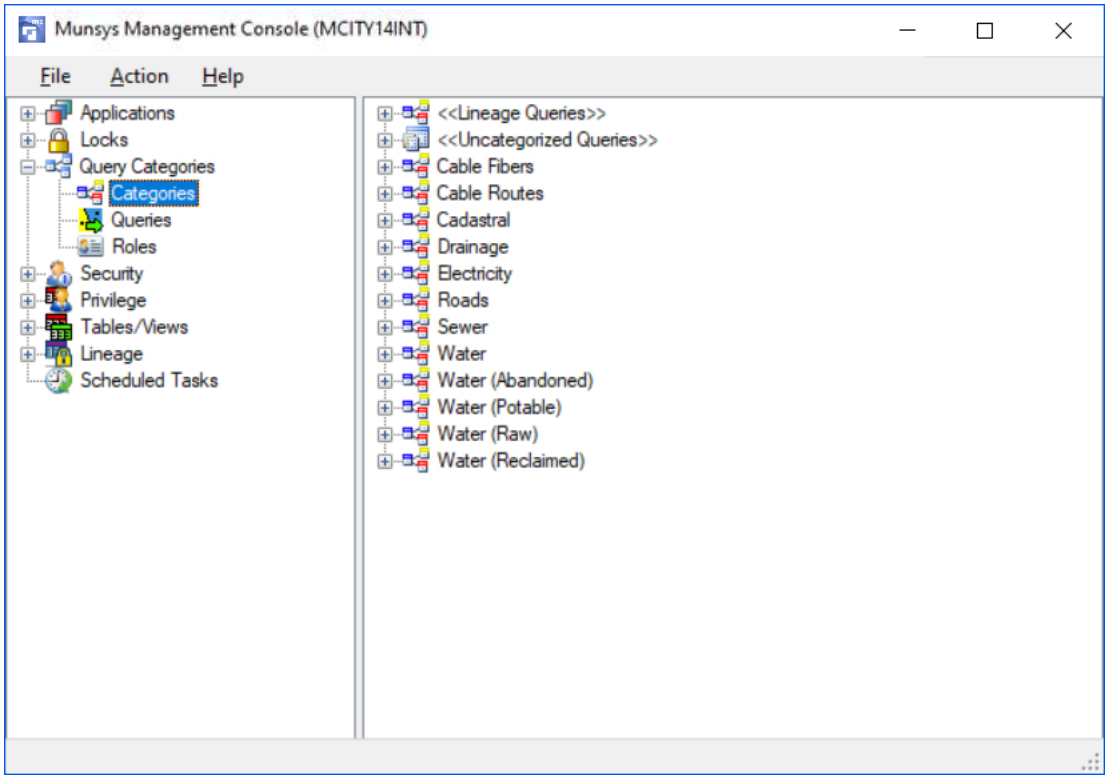


Figure 1 Munsys Management Console: Query Categories tree

The sub-trees within the Query Categories tree are displayed when a user has the MUNSYS_POWER or MUNSYS_ADMIN role assigned. For more information about the various Munsys roles, see Chapter 8: Security in the Munsys Management Console.

Creating a new query category

To be able to create a new query category, you need to have the following roles/privileges assigned:

- Munsys role: MUNSYS_POWER or MUNSYS_ADMIN
- Database privileges: UPDATE on query-related tables

To create a new query category

1 Right-click on the **Categories** node of the **Query Categories** tree, and then select the **New Category...** option on the context-sensitive menu.

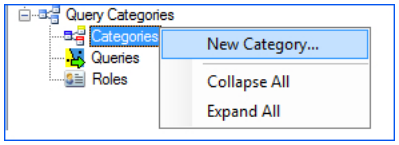


Figure 2 Creating a new Query

The New Query Category dialog box is displayed:

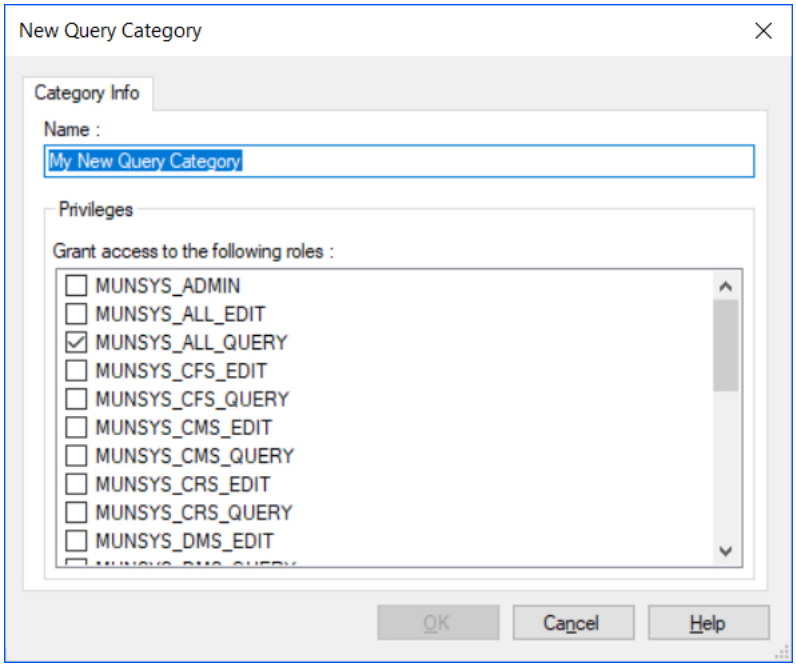


Figure 3 The New Query Category dialog box

2 On the **New Query Category** dialog box, do the following:

- Specify a name for the new query category, with a maximum of 40 characters. Query categories are uniquely named in a schema; a new query category name must therefore not

already exist in the MUNSYS_QUERY_CAT table. The name may also not begin or end with the characters << or >>, as this sequences denotes system reserved names.

- Grant access to the query category by selecting the appropriate Munsys role(s) from the list that is provided. The MUNSYS_ALL_QUERY role is granted access to a new query category by default.

3 Click **OK** to create the new query category.

The new query category is saved in the MUNSYS_QUERY_CAT table.

Renaming a query category

To be able to rename a query category, you need to have the following roles/privileges assigned:

- Munsys role: MUNSYS_POWER or MUNSYS_ADMIN
- Database privileges: UPDATE on MUNSYS_QUERY_CAT

To rename query category

1 In the **Query Categories** content pane, right-click on the category that you want to rename, and then select the **Rename** option on the context-sensitive menu, as seen below:

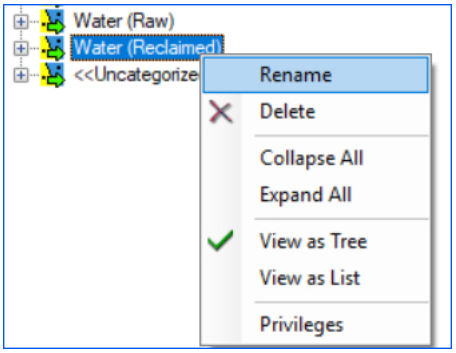


Figure 4 Rename Query Categories

2 Specify a new name for the query category, with a maximum of 40 characters. Query categories are uniquely named in a schema; a new query category name must therefore not already exist in the MUNSYS_QUERY_CAT table. The name may also not begin or end with the characters << or >>, as this sequences denotes system reserved names. If the new name that you want to give to the query already exists, an error message to this effect is displayed.

Changing access to a query category

With this function, you can view or modify the access to a selected query category. You will need to have the following roles/privileges assigned:

- Munsys role: MUNSYS_POWER or MUNSYS_ADMIN
- Database privileges: UPDATE on MUNSYS_QUERY_CAT, MUNSYS_PRIV

To change access to a query category

1 In the **Query Categories** content pane, right-click on the category that you want to modify, and then select the **Privileges** option on the context-sensitive menu.

The Query Category Privileges dialog box is displayed.

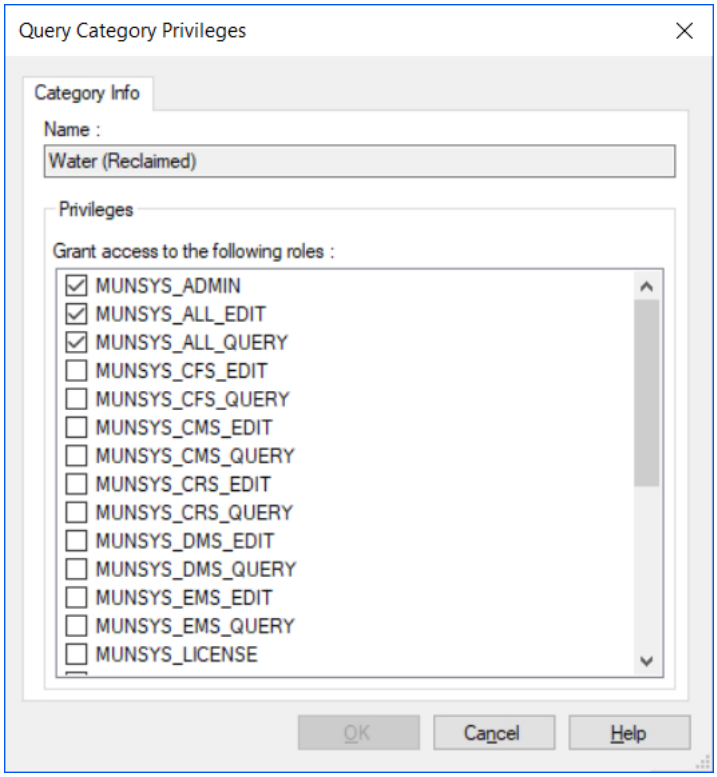


Figure 5 The Query Category Privileges dialog box

2 On the **Query Category Privileges** dialog box, change the access to the query category by selecting or clearing the check boxes adjacent to the roles displayed in the list.

3 Click **OK** to save the changes to the database.

Deleting a query category

When a query category is deleted, all the references to the queries in that category are removed and the queries are marked as Uncategorized. To be able to delete a query category, you need to have the following roles/privileges assigned:

- Munsys role: MUNSYS_POWER or MUNSYS_ADMIN
- Database privileges: UPDATE on MUNSYS_QUERY_CAT, MUNSYS_PRIV

To delete a query category

1 In the **Query Categories** content pane, right-click on the category that you want to delete, and then select the **Delete** option on the context-sensitive menu.

The Delete Query Category confirmation dialog box is displayed:

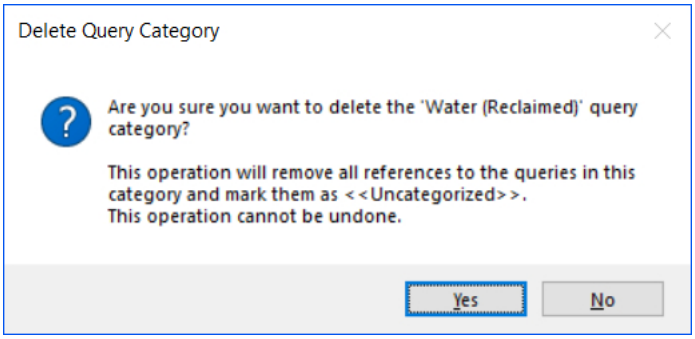


Figure 6 The Delete Query Category dialog box

2 Click **Yes** on the **Delete Query Category** confirmation dialog box to delete the query category.

The query category is removed from the MUNSYS_QUERY_CAT table. Any reference between the queries and the category are removed from the MUNSYS_PRIV table, and any queries that belonged to the category are moved to the <<*Uncategorized*>> category.

Working with queries in a query category

The Queries item in the Query Categories tree of the Munsys Management Console is used to maintain queries within a query category. The queries can be viewed as either a tree or a list. The view is set by right-clicking on Queries, and then selecting the View as Tree or View as List option.

- The View as Tree option displays the queries in a tree control, with the top level of the tree containing the query category and the sub-level containing queries that belong to the various categories.

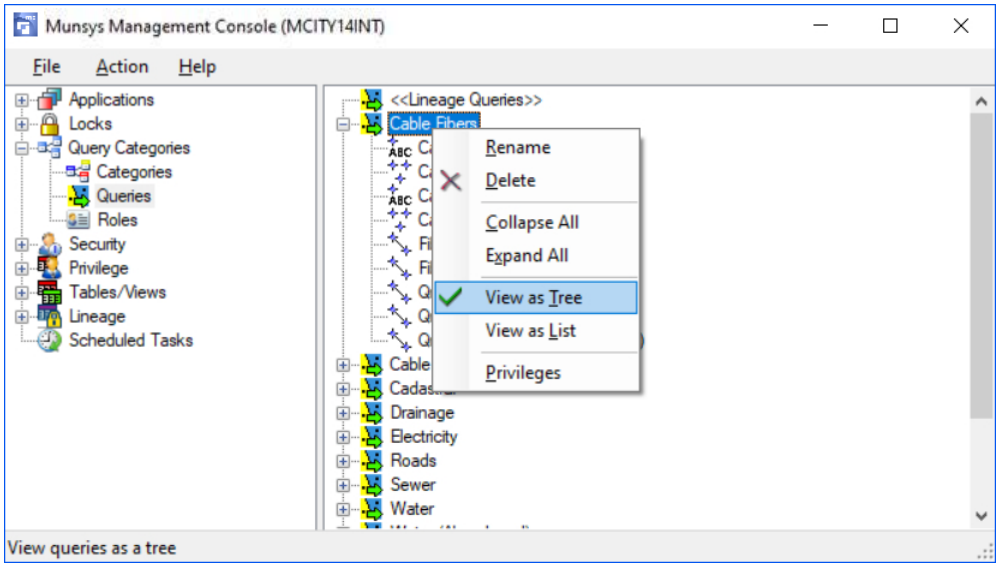


Figure 7 View queries as tree

- The View as List option displays a sortable multi-column list.

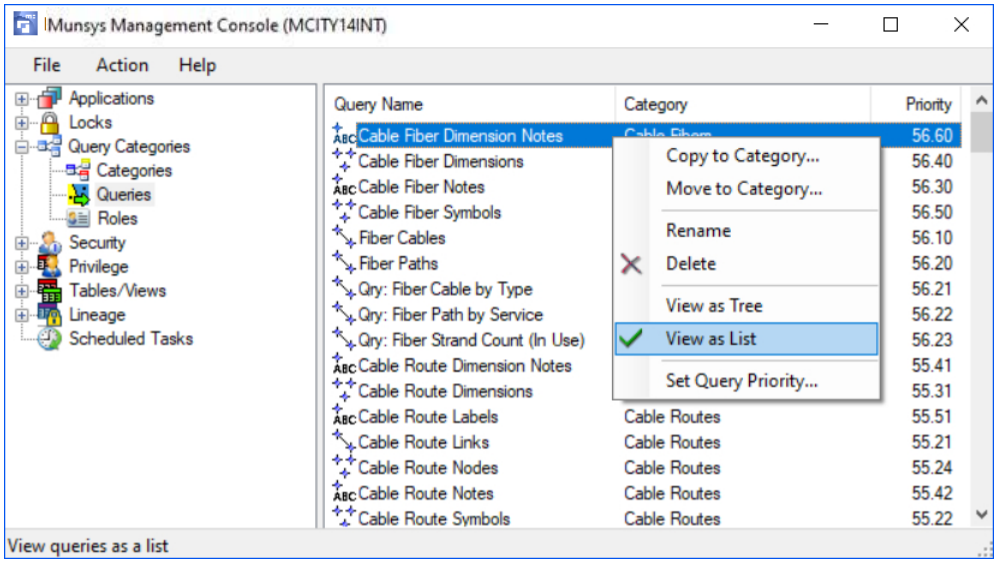


Figure 8 View queries as list

Using the context-sensitive (right-click) menu, queries can be:

- Copied to other categories
- Moved to another category
- Renamed
- Deleted
- Assigned a query priority

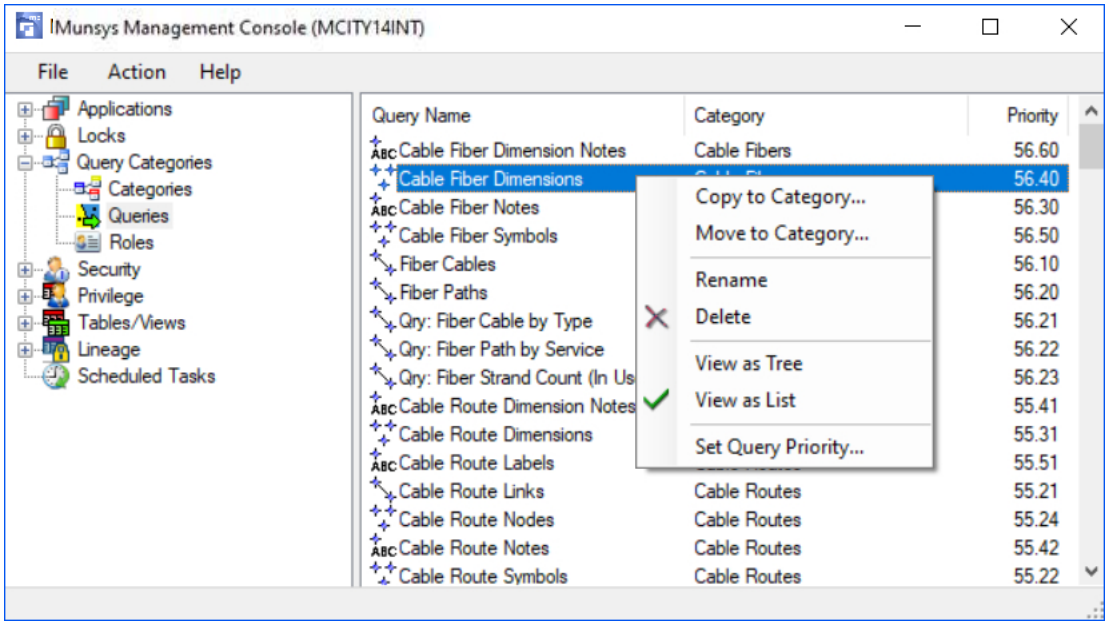


Figure 9 View query options

Copying a query to one or more categories

1 Right-click on the query that you want to copy, and then select **Copy to Category...** from the context-sensitive menu.

Tip If you are copying a query to the same category it is currently associated with, you will have the option to replace the query, which retains the QRY_ID of the original query, but replaces the content of the query in the MUNSYS_QUERY table.

The **Copy Query** dialog box is displayed, showing the name of the query that you have selected.

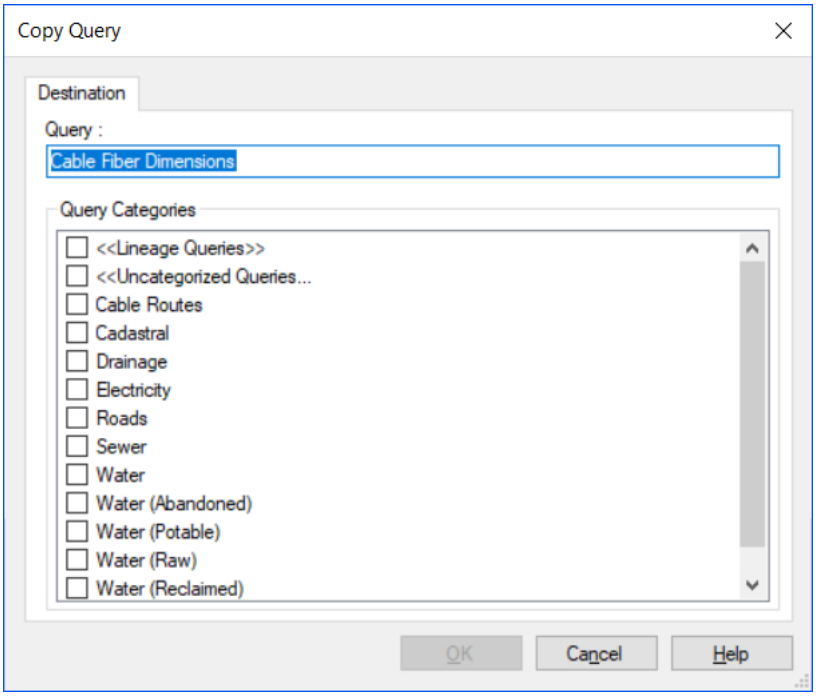


Figure 10 The Copy Query dialog box

2 Change the query name if required.

3 From the **Query Categories** list, select the category(ies) that you want the query copied to by checking the boxes, and then click **OK**.

A copy of the query is created for each of the categories that you selected. The query name remains the same, but a new QRY_ID is assigned for each copy.

Tip

Moving a query to another category

When a query is moved to another category, the QRY_ID and the query record in the MUNSYS_QUERY table stay the same, but the MUNSYS_QUERY_CAT table is updated to change the category reference to the query.

To move a query to another category, do the following:

1 Right-click on the query that you want to move, and then select **Move to Category...** from the context-sensitive menu.

The Move Query dialog box is displayed, with the current category of the query excluded.

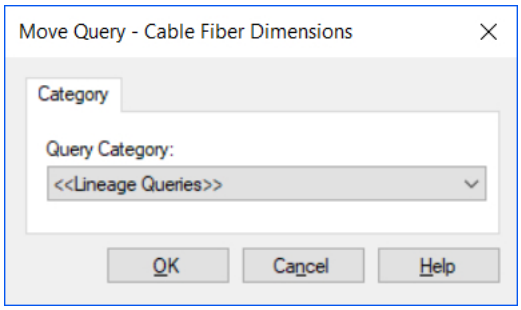


Figure 11 The Move Query dialog box

2 On the **Move Query** dialog box, select the category that you want to move the query to from the **Query Category** list.

The query is moved to the selected category. If a query with the same name already exists within the new category, an error message will be displayed, and you will not be able to move the query.

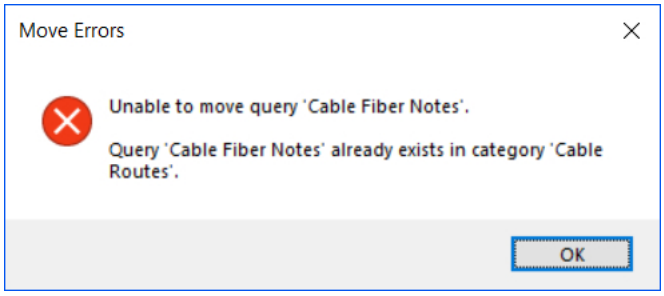


Figure 12 Move Query Warning Message

Renaming a query

1 Right-click on the query that you want to rename, and then select **Rename** from the context-sensitive menu.

2 Enter a new name for the query.

The query is renamed and the record in the MUNSYS_QUERY table is changed accordingly. If a query with the same name already exists within the category, an error message will be displayed, and you will not be able to rename the query.

Deleting a query

Right-click on the query that you want to delete, and then select Delete from the context-sensitive menu.

The following message is displayed:

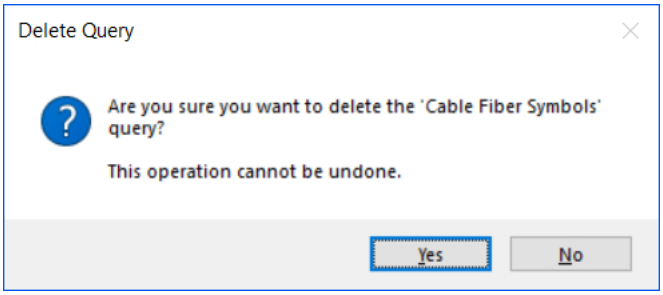


Figure 13 Delete Query warning message

3 Click **Yes** to delete the query.

The query is deleted from the MUNSYS_QUERY table, and the association with the query category is deleted from the MUNSYS_QUERY_CAT table.

Tip If you want to remove a query from a query category, but do not want not delete the query from the database, use the **Move to Category...** item on the context-sensitive menu.

Setting a query priority

The query priority of any query determines which objects will be queried on top of others. You have to have the MUNSYS_POWER role assigned to be able to set a query priority.

1 Right-click on the query for which you want to set/change the priority, and then select **Set Query Priority...** from the context-sensitive menu.

The Query Priority dialog box is displayed.

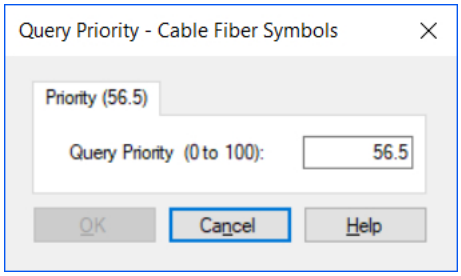


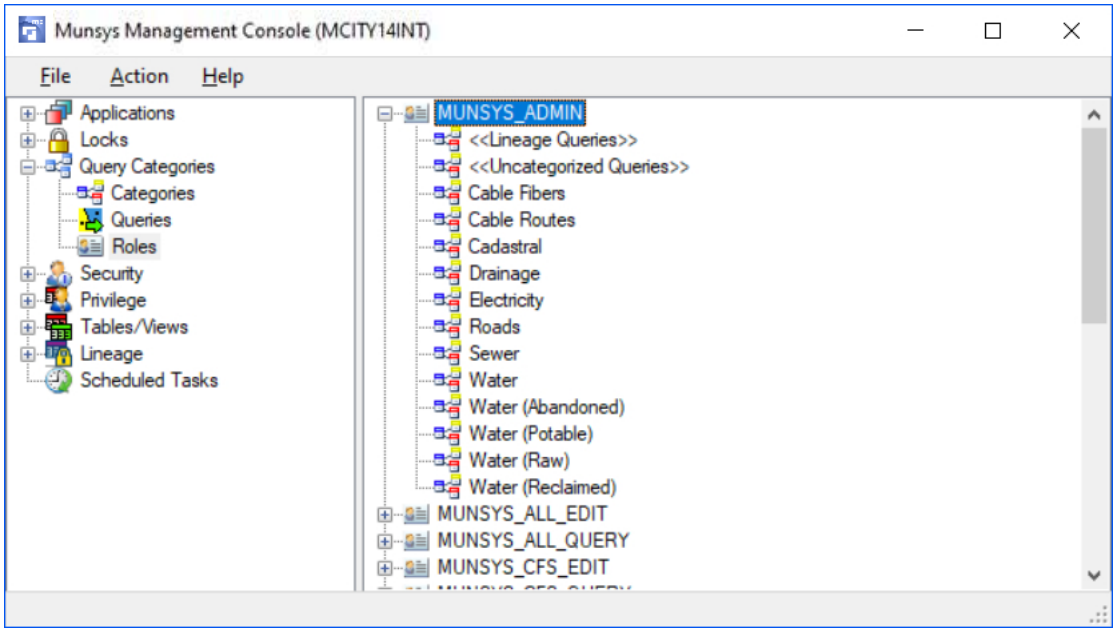
Figure 14 The Query Priority dialog box

2 Enter a value between 0 and 100. Clicking on the arrows will increment the priority by a whole number; decimal numbers have to be entered manually. A low number means that the query will be executed before queries with higher numbers. This means that queries with a higher number will be executed towards the end of the query process, causing the objects with a higher priority to be placed on top of the objects queried from queries with a low query priority.

3 Click **OK** to assign the query priority to the selected query.

Working with query category roles

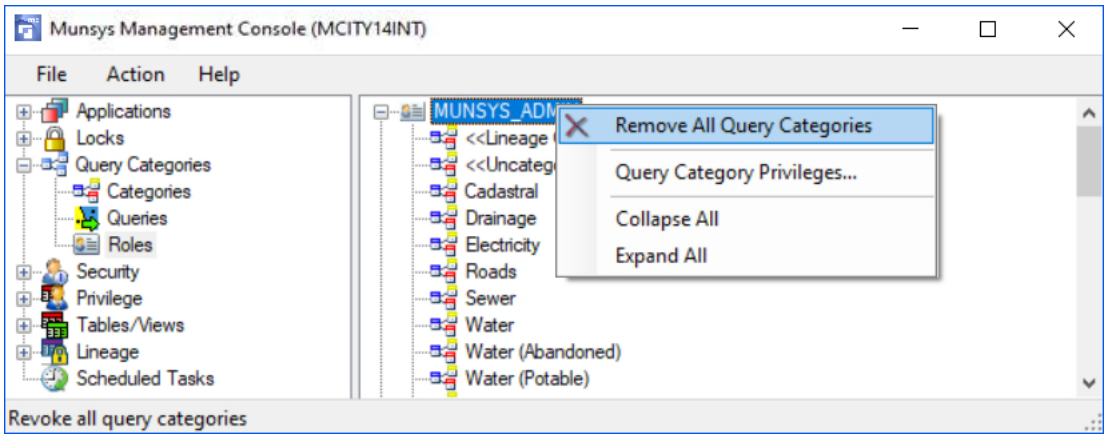
The Roles tree item in the Query Categories tree of the Munsys Management Console is used to maintain access privileges to query categories. When the Roles tree item is activated, all Munsys roles (MUNSYS_) are displayed in a tree view control in the content pane. The tree contains all the available MUNSYS_roles as the top level nodes, and the query categories as the sub-level nodes.



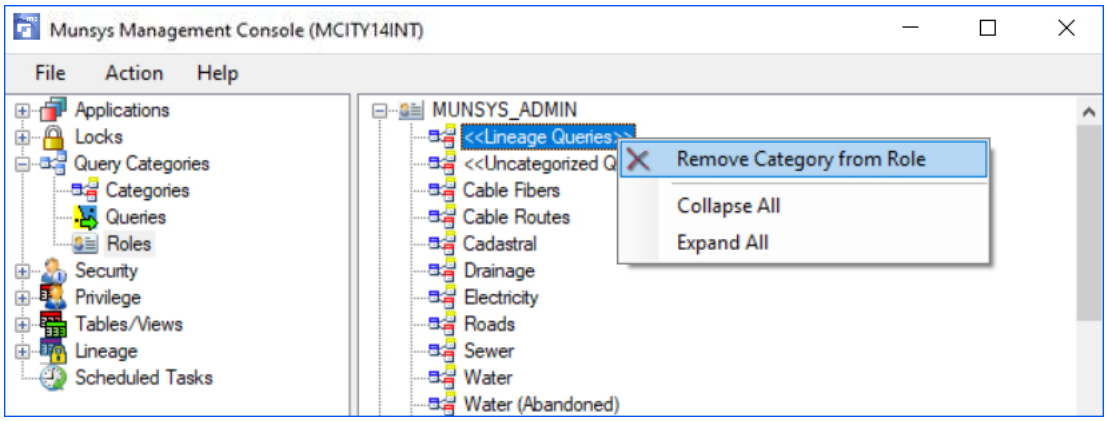
When you right-click on a role in the content pane, the context-sensitive menu can be used to change the privileges for accessing categories by:

- Removing access to query categories from a role – revokes access from a role so that it no longer has access to any query categories.

- Granting or revoking access of a role to multiple query categories – changes the access that a particular role has with regard to various query categories.



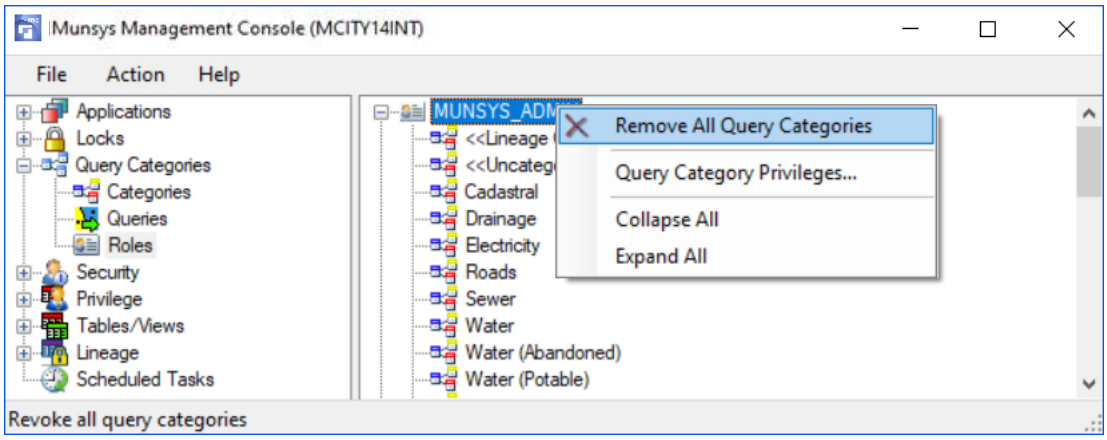
- When you right-click on a query category assigned to a role in the content pane, you can remove the access from the role to which it has been assigned using the Remove Privilege from Role context-sensitive menu item. You will be prompted for confirmation before access is removed for the role.



Removing all query categories from a role

With this function, you can revoke access to query categories from a selected Munsys role. When this is done, all the entries associated with the role are removed from the MUNSYS_PRIV table.

1 Right-click on the appropriate role in the content pane, and then select **Remove All Query Categories** from the context-sensitive menu.



The following message is displayed:

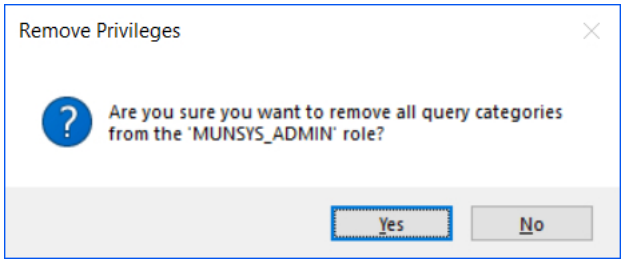
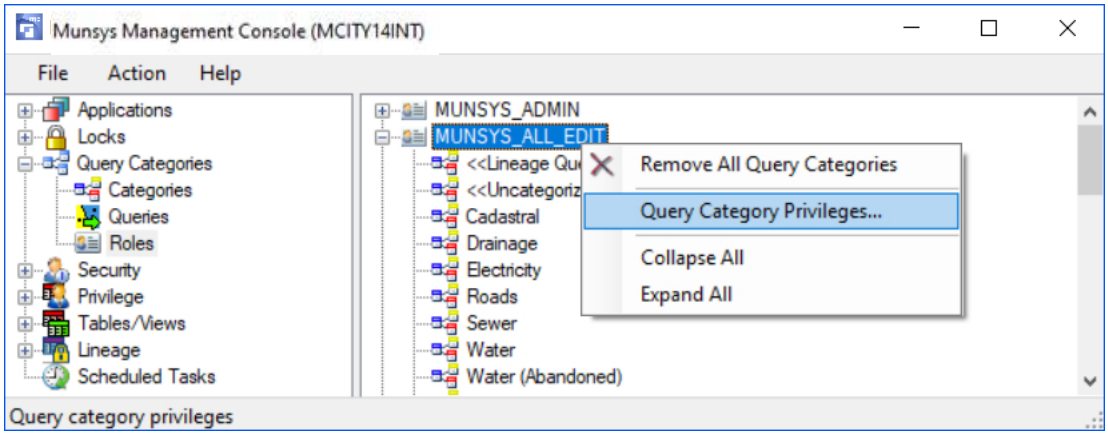


Figure 15 The Remove Privileges dialog box

2 Click **Yes** to remove access to all the query categories from the role.

Granting or revoking access of a role to multiple query categories

1 Right-click on the appropriate role in the content pane, and then select **Query Category Privileges**... from the context-sensitive menu.



The Query Category Privileges dialog box is displayed, showing the query categories to which access has already been granted for the role, as well as the query categories that are available to the role.

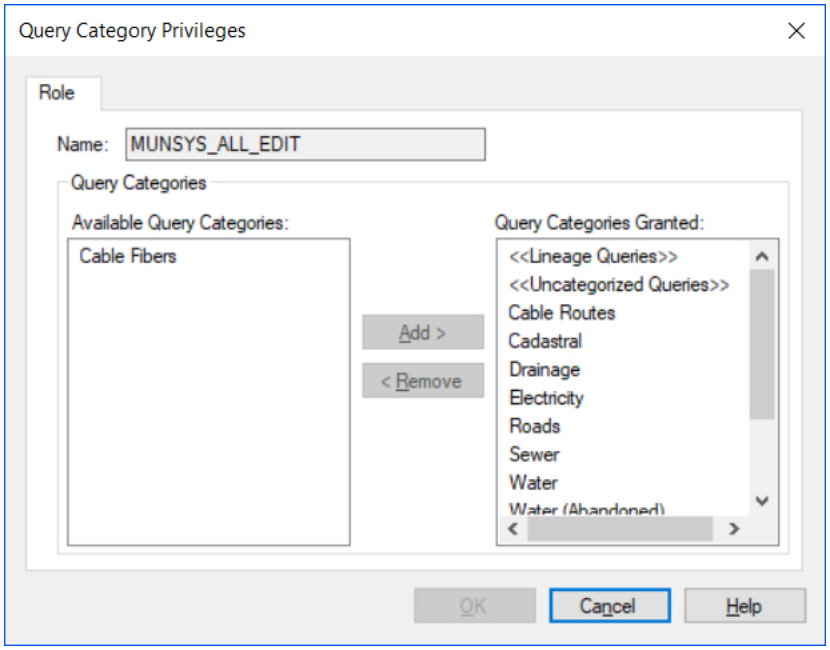


Figure 16 The Query Category Privileges

2 Add or remove the query categories to the role as required.

3 Click **OK** to apply any changes that you have made.

The MUNSYS_PRIV table is updated accordingly.

Chapter 8
Security

Introduction

In the Munsys Management Console, security management consists of the creation and maintenance of database users and roles. Database users and roles are managed on the Security tree. Roles place users in a "category" where certain responsibilities are associated with these roles. Depending on the role of a user, varying components and/or menu items in the console will be available for manipulation. The Munsys Management Console recognizes the following roles:

- *MUNSYS_ADMIN – this role is reserved for users who have the authority to change structure such as creating, dropping or validating schemas, tables or views, re-indexing tables, overriding object locking status, or manipulating database users and the roles or privileges that may be assigned to them.
- MUNSYS_POWER – this role is reserved for users who have the authority to change the content of items such as rules, lookup tables and application settings.
- MUNSYS_APP_EDIT – users with this role assigned can create and modify objects in a specified Munsys application only (where APP is the relevant application, for example MUNSYS_CDS_EDIT or MUNSYS_DRN_EDIT)
- MUNSYS_ALL_EDIT – users with this role assigned can create and modify objects in all of the Munsys applications
- MUNSYS_APP_QUERY – users with this role assigned can query objects from a specific Munsys application only (where APP is the relevant application, for example MUNSYS_CDS_QUERY or MUNSYS_DRN_QUERY)
- MUNSYS_ALL_QUERY – users with this role assigned can query objects from all the Munsys applications

Note Having the **MUNSYS_ADMIN** role assigned is not enough to create and manipulate objects; Oracle expects other privileges to also be assigned to a user as well before objects such as tables, etc. can be created or manipulated. The database administrator has the ability to still restrict other privileges in respect of a user with the **MUNSYS_ADMIN** role assigned.

The Security tree: overview

The Security tree consists of two branches (nodes):

1 **Users** – used for the creation and maintenance of users in the database. When you click on **Users** in the **Security** tree, the content pane displays the relevant user name, with all the assigned Munsys roles as sub-items. Right-clicking on an item displays a context-sensitive menu, from which user maintenance can be done.

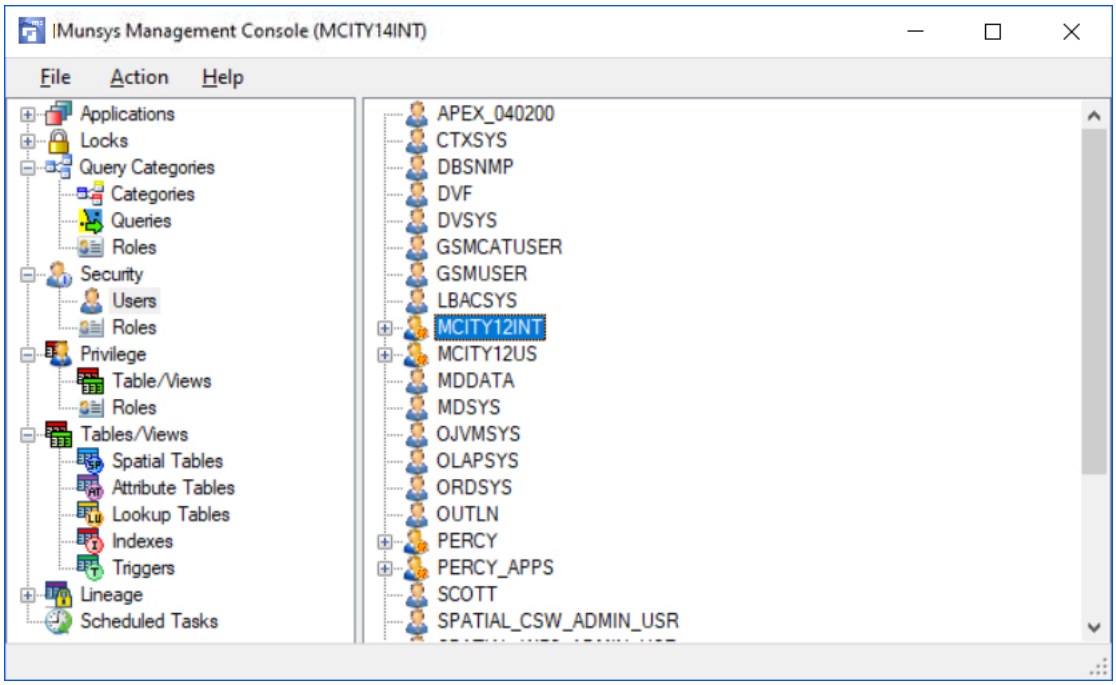
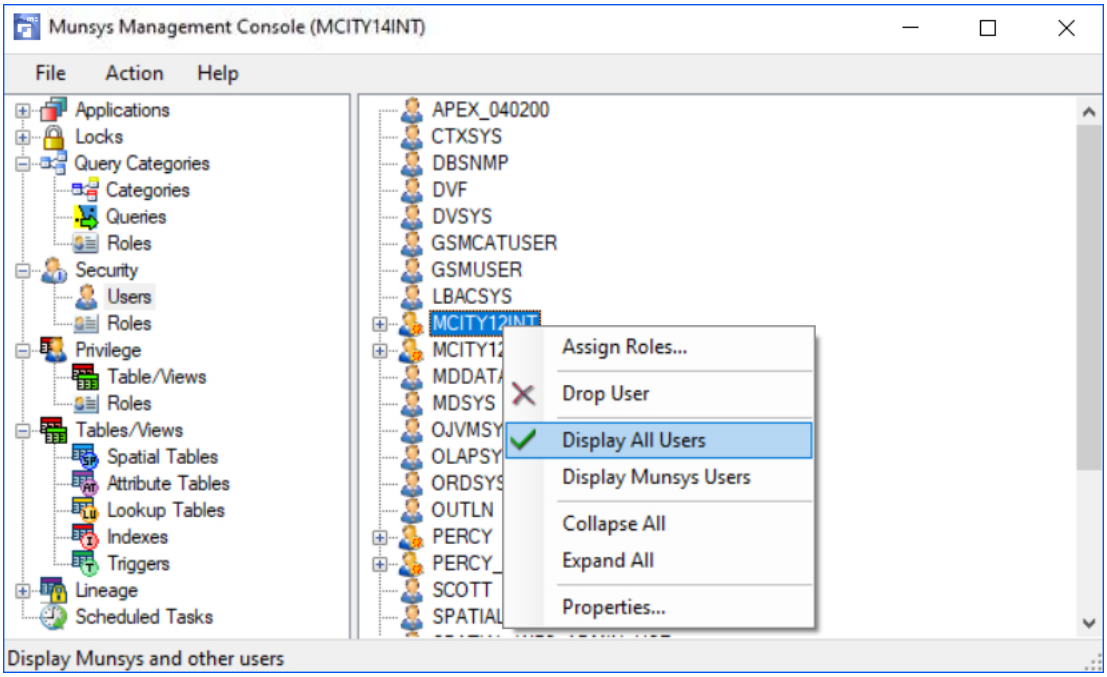


Figure 1 Security tree: Users

Note You have the option of displaying all the database users, or only the Munsys users. This is set by selecting a user, and then selecting the **Display all Users** or **Display Munsys Users** option on the **Action** menu or the context-sensitive (right-click) menu.



2 **Roles** – used for maintenance of the database roles and the creation of Munsys roles. When you click on **Roles** in the **Security** tree, the content pane displays all the database roles, with their assigned users as sub items. Right-clicking on an item displays a context-sensitive menu, from which role maintenance can be done.

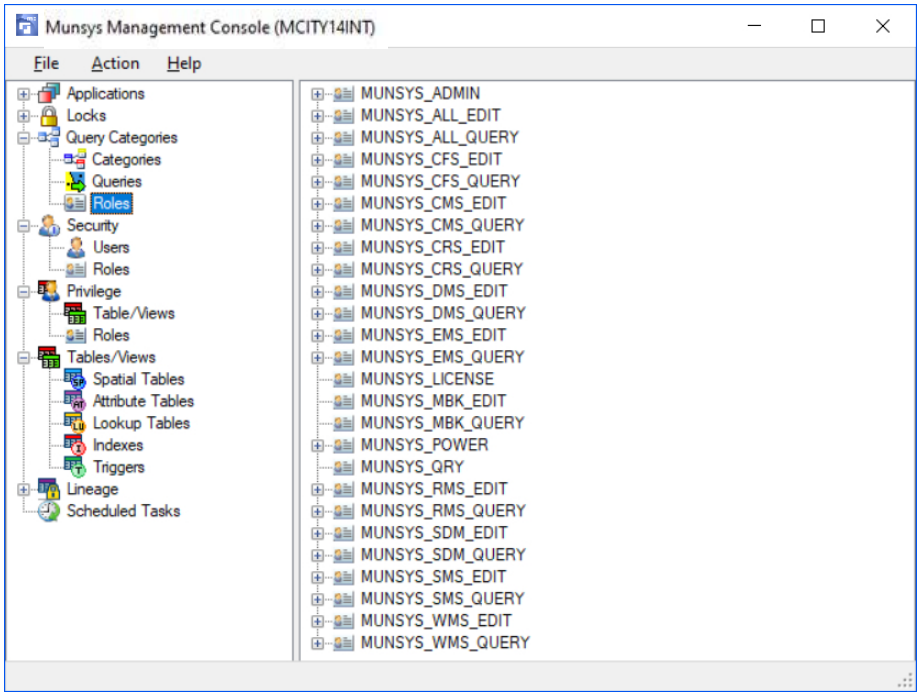
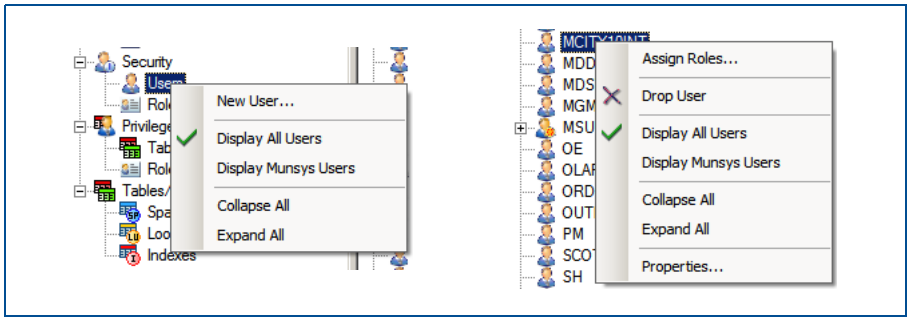


Figure 2 Security tree: roles

Working with Munsys users

Using the context-sensitive (right-click) or Action menu, you can create a new user, assign roles to a user, drop a user, change the role assignment of a user or revoke a role from a user.



Creating a new Munsys user

When a new Munsys user is created, the MUNSYS_ALL_QUERY and CONNECT roles are assigned by default.

1 To create a new Munsys user, do one of the following:

- Click on **Users** in the **Security** tree, and then select **New User...** from the **Action** menu
- Right-click on **Users** in the **Security** tree, and then select **New User...** from the context-sensitive menu.

The User Details dialog box is displayed, showing the User Details tab.

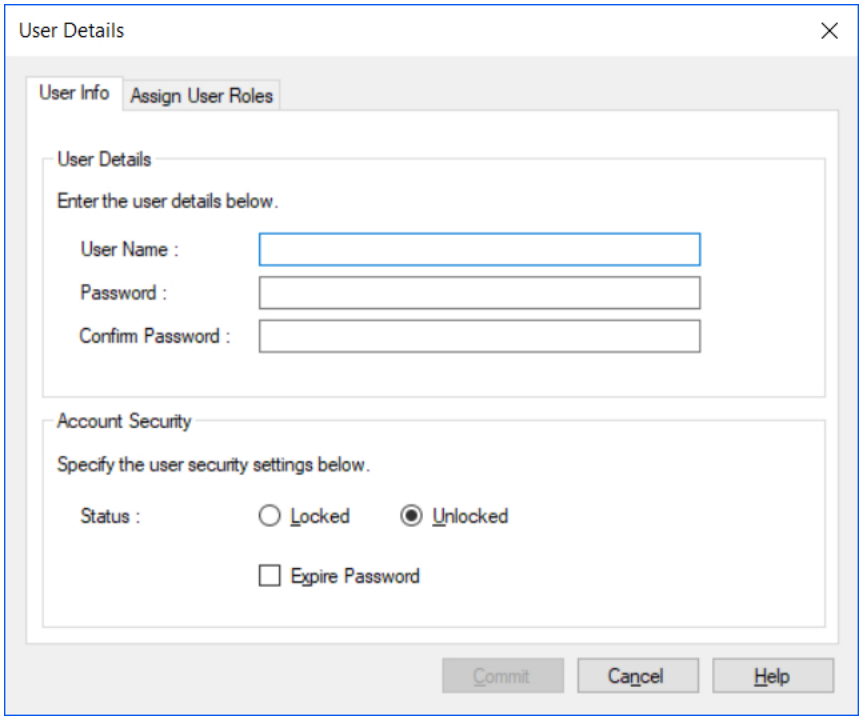


Figure 3 The User Details dialog box: User Details tab

2 Under **User Details**, enter the following information for the new user:

- the user name
- password
- password confirmation

3 Under **Account Security**, you can lock a user, or expire a user's password.

4 Click the **Assign User Roles** tab to assign Munsys roles to the new user. The **MUNSYS_ALL_QUERY** role is assigned by default.

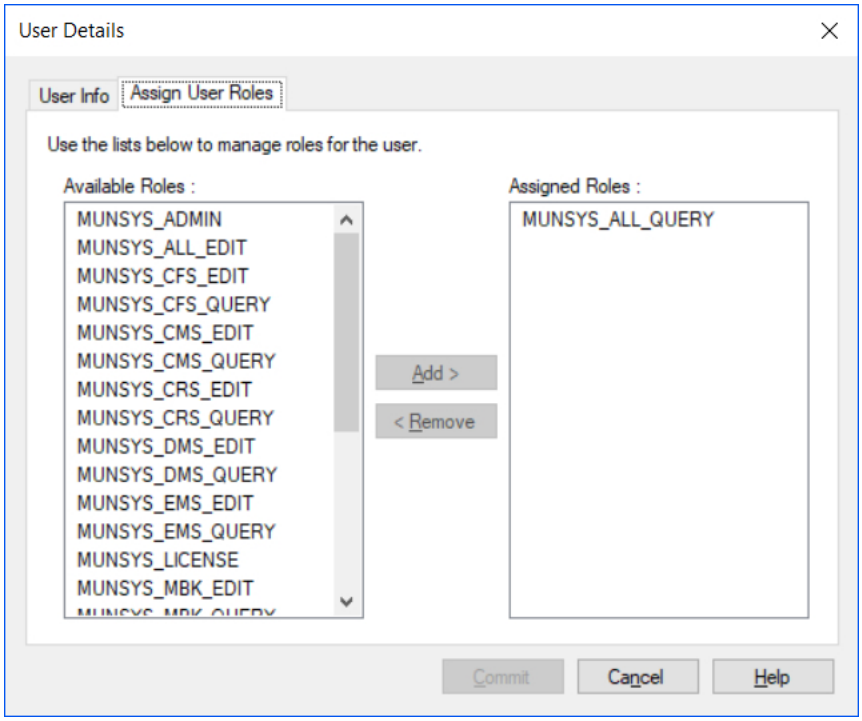


Figure 4 The User Details dialog box: Assign User Roles tab

5 Select the check box adjacent to the role that you want to add or remove, and then use the **Add** or **Remove** buttons as required.

When you have specified all the details for the new user, click Commit to save the new user to the database.

Changing user properties

Using the Properties option on the context-sensitive menu or the Action menu when a user name is selected, the password, security settings and role allocation for a user can be changed.

To change user properties, do the following:

1 Select a user name in the content pane, and then choose either **Action** > **Properties**, or the **Properties** option on the context-sensitive menu.

2 The **User Details** dialog box is displayed, showing the **User Details** tab.

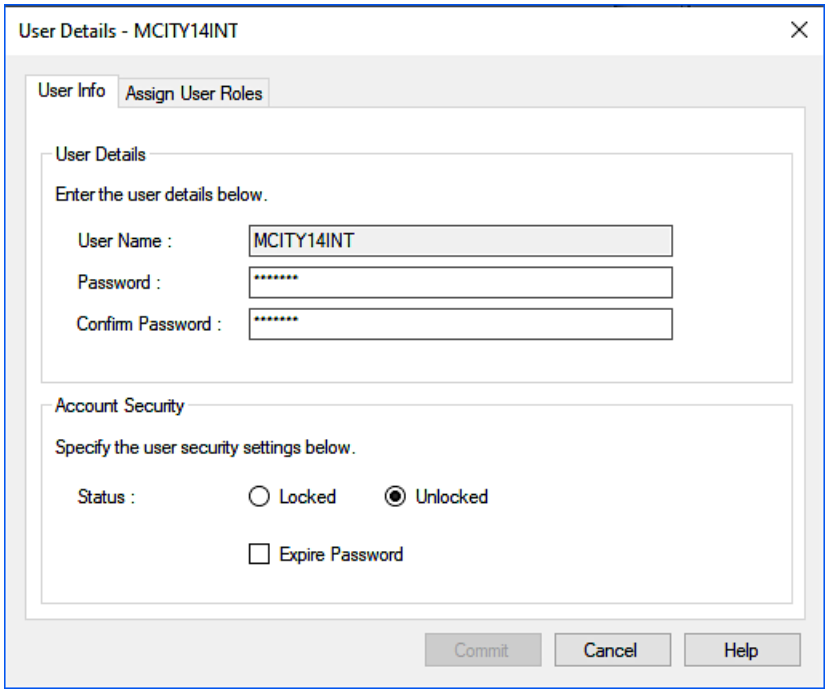


Figure 5 The User Details dialog box: Properties

3 Change the user details or account security as required.

4 On the **Assign User Roles** tab, add or remove roles from the user.

5 Click **Commit** to save the changes to the database.

Tip Selecting the **Assign Roles** option from the context-sensitive or the **Action** menu when a user is selected directly displays the **Assign User Roles** tab.

Dropping a user

To drop a user do the following:

1. Right-click on the user name, and then select the **Drop User** option from the context-sensitive menu, or select the **Drop User** menu item from the **Action** menu.
2. The following confirmation message is displayed:

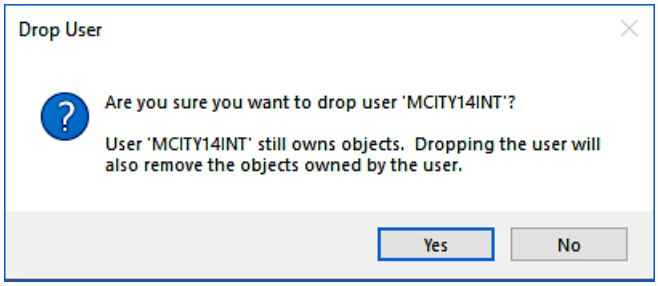


Figure 6 Drop user confirmation message

Working with Munsys roles

In the Munsys Management Console, Munsys roles are created and maintained from the Roles item in the Security tree. You have the option of displaying all the database users in the content pane, or only the Munsys users. This is set by selecting a user, and then selecting the Display all Users or Display Munsys Users option on the Action menu or the context-sensitive (right-click) menu.

Creating a new Munsys role

To create a new Munsys role, do the following:

1. Select **Roles** in the **Security** tree, and then select the **New Munsys Role** option on the **Action** menu, or right-click on **Roles** in the **Security** tree, and then select the **New Munsys Role** option from the context-sensitive menu.

   The New Munsys Role dialog box is displayed.

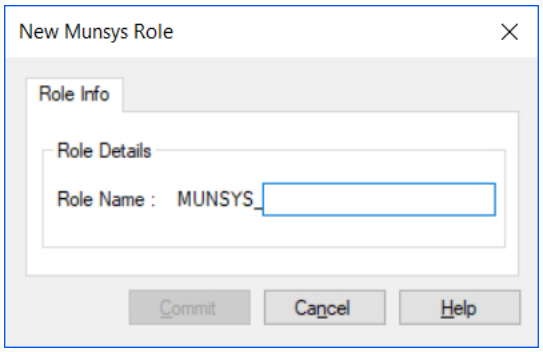


Figure 7 The New Munsys Role dialog box

2. On the **New Munsys Role** dialog box, enter a name for the role, and then click **Commit** to save the new Munsys role to the database.

Dropping a role from the database

To drop a role from the database, do the following:

1. Right-click on the role that you want to drop in the content pane, and then select the **Drop Role** option from the context-sensitive menu, or select the **Drop Role** menu item from the **Action** menu.
2. The following confirmation message is displayed:

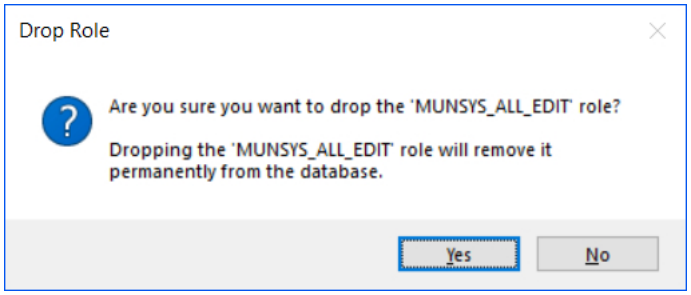


Figure 8 Drop Role confirmation message

3. Click **Yes** to drop the role from the database.

Revoking a role from a user

To revoke a role from a user, do the following:

1. Right-click on a user name in the content pane, and then select the **Remove** option from the context-sensitive menu, or select the **Remove** option from the **Action** menu.
2. The following confirmation message is displayed:

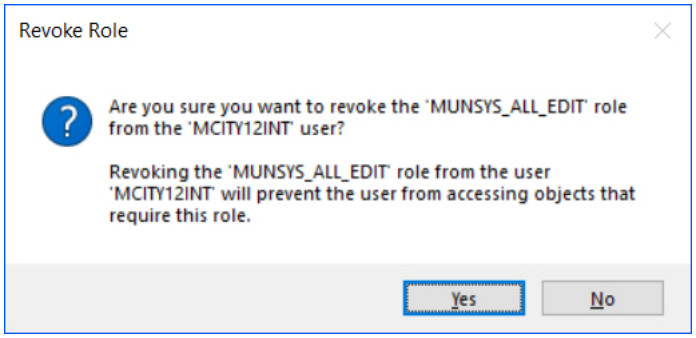


Figure 9 Revoke Role confirmation message

3. Click **Yes** to remove the role from the user.

Chapter 9
Privilege

Introduction

In the Munsys Management Console, privilege management consists of granting or revoking privileges on database objects to the users. The administrator can grant or revoke privileges assigned to the Munsys roles on a table or view and visa-versa.

The Munsys roles are a collection of privileges or access rights to the various tables and views in the Munsys schema. The administrator assigns a user to Munsys role and then assigns the role to a spatial table or spatial view. They then grant the role a combination of SELECT, INSERT, DELETE and/or UPDATE privileges for that spatial table or spatial view.

- SELECT – roles with this privilege can select data from a database object. This is used when users run any of the queries to which they have access in the Munsys applications.
- INSERT – roles with this privilege can insert rows into a spatial, attribute or lookup table. This is used when users post new objects to the database or when a user with the MUNSYS_POWER role edits the lookup table values in the Munsys Management Console application.
- DELETE – roles with this privilege can delete objects from the table or view.
- UPDATE – roles with this privilege assigned can modify existing data in the Munsys tables. This is used when using any of the change commands in the Munsys applications.

The privilege tree: overview

The Privilege tree consists of two branches (nodes):

1 **Tables/Views** – used for assigning roles to the tables and views. When you click on **Tables/Views** in the **Privilege** tree, the content pane displays all the table and views in the schema. Right-clicking on an item displays a context-sensitive menu, where the administrator can assign a role to a table or view.

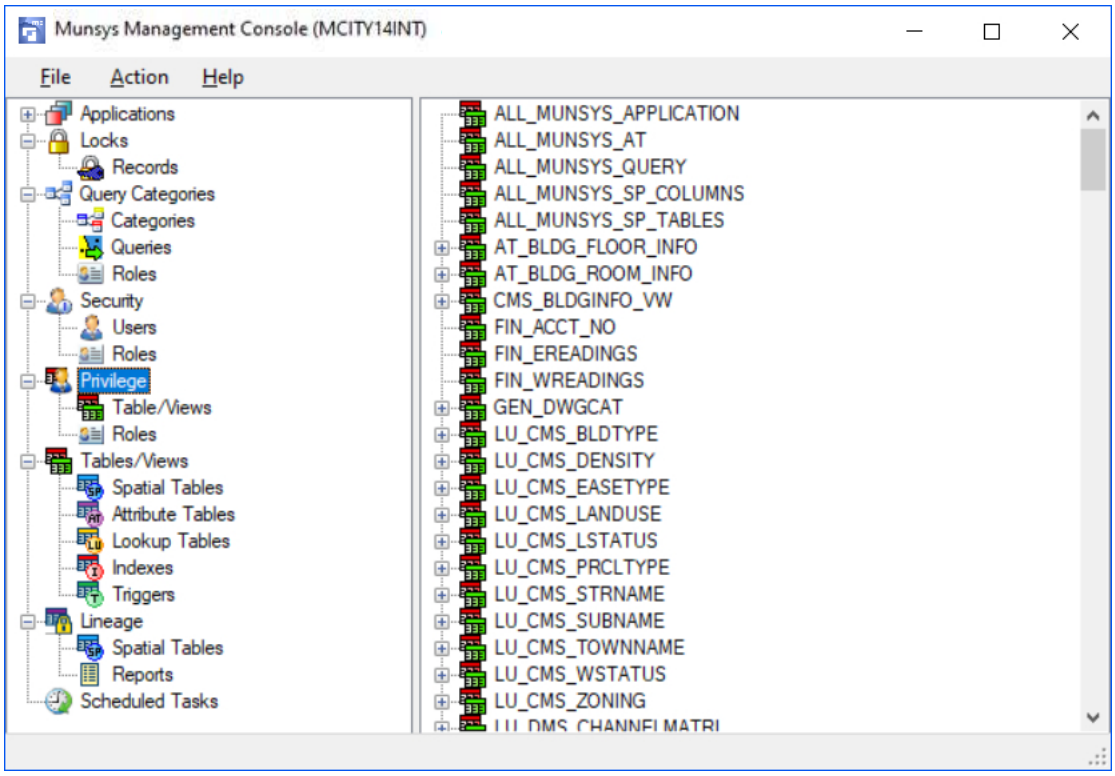


Figure 1 Privilege tree: Tables/Views

2 **Roles** – used for maintenance of tables and views assigned to Munsys roles. When you click on **Roles** in the **Privilege** tree, the content pane displays all the Munsys database roles. Right-clicking on an item displays a context-sensitive menu, where the administrator can assign a table or view to a role.

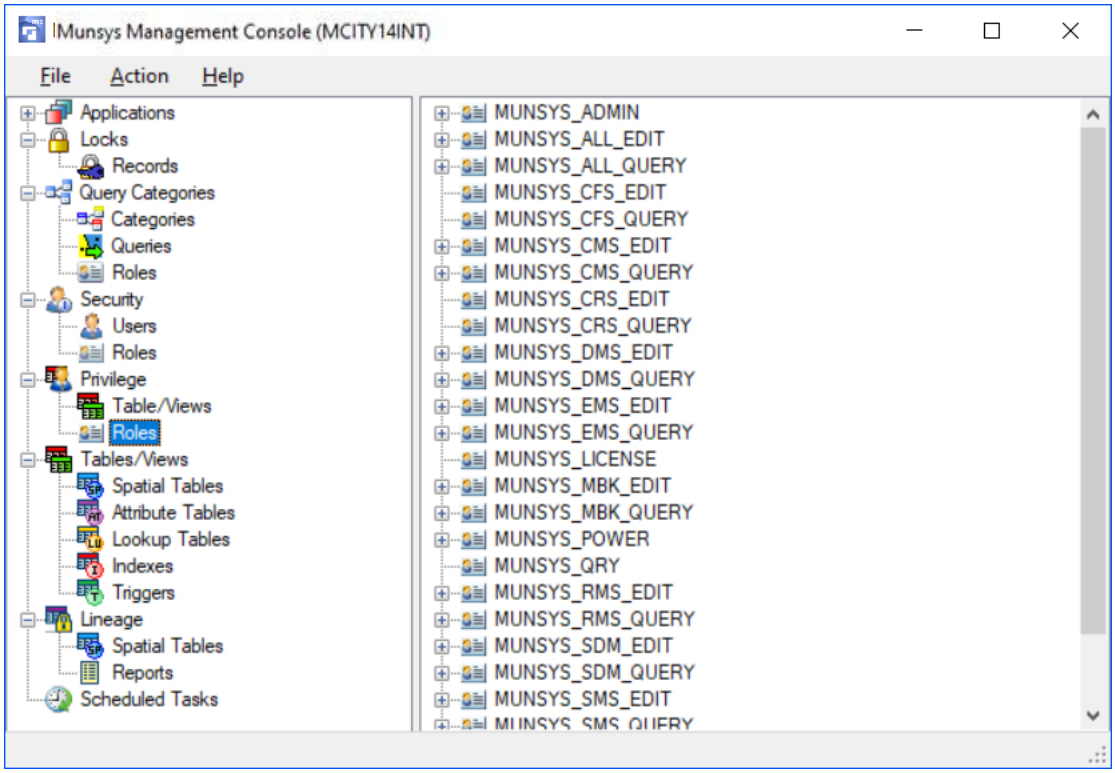
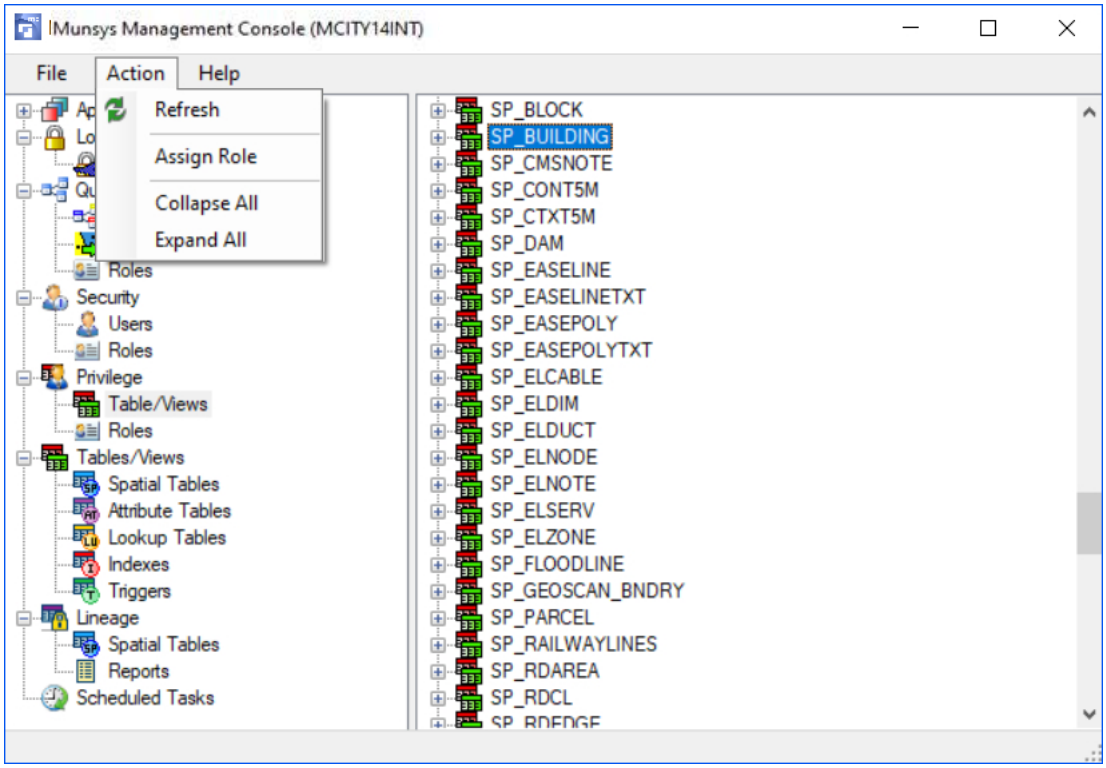


Figure 2 Privilege tree: roles

Working with Tables and Views

Using the context-sensitive (right-click) or Action menu, you can assign a role to a table or view.



Assigning a role to a table or view

When a new Munsys table is created using the Munsys Management *Tables/Views* > *Create New Table* option, the following roles are automatically assigned to the new table:

- **MUNSYS_ADMIN**
- **MUNSYS_ALL_EDIT**
- **MUNSYS_ALL_QUERY**
- **MUNSYS_POWER**
- **MUNSYS_SDM_EDIT**
- **MUNSYS_SDM_QUERY**

Should the administrator wish to make the table available to other Munsys applications for querying or editing, they will need to assign the Munsys roles for the various applications to the table or view.

1 To assign a role to a table do the following:

- Click on **Tables/Views** in the **Privilege** tree and the content pane is populated with a list of all tables and views in the schema.
- In the content pane select the table or view to which you want to assign the role to.
- Right-click on table name in the content pane and select the **Assign Role** option from the context-sensitive menu.

Note The administrator has an option of using the right click context-sensitive menu **Assign Role** to assign the role or they can select the table or view in the content pane and select **Action > Assign Role** from the main menu.

2 The **Assign Role** dialog box is displayed showing the **Assign Role** tab with the table or view name in the dialog box header.

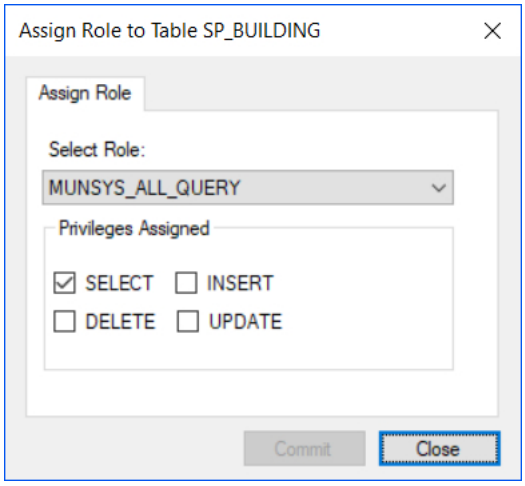


Figure 3 The Assign Role dialog box: Assign Role tab

3 From the **Select Role** drop down list select the role to be assigned to the table or view.

4 Check the tick boxes next to the privileges you want to assign to the role on the table or view.

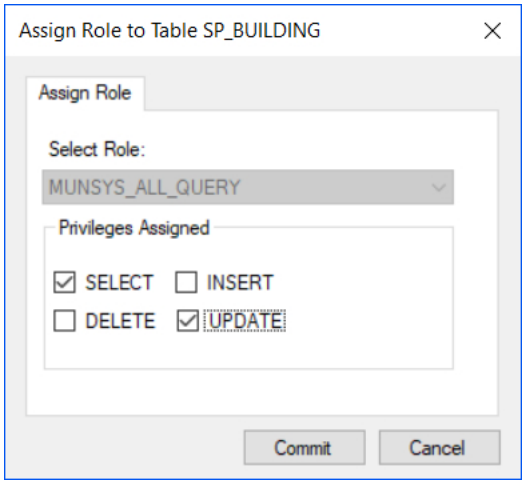


Figure 4 The Assign Role dialog box: Privileges Assigned

5 Select the **Commit** button to save the changes to the database.

This selection process must be repeated for all roles added to the table or view.

Note At least one privilege must be assigned to the role to activate the **Commit** button.

Removing a role from a table or view

Should the administrator wish to remove the table from the Munsys applications for querying or editing, they will need to remove the Munsys roles for the various applications to the table or view.

1 To remove a role assigned to a table do the following:

- Click on **Tables/Views** in the **Privilege** tree and the content pane is populated with a list of all tables and views in the schema.
- In the content pane select the expand node next to the table or view whose roles you want to remove.
- Right-click on role and select the **Remove** option from the context-sensitive menu.

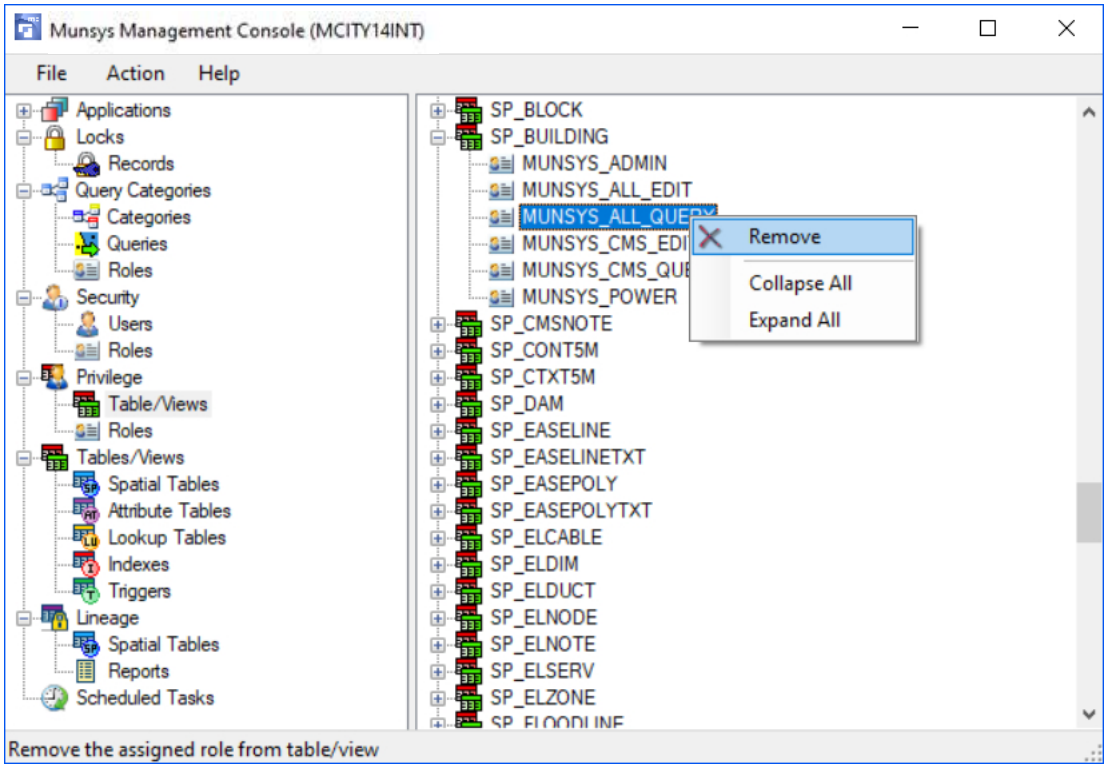


Figure 5 The Remove command in the context-sensitive menu

Note The administrator has an option of using the right click context-sensitive menu **Remove** to remove the role or they can select the role in the expanded list of roles in the content pane and select **Action > Remove** from the main menu.

2 The **Remove Role** warning message is displayed.

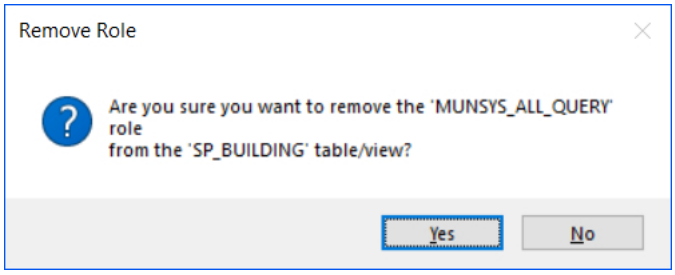
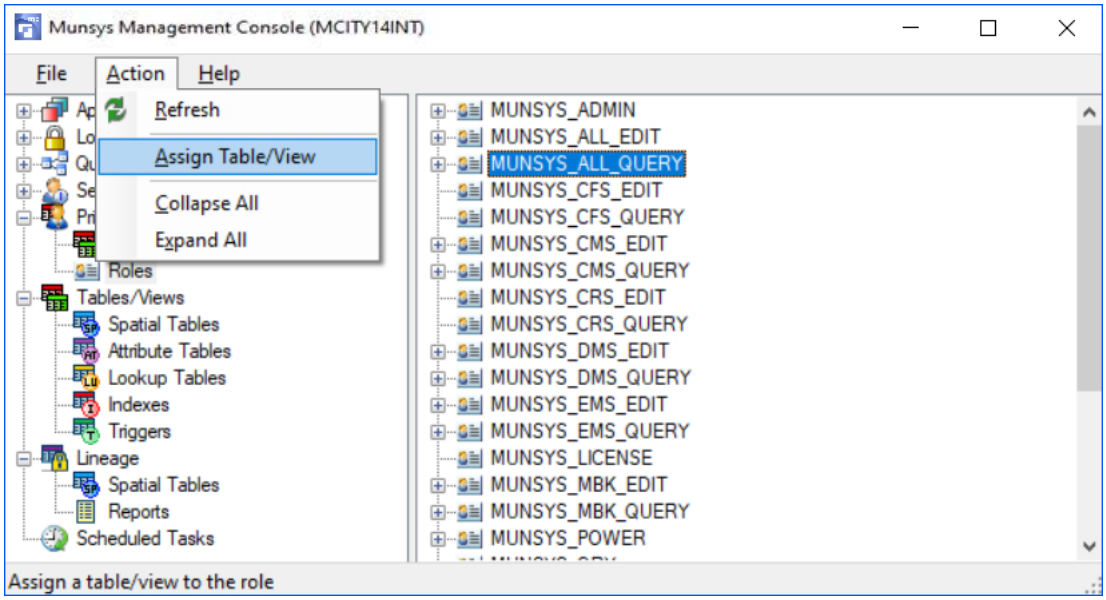


Figure 6 Remove Role warning message

3 Click **Yes** to remove the role.

Working with Roles

Using the context-sensitive (right-click) or Action menu, you can assign a table or view to a Munsys role.



Assigning a table/view to a role

1 To assign a table or view to a Munsys role do the following:

- Click on **Roles** in the **Privilege** tree and the content pane is populated with a list of all Munsys roles in the schema.
- In the content pane select the role you want to assign the table or view to.
- Right-click on the role in the content pane and select the **Assign Table/View** option from the context-sensitive menu.

The Assign Table dialog box is displayed showing the Assign Table tab with the Munsys role name in the dialog box header.

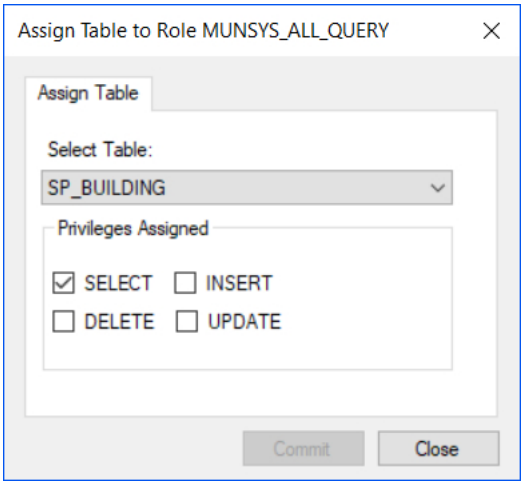


Figure 7 The Assign Table to Role dialog box: Assign Table tab

2 From the **Select Table** drop down list select the table or view to be assigned to the role.

3 Check the tick boxes next to the privileges you want to assign to the role on the table or view.

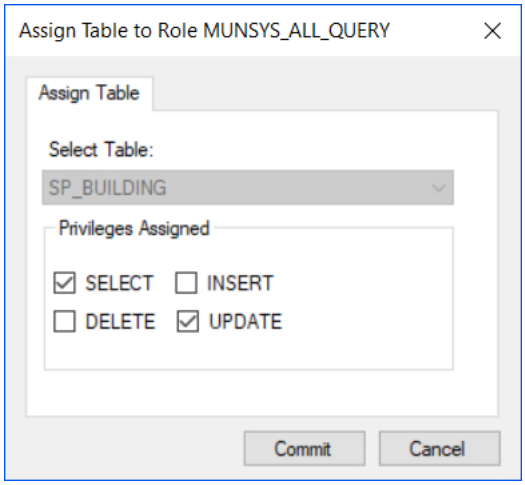


Figure 8 The Assign Table dialog box: Privileges Assigned

4 Select the **Commit** button to save the changes to the database and then click the **Close** button to close the dialog box

This selection process must be repeated for all table or views added to the Munsys roles.

Note At least one privilege must be assigned to the role to activate the **Commit** button.

Removing a table/view from a role

1 To remove a table or view from a Munsys role do the following:

- Click on **Roles** in the **Privilege** tree and the content pane is populated with a list of all Munsys roles in the schema
- In the content pane select the expand node next to the Munsys role whose table or view you want to remove.
- Right-click on the table name and select the **Remove** option from the context-sensitive menu.

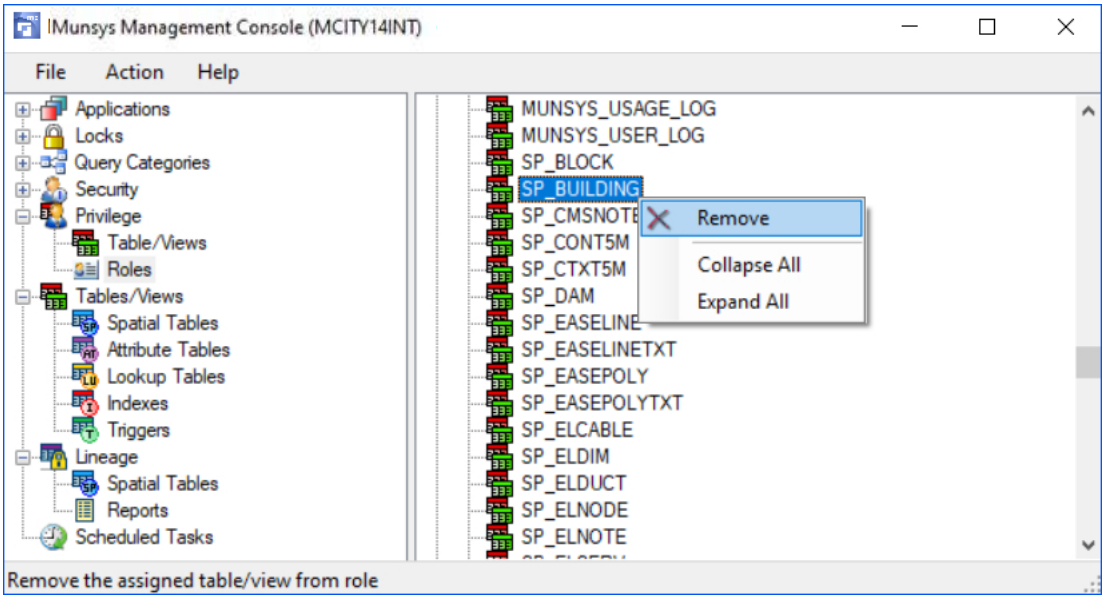


Figure 9 The Remove command in the context-sensitive menu

2 The **Remove Table/View** warning message is displayed.

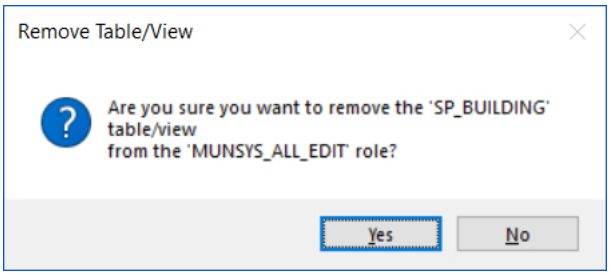


Figure 10 Remove Table/View warning message

3 Click **Yes** to remove the table or view.

Chapter 10
Tables and indexes

Introduction

The Tables/Views tree of the Munsys Management Console is used to manage spatial tables, lookup tables and indexes in the database.

With this tree, you can:

- Create, maintain and validate spatial tables
- Create and maintain lookup tables
- Create and maintain indexes on spatial tables, attributes and lookup tables

The Tables/Views tree: overview

The Tables/View main tree consists of three branches (nodes):

- Spatial Tables
- Attribute Tables
- Lookup Tables
- Indexes
- Triggers

Using the Action menu or context-sensitive (right-click) menu that is activated when an item is selected, tables and indexes as created and maintained.

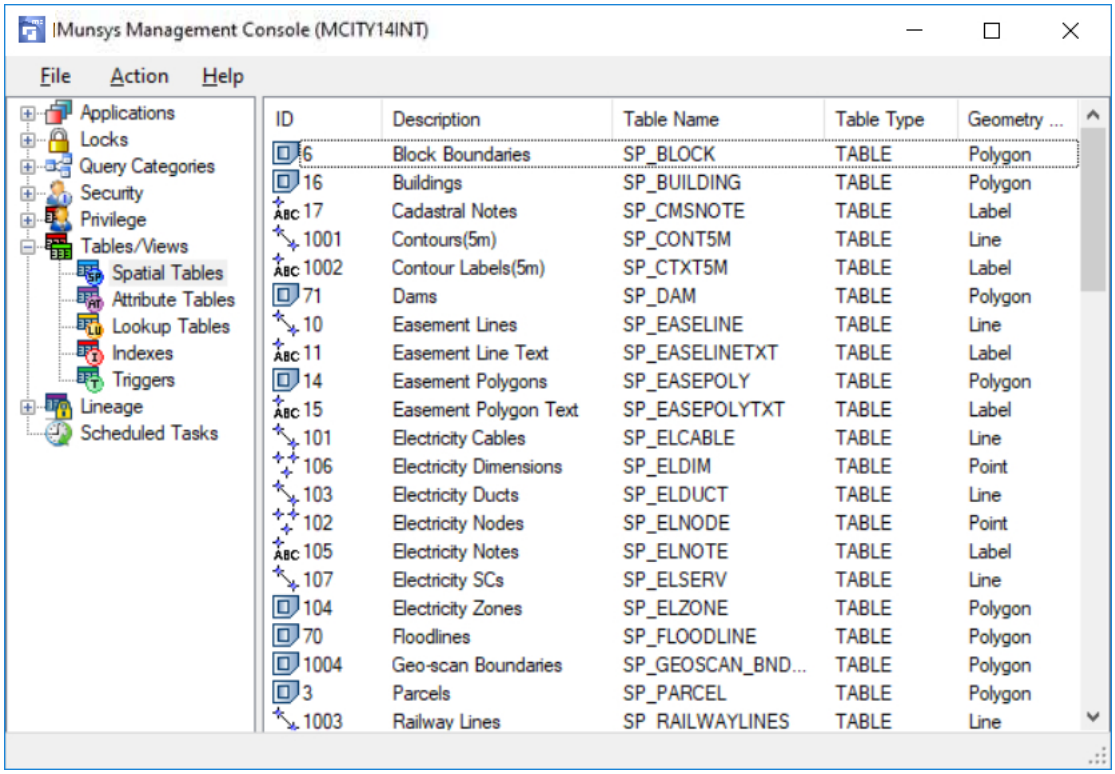


Figure 1 Munsys Management Console: Tables/Views tree

Working with spatial tables in the Munsys Management Console

Using the Spatial Tables item in the Tables/Views tree, you can do the following:

- Create custom spatial tables
- Drop a spatial table
- Validate a spatial table
- Add, modify or delete a column in a spatial table
- Edit Table Description

Creating custom spatial tables

Custom spatial tables are created by selecting the Create New Table or Create from Existing Table option on the Action menu, or from the context-sensitive (right-click) menu that is displayed when you right-click on Spatial Tables in the Tables/View tree.

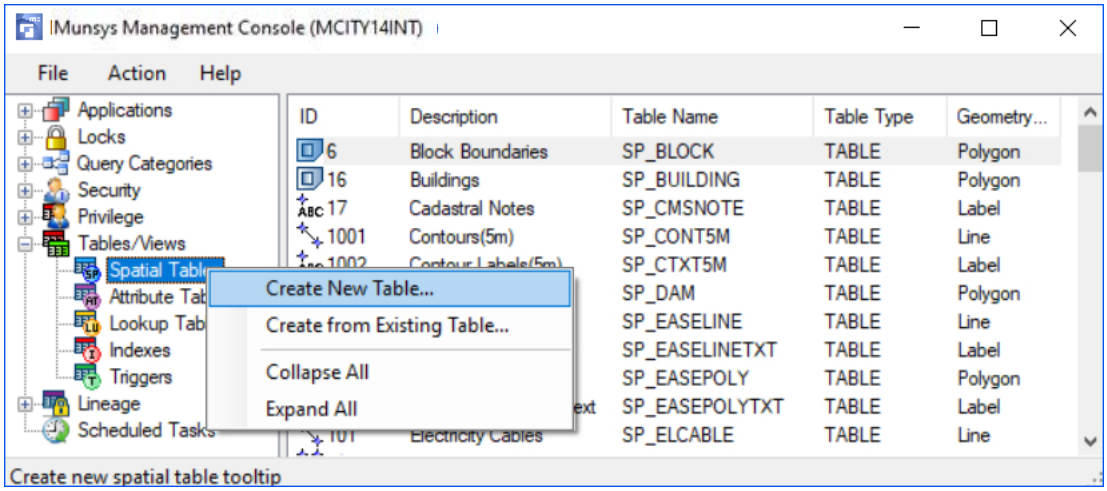


Figure 2 Select Create New Spatial Table

Spatial table names and descriptions have to be unique; when a new table is created in the database, the system checks whether the name or description is not yet present. If this is the case, the table will not be created. A custom spatial table is always assigned a MUNID of >1000.

When a new spatial table is created from an existing table, the following happens:

- The default columns are added to the table
- Columns are created in new table that exist in the old table but not in the new one
- Columns with the prefix _OLD in are created in the new table for all columns that exist in both the old and new tables.
- All the data from the original table is copied into new table into the corresponding OLD_ columns
- The table is inserted into MUNSYS_SP_TABLES
- The table is inserted into MUNSYS_SP_COLUMNS

- The Geometry Metadata (USER_SDO_GEOM_METADATA) is updated
- A default query is created (MUNSYS_QUERY)
- Roles are assigned to the table (MUNSYS_SDM_QUERY, MUNSYS_ALL_QUERY, MUNSYS_SDM_EDIT, MUNSYS_ALL_EDIT, MUNSYS_POWER, MUNSYS_ADMIN)
- Indexes are created on the GID and Geometry columns of the new table

When a custom spatial table is created, the Munsys system table MUNSYS_SP_COLUMNS is populated with the default columns for the new table and its corresponding geometry type. Default Munsys system queries are also created at this time.

Warning A maximum of 1000 columns may be added to a single spatial table, inclusive of the default columns per spatial object type.

To create a custom spatial table

1 Right-click on the **Spatial Tables** node of the **Tables/View** tree, and then select the **Create New Table...** option on the context-sensitive menu or the **Action** menu.

The New Spatial Table dialog box is displayed.

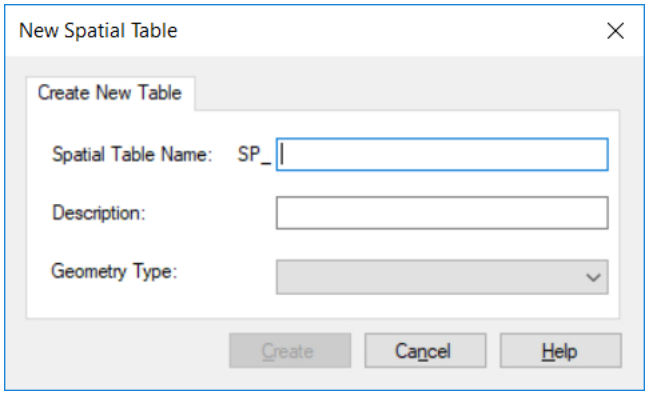


Figure 3 The New Spatial Table dialog box

2 Enter a unique name used for the spatial table. This name is also used as the primary spatial layer in **Munsys Spatial Data Manager**. The name specified is prefixed with **SP_** when creating the table in the database.

3 Enter a descriptive string that will identify which object the new spatial table represents. The description should be unique in describing the contents of the table.

4 From the **Geometry Type** list, select the type of geometry data that will be stored in the table. The available geometry types are **Point**, **Label**, **Polygon** and **Line**.

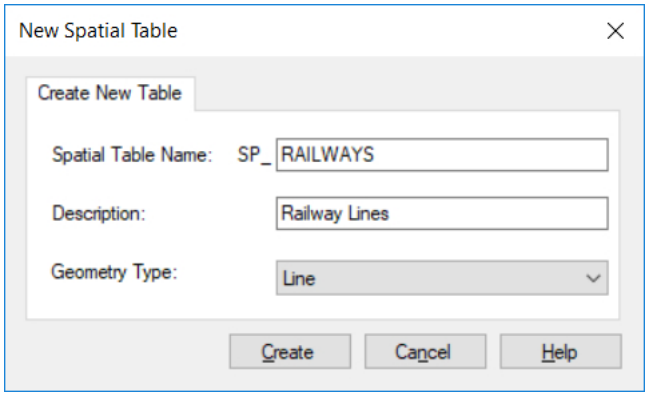


Figure 4 Creating a new spatial table

5 Click **Create**.

The spatial table is created in the database.

To create a custom spatial table from an existing table

1 Right-click on the **Spatial Tables** node of the **Tables/View** tree or on an existing spatial table, and then select the **Create From Existing Table...** option on the context-sensitive menu or the **Action** menu.

The New Spatial Table dialog box is displayed.

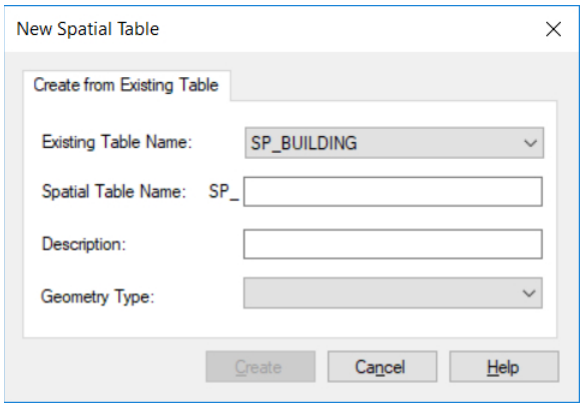


Figure 5 The New Spatial Table dialog box: create from existing

2 From the **Existing Table Name** list, select the spatial table on which you want to base the new table.

3 Enter a unique name used for the spatial table. This name is also used as the primary spatial layer in Munsys Spatial Data Manager. The name specified is prefixed with **SP_** when creating the table in the database.

4 Enter a descriptive string that will identify which object the new spatial table represents. The description should be unique in describing the contents of the table.

5 From the **Geometry Type** list, select the type of geometry data that will be stored in the table. The available geometry types are **Point**, **Label**, **Polygon** and **Line**.

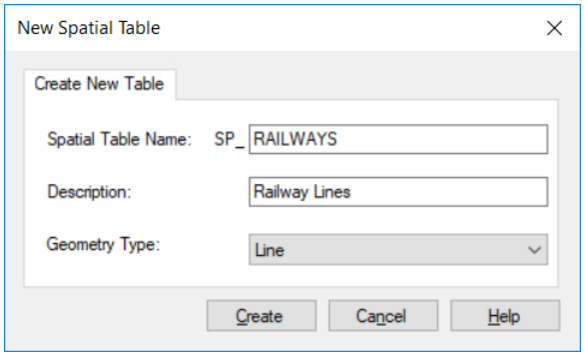


Figure 6 Creating a new spatial table

6 Click **Create**.

The spatial table is created in the database.

Dropping a spatial table

Custom spatial tables (tables with MunID > 1000) can be dropped from the database. Munsys-defined spatial tables cannot be dropped from the Munsys Management Console.

When a table is dropped from the database, all references to the table is deleted from:

- The system query table (MUNSYS_ QUERY)
- MUNSYS_SP_TABLES
- USER_SDO_GEOM_METADATA
- MUNSYS_SP_COLUMNS

To drop a spatial table, do the following:

1 Select the table that you want to drop, and then select the **Drop Table**... option on the **Action** menu or the **context-sensitive** (right-click) menu.

The following confirmation message is displayed:

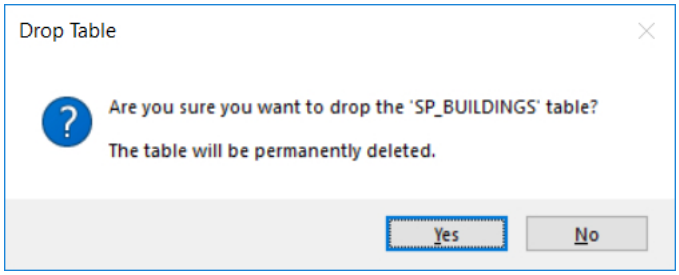


Figure 7 Drop Table confirmation message

2 Click **Yes** to drop the table from the database.

Validating a spatial table

The Spatial Table Validation function validates the geometry of spatial tables. You need to have the MUNSYS_ADMIN role assigned and be logged in as the schema owner to be able to validate a spatial table.

To validate a spatial table, do the following:

1 Select the table that you want to validate, and then select the **Validate Table**... option on the **Action** menu or the **context-sensitive** (right-click) menu.

The Validation Results dialog box is displayed.This dialog box consists of three tabs:

- Summary – this tab lists the main category of the tests that were performed, and whether the entire category failed or passed the validation test.

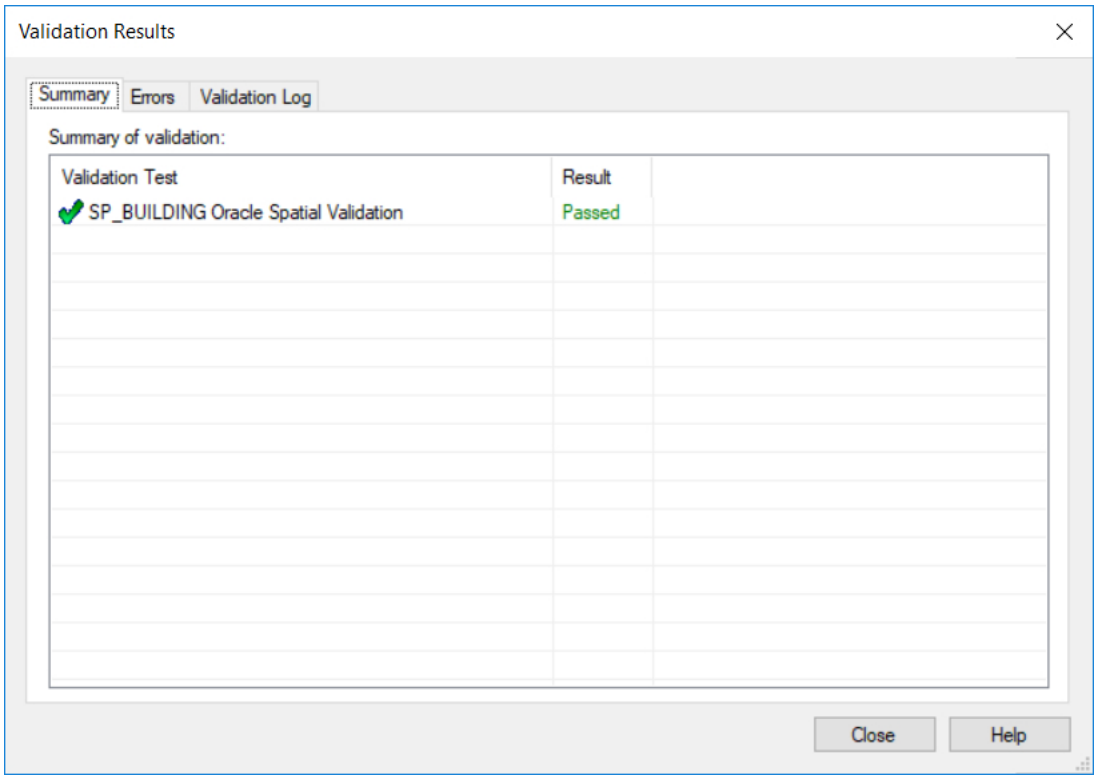


Figure 8 Validation Results: Summary tab

- Errors – this tab contains a list of all the errors that were encountered during the validation. The results are grouped under the validation category that was performed.
- Validation Log – this tab contains a list of the tests that were performed, as well as a detailed account of the outcome. The validation log can be saved as a text file by clicking the Save Log button.

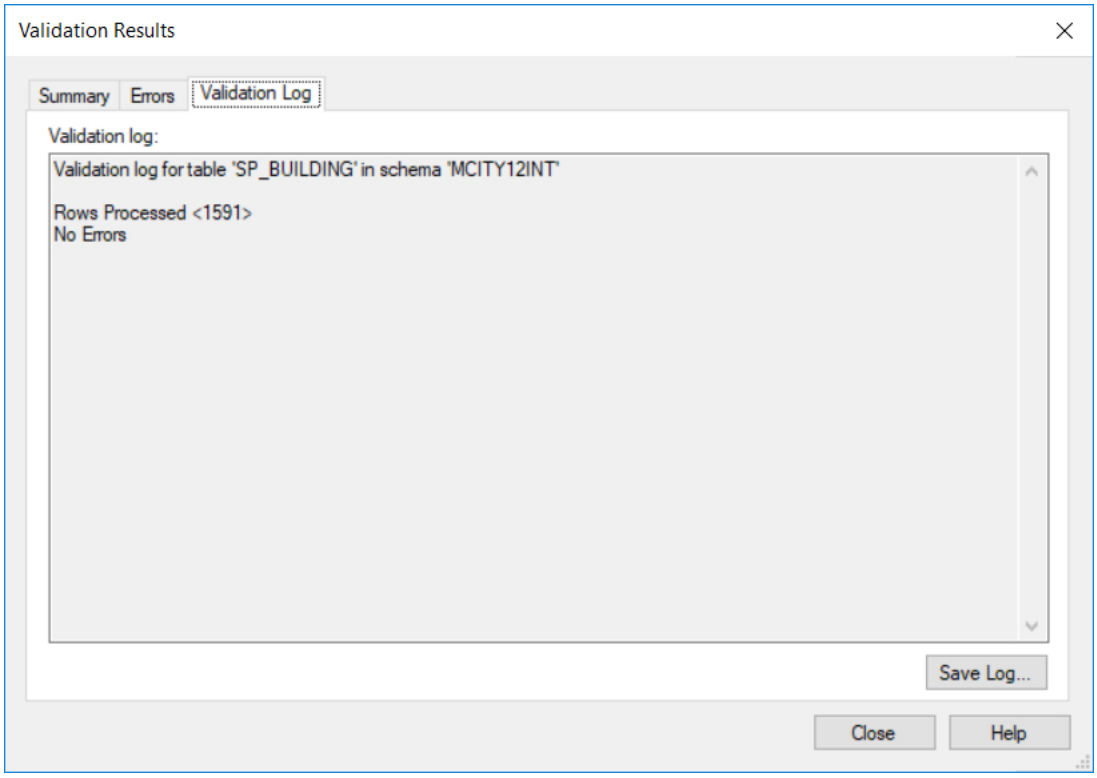


Figure 9 Validation Results: Log Tab

Editing columns in a spatial table

Creating a spatial table requires a predefined set of columns to be included in the table. When using the Munsys Management Console to create a spatial table, these columns are inserted into the table automatically.

One of the columns that needs to be present in the table is the GID column. This column is used as the geometry identifier for an object inserted into the table. It is a unique identifier throughout the Munsys schema, and is used as primary index for the spatial table.

Other required columns are used to hold label or text attributes for the object, such as angle, size and position of a town name.

The predefined columns cannot be deleted from a table. The database administrator can add new columns to any spatial table and also delete such columns.

Adding a column to a spatial table

1 Select the spatial table to which you want to add a column, and then select the **Edit Columns**... option on the **Action** menu or the **context-sensitive** (right-click) menu.

The Edit Spatial Table Columns dialog box is displayed.

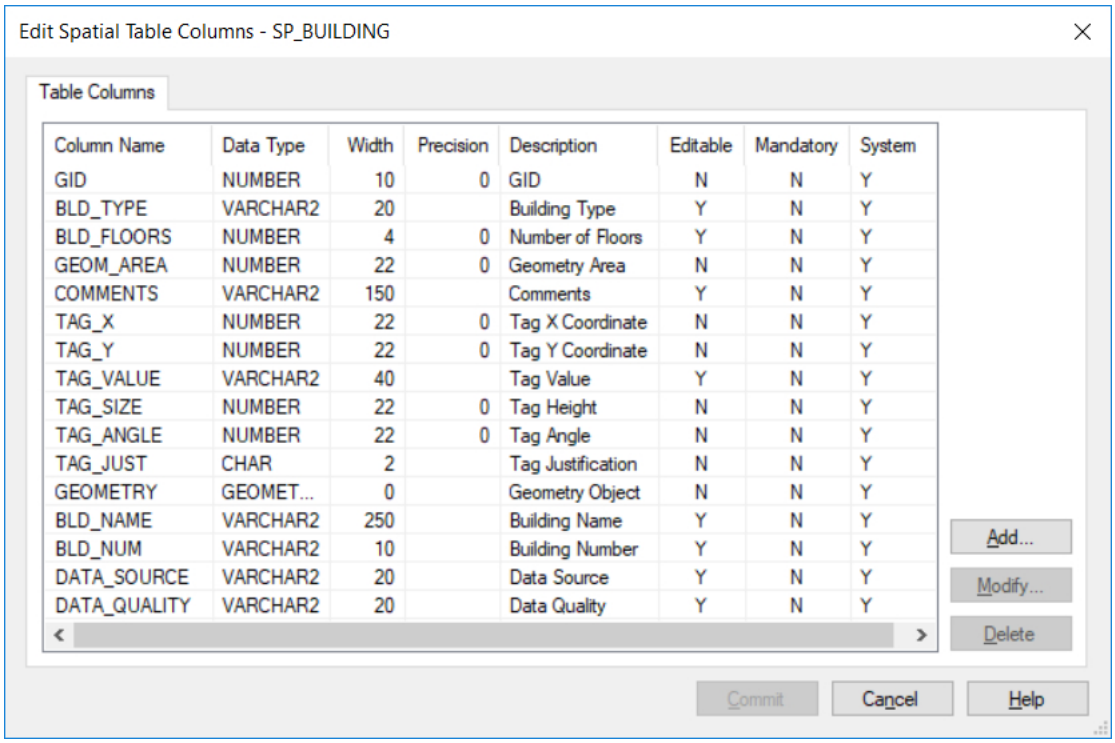


Figure 10 Edit Spatial Table Columns

2 Click the **Add** button.

The Add Column dialog box is displayed.

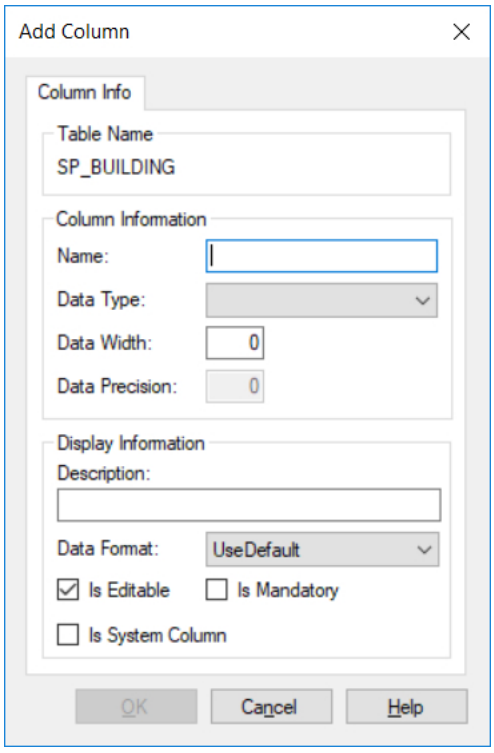


Figure 11 The Add Column dialog box

3 Enter the following column information:

- Name – the name of the column. This value cannot be changed.
- Data Type – the data type of the column (CHAR, NUMBER, VARCHAR2, DATE or GEOMETRY).
- Data Width – the maximum width of the column.
- Data Precision – the number of decimal digits that the column will display. This only applies to Number data types.

4 Enter the following display information:

- Description – enter a description for the new column, or change the description of an existing column.
- Data Format – select a data format from the list (Use Default, Angle, Date/Time, Hyperlink, or Lookup Table).
- Is Editable – specify whether the value may be edited in Munsys Applications.
- Is Mandatory - specify whether an integrity check rule must be added to the MUNSYS_INTEG_ATTR table. If checked, an entry is added and the default RULE_TYPE is set to NOTNULL. By default this remains unchecked.

5 Click **OK** to add the column to the spatial table.

Modifying a column in a spatial table

1 Select the spatial table of which you want to edit a column, and then select the **Edit Columns**... option on the **Action** menu or the **context-sensitive** (right-click) menu.

Note Only columns in custom (user-defined) spatial tables can be edited.

The Edit Spatial Table Columns dialog box is displayed.

2 Select the column that you want to edit, and then click the **Modify** button.

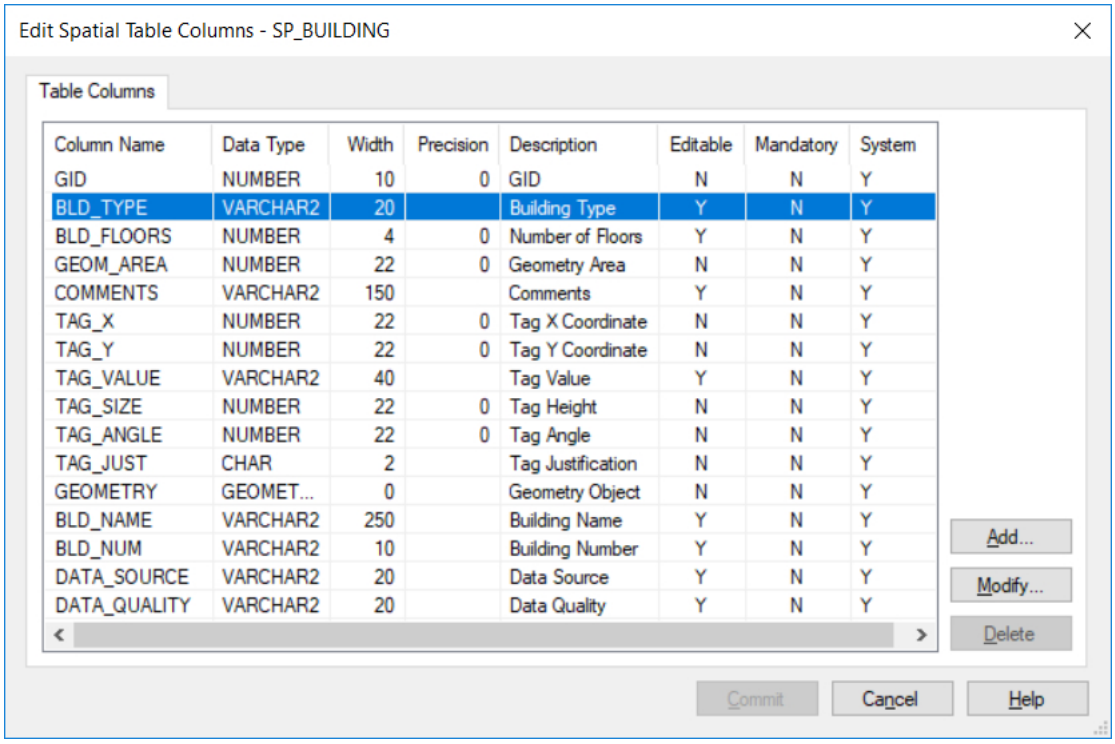


Figure 12 Edit Spatial Table Columns: Modify

The Modify Column dialog box is displayed.

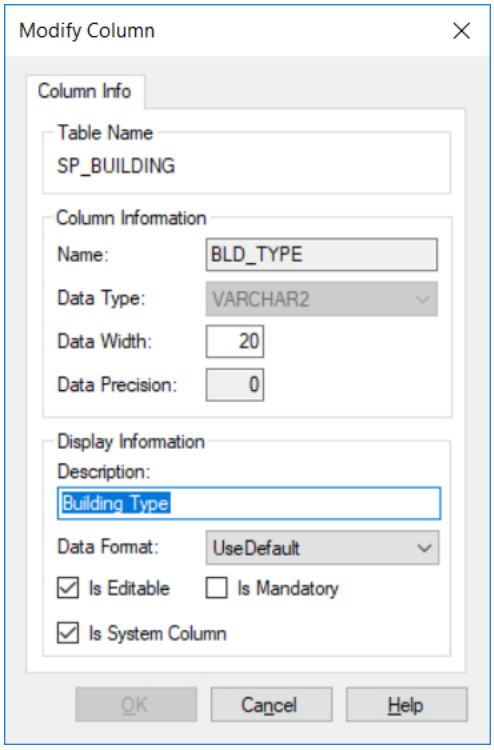


Figure 13 Modify Column

3 Edit the column as required, and then click **OK**.

4 Click the **Commit** button to commit the transaction to the database.

Deleting a column from a spatial table

The administrator can only delete custom columns from the spatial tables. When selecting a column to delete, the delete process checks to see if there are any integrity rules defined for the deleted column in the MUNSYS_INTEG_ATTR table, and if so, removes the reference when the deletion is committed.

Select the column to be deleted so that it is highlighted

1 Click the **Delete** button

The user is presented with the following Delete Column message

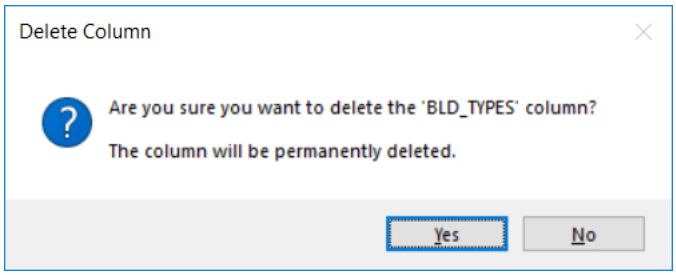


Figure 14 Delete Column message

2 Click the **Yes** button to continue with the deletion process, or click the **No** button to cancel the command.

3 Click the **Commit** button to commit the transaction to the database.

Working with attribute tables in the Munsys Management Console

Using the Attribute tables item in the Tables/Views tree, you can do the following:

- Create custom attribute tables
- Drop an attribute tables
- Add, modify or delete columns in an attribute table
- Modify table description

Create attribute tables

Attribute tables are created by selecting the Create New Table option on the Action menu, or from the context-sensitive (right-click) menu that is displayed when you right-click on Attribute tables in the Tables/View tree.

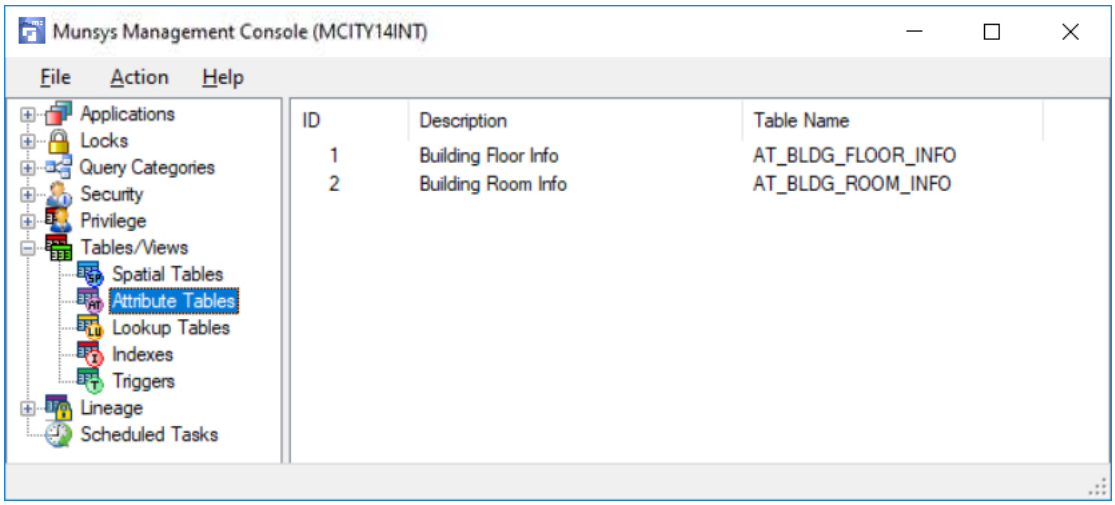


Figure 15 Munsys Management Console: Attribute Tables

When creating a new table, the system will automatically add the prefix [AT_] to the name supplied. Both the table name AND the descriptions must be unique; when a new table is created in the database, the system checks whether either the name or description already exists. If this is the case, the table will not be created. A custom attribute table is always assigned a MUN_ATID of >1000. When a new attribute table is created, the system

- automatically adds an ID column
- Populates MUNSYS_AT_TABLES Munsys system table
- Populates MUNSYS_AT_COLUMNS Munsys system table

Note Oracle imposes a limit of a 1000 columns per table.

To create a custom attribute table

1 Right-click on the Attribute tables node of the Tables/View tree, and then select the **Create New Table...** option on the context-sensitive menu or the Action menu.

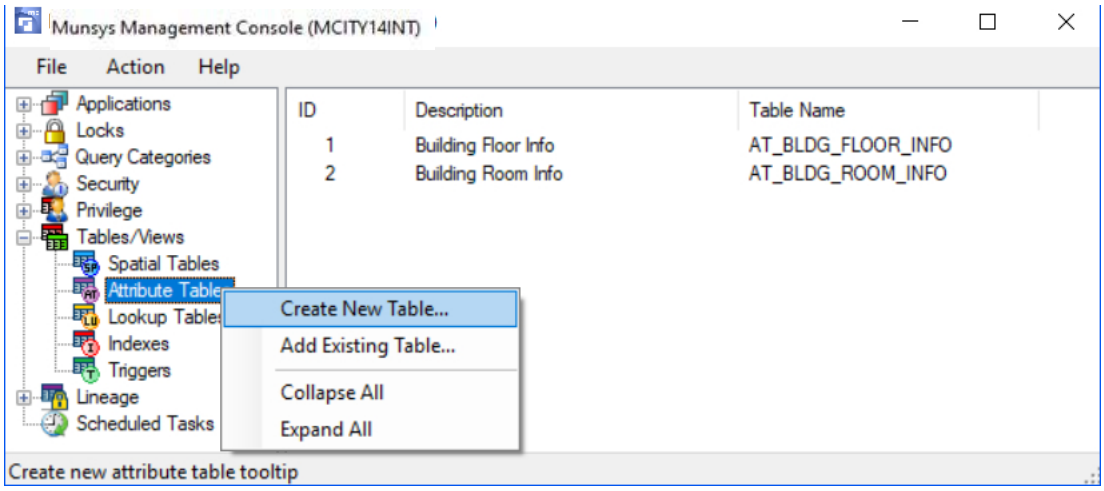


Figure 16 Create Attribute Table

The New Attribute Table dialog box is displayed.

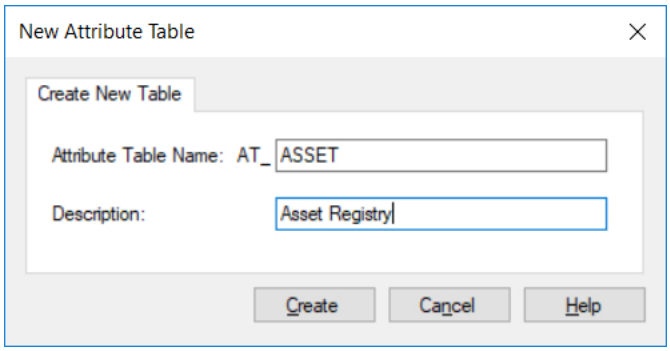


Figure 17 New Attribute dialog box

2 Enter a unique name used for the attribute table. The name specified is prefixed with AT_ when creating the table in the database.

3 Enter a descriptive string that will identify which object the new attribute table represents. The description should be unique in describing the contents of the table.

4 Click **Create**.

The attribute table is created in the database.

Note When creating a new attribute table in Munsys, the system will automatically add one column. This is the ID column which is used as a primary index for the attribute table. It is also possible to make this column unique throughout the Munsys schema by assigning the sequence MUNSEQ_ID to the ID field. (This is not done automatically.)

To add an attribute table from an existing table

1 Right-click on the **Attribute tables** node of the **Tables/View tree**, and then select the **Add Existing Table...** option on the context-sensitive menu or the Action menu.

A dialog box is displayed that lists all available non-system or spatial tables.

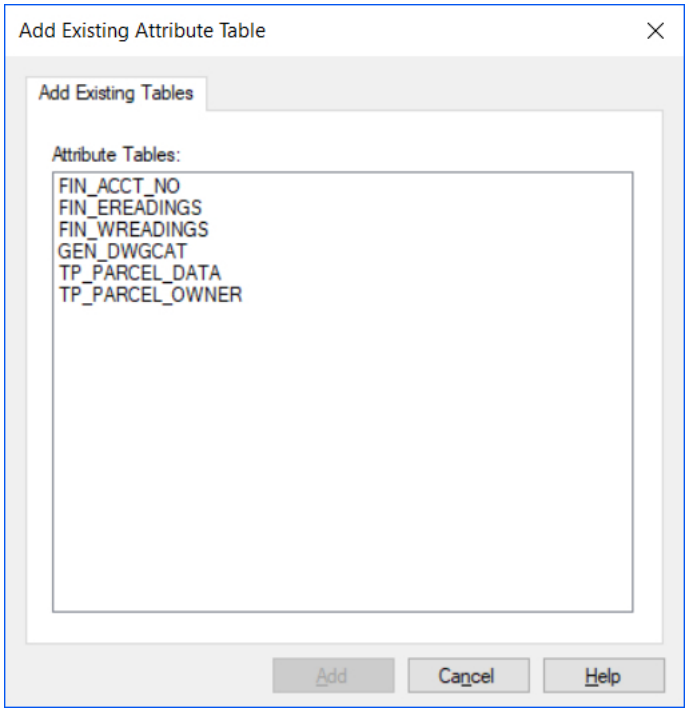


Figure 18 Add Existing Attribute Table

2 Select the attribute table that you want to add to the Table Name List and Click **Add**.

The attribute table is added in the database.

Note When adding an Attribute Table it enables you to now administer the table i.e. add/modify/delete columns. It also lists only the tables that you added so even if you have 100 attribute tables in the schema but only 10 are relevant then you can simply add those 10 tables.

Dropping an attribute table

Tables that are listed under Attribute tables and have a MUN_ATID > 1000 are custom attribute tables and they can be dropped from the database.

When a table is dropped, references to the table are also removed from the attribute table list, all references to the table is deleted from:

- MUNSYS_AT_TABLES
- MUNSYS_AT_COLUMNS

To drop an attribute table, do the following:

1 Select the table that you want to drop, and then select the **Drop Table...** option on the **Action** menu or the **context-sensitive** (right-click) menu.

The following confirmation message is displayed:

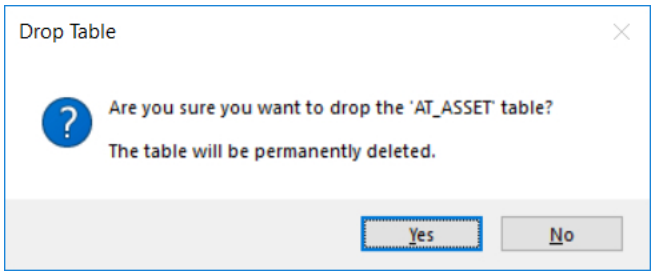


Figure 19 Drop Attribute Table notification

2 Click **Yes** to drop the table from the database.

Removing an attribute table

Tables that are listed under Attribute tables and have a MUN_ATID > 1000 are custom attribute tables and they can be dropped from the database.

When a table is removed from the attribute table list, all references to the table is deleted from:

- MUNSYS_AT_TABLES
- MUNSYS_AT_COLUMNS

To remove an attribute table, do the following:

1 Select the table that you want to drop, and then select the **Remove Table...** option on the **Action** menu or the **context-sensitive** (right-click) menu.

The following confirmation message is displayed:

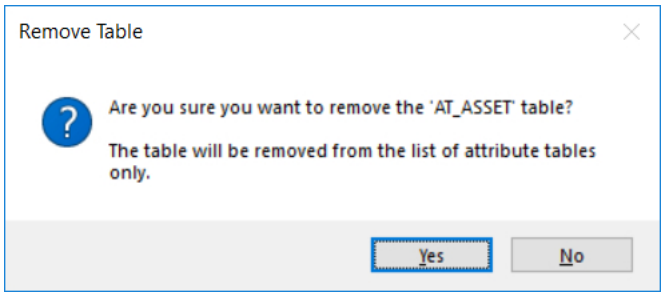


Figure 20 Remove Attribute Table notification

2 Click **Yes** to remove the table from the attribute list.

Editing columns in an attribute table

When creating a new attribute table in Munsys, the system will automatically add one column.Any other columns that are needed will have to be added after the table is created.

When you have a configured Attributed table, you can also use the edit function to add or edit columns to that table.

Adding a column to an attribute table

1 Select the attribute table to which you want to add a column, and then select the **Edit Columns...** option on the **Action** menu or the **context-sensitive** (right-click) menu.

The Edit Attribute Table Columns dialog box is displayed:

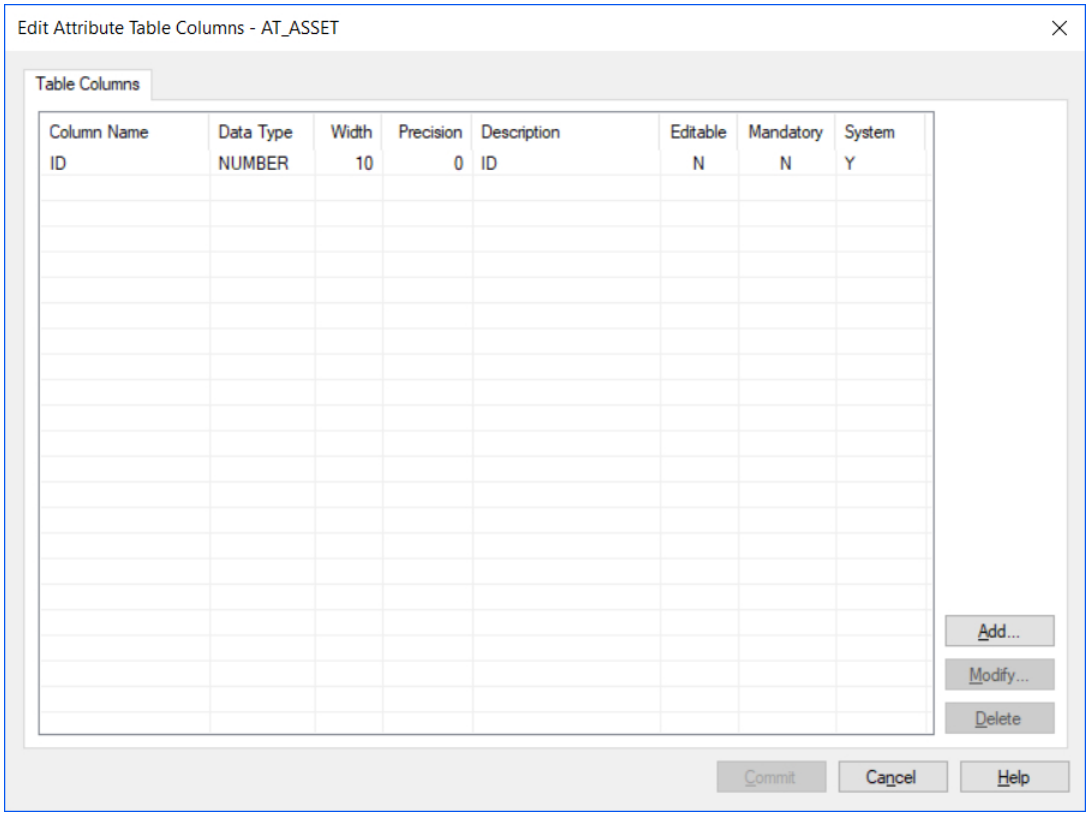


Figure 21 Add a column to an Attribute table

2 Click the **Add** button.

The Add Column dialog box is displayed:

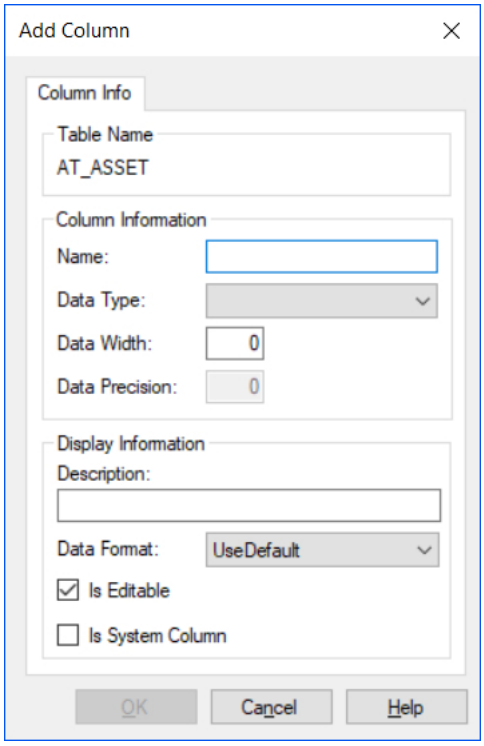


Figure 22 Add Column Dialog box

3 Enter the following column information:

- Name – the name of the column. This value cannot be changed.
- Data type – the data type of the column (CHAR, NUMBER, VARCHAR2, DATE or GEOMETRY).
- Data Width – the maximum width of the column.
- Data Precision – the number of decimal digits that the column will display. This only applies to Number data types.

4 Enter the following display information:

- Description – enter a description for the new column, or change the description of an existing column.
- Data Format – select a data format from the list (Use Default, Angle, Date/Time, Hyperlink, or Lookup Table).
- Is Editable (This specifies whether the value may be edited in Munsys Applications and is turned on by default.)
- Is System Column – flagging a column as a System column prevents it from been deleted/dropped.

5 Click **OK** to add the column to the attribute table

Modifying a column in an attribute table

1. Select the attribute table within which you want to edit a column, and then select the **Edit Columns...** option on the **Action** menu or the **context-sensitive** (right-click) menu.
2. The Edit Attribute Table Columns dialog box is displayed. Note that only user-defined attributes within registered attribute tables can be edited.
3. Select the column that you want to edit, and then click the **Modify** button.

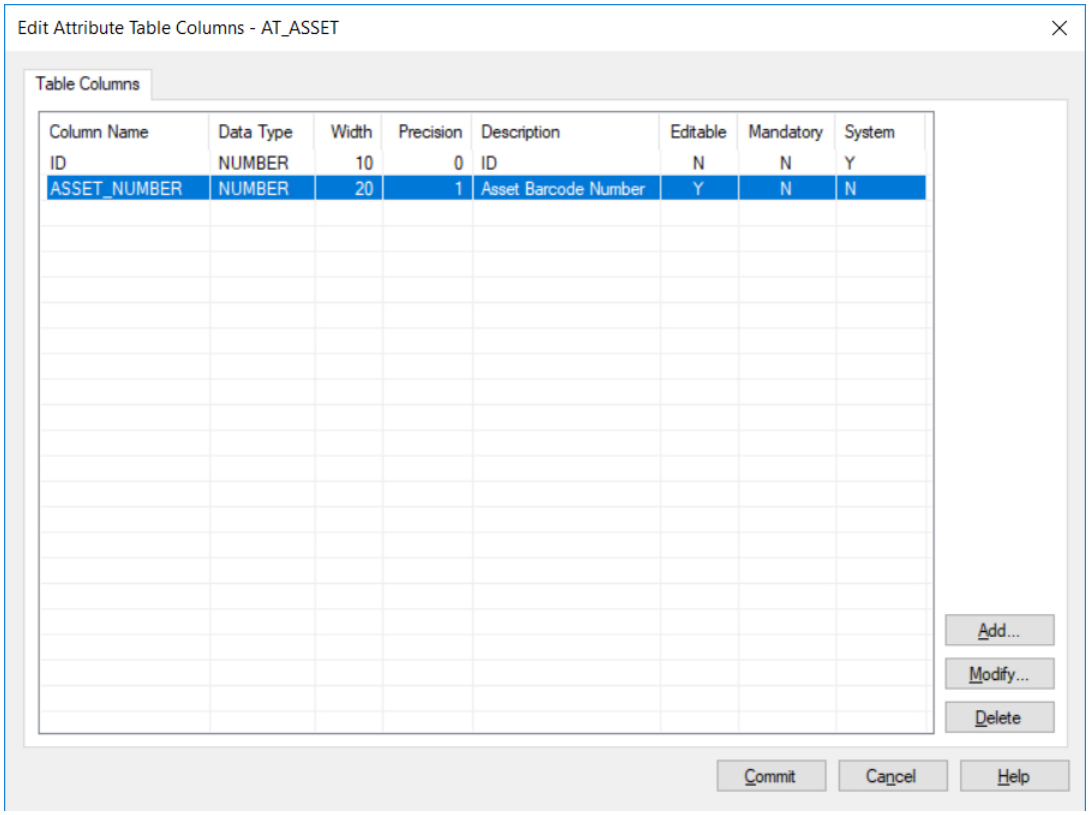


Figure 23 Modify Attribute Table Columns

The Modify Column dialog box is displayed:

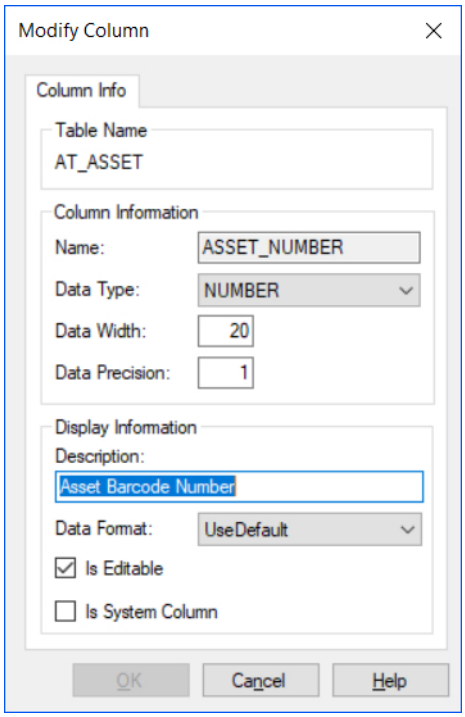


Figure 24 Modify Column

4 Edit the column as required, and then click **OK**.

5 Click the **Commit** button to commit the transaction to the database.

Deleting a column from an attribute table

The administrator can only delete custom columns from the attribute tables. When selecting a column to delete, the delete process checks to see if there are any integrity rules defined for the deleted column in the MUNSYS_INTEG_ATTR table, and if so, removes the reference when the deletion is committed.

1. Select the column to be deleted so that it is highlighted.
2. Click the **Delete** button

   The user is presented with the following Delete Column message:

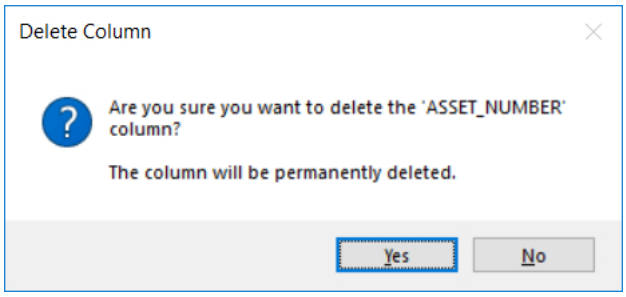


Figure 25 Delete column notification

3. Click the **Yes** button to continue with the deletion process or click the **No** button to cancel the command.
4. Click the **Commit** button to commit the transaction to the database.

Working with lookup tables in the Munsys Management Console

A lookup table is designed in the same way as an attribute table, with the difference that lookup tables are not used to capture data, but rather to look up data values. All possible values for a single field, for example all the possible values for zonings, are grouped into one lookup table. The user is presented with a list of values to choose from. This method of data capture enhances data accuracy. The database administrator can add lookup values to lookup tables, or modify or delete existing values.

Note The lookup tables have a unique primary key index assigned to the lookup code column enforcing the LCODE value to be unique.

Using the Lookup Tables item in the Tables/Views tree, you can do the following:

- Create new lookup tables
- Create and edit links between lookup tables and spatial tables
- Delete lookup links
- Drop lookup tables
- Create and edit lookup values in a lookup table

Creating a new lookup table

To create a new lookup table, do the following:

1 Select the **New Lookup Table...** option on the **Action** menu or the context-sensitive (right-click) menu that is displayed when you select **Lookup Tables** on the **Tables/Views** tree.

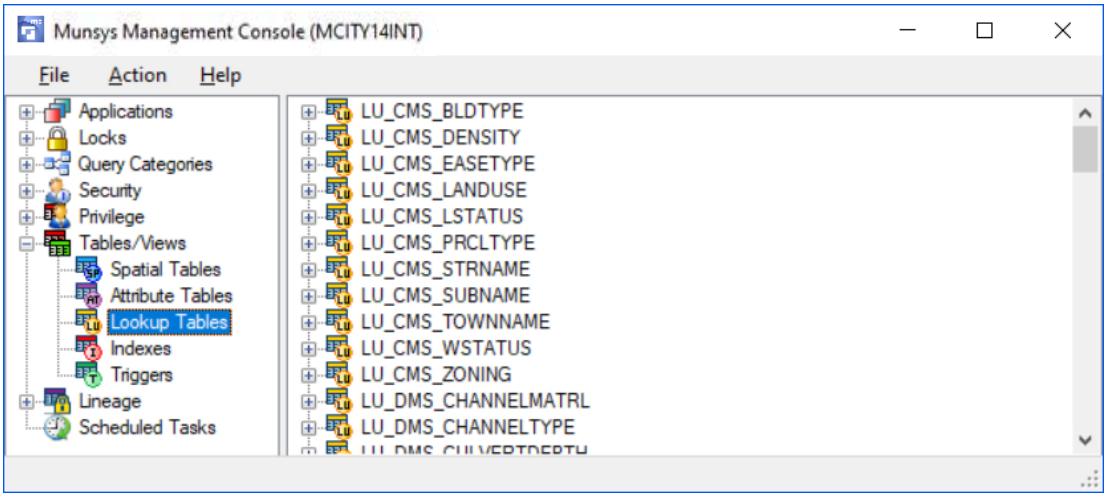


Figure 26 Munsys Management Console: Lookup Tables

The Create Lookup Table dialog box is displayed:

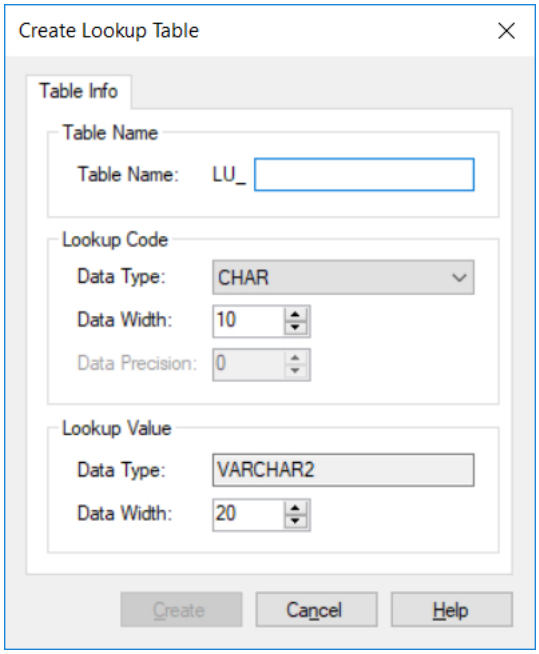


Figure 27 The Create Lookup Table dialog box

2 Enter the following information:

- Table Name – enter a descriptive name for the lookup table. The table is automatically prefixed with LU_.
- Lookup Code: Data Type – enter the appropriate data type for the lookup table: CHAR, VARCHAR2 or NUMBER. If you select NUMBER, the data precision field becomes available.
- Lookup Code: Data Width – enter the width for the column. The maximum is 20.
- Lookup Code: Data Precision – enter the number of decimal digits that the column will display. This field is only available when you selected NUMBER as the data type.
- Lookup Value: Data Type – the data type for the lookup value is set to VARCHAR2 by default. This value cannot be changed.
- Lookup Value: Data Width – enter the maximum width for the column. The maximum is 200

3 Click the **Create** button to create the new lookup table.

Creating links between lookup tables and spatial tables

To create a link between a lookup table and a spatial table, do the following:

1 Select the **Lookup Links...** option on the **Action** menu or the context-sensitive (right-click) menu that is displayed when you select the appropriate **Lookup Table** on the **Tables/Views** tree.

The Lookup Links dialog box is displayed:

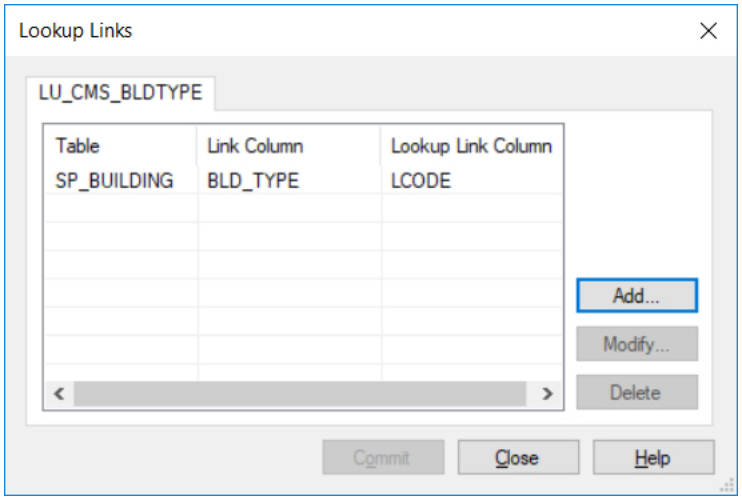


Figure 28 The Lookup Links dialog box

2 Click the **Add**... button.

The Lookup Link dialog box is displayed.

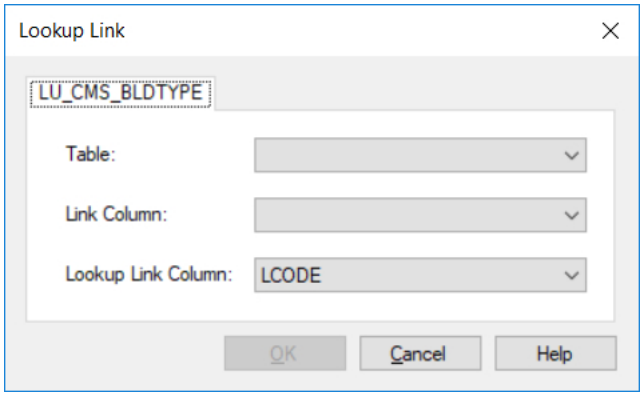


Figure 29 The Lookup Link dialog box

3 Select the required spatial table from the list.

4 Select the required spatial link column from the list.

5 Select the required lookup link column as required.

6 Click **OK** to create the lookup link.

Editing links between lookup tables and spatial tables

To edit a link between a lookup table and a spatial table, do the following:

1 Select the **Lookup Links...** option on the **Action** menu or the context-sensitive (right-click) menu that is displayed when you select the appropriate **Lookup Table** on the **Tables/Views** tree.

The Lookup Links dialog box is displayed:

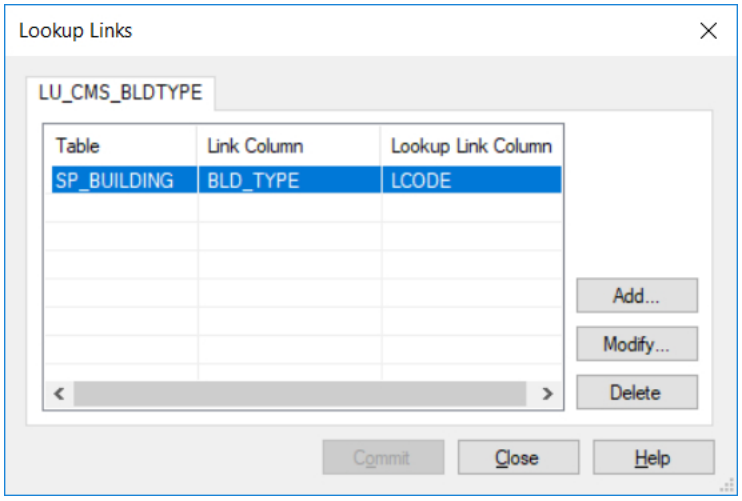


Figure 30 The Lookup Links dialog box

2 Select the lookup link that you want to edit, and then click the **Modify**... button.

The Lookup Link dialog box is displayed.

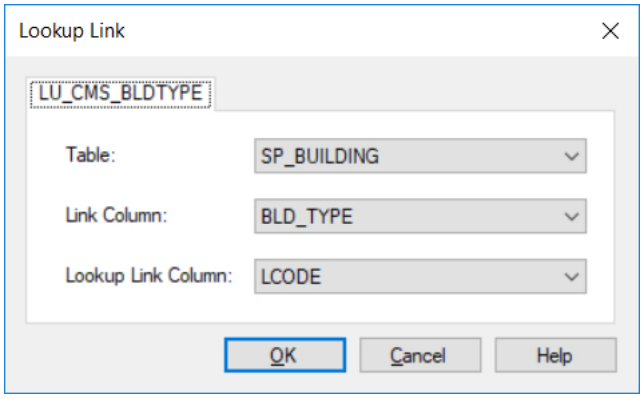


Figure 31 The Lookup Link dialog box

3 Change the spatial table if required.

4 Change the spatial link column and the lookup link column as required.

5 Click **OK** to apply the changes to the lookup link.

Deleting a link between a lookup table and a spatial table

To delete a link between a lookup table and a spatial table, do the following:

1 Select the **Lookup Links...** option on the **Action** menu or the context-sensitive (right-click) menu that is displayed when you select the appropriate **Lookup Table** on the **Tables/Views** tree.

The Lookup Links dialog box is displayed:

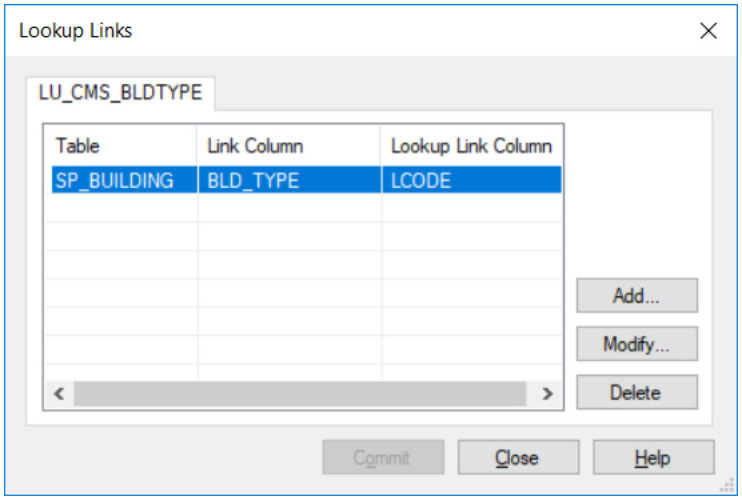


Figure 32 The Lookup Links dialog box

2 Select the lookup link that you want to delete, and then click the **Delete**... button.

The lookup link is deleted.

3 Click **Commit** to save the changes to the database.

Tip You can also remove a link by selecting the appropriate link in the content pane, and then selecting the **Remove Link** option on the Context-sensitive menu.

The following dialog is displayed:

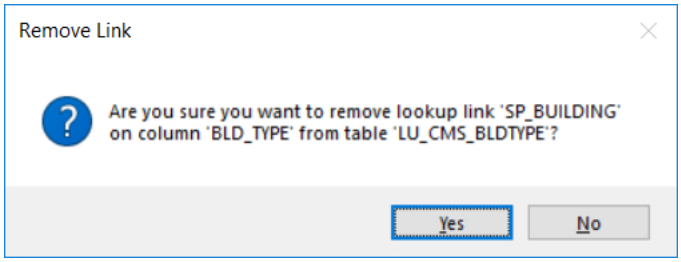


Figure 33 Remove Link notification

Dropping a lookup table from the database

To drop a lookup table from the database, do the following:

1 Select the **Drop Lookup Table...** option on the **Action** menu or the context-sensitive (right-click) menu that is displayed when you select the appropriate **Lookup Table** on the **Tables/Views** tree.

A confirmation message is displayed:

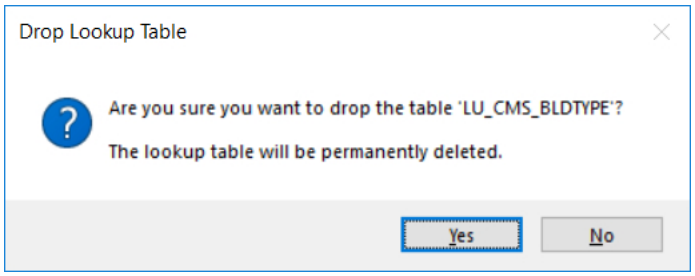


Figure 34 Drop Lookup Table notification

2 Click **Yes** to drop the lookup table from the database.

Creating a lookup value in a lookup table

To create a new lookup value for a lookup table, do the following:

1 Select the **Lookup Values...** option on the **Action** menu or the context-sensitive (right-click) menu that is displayed when you select the appropriate **Lookup Table** on the **Tables/Views** tree.

The Lookup Values dialog box is displayed.

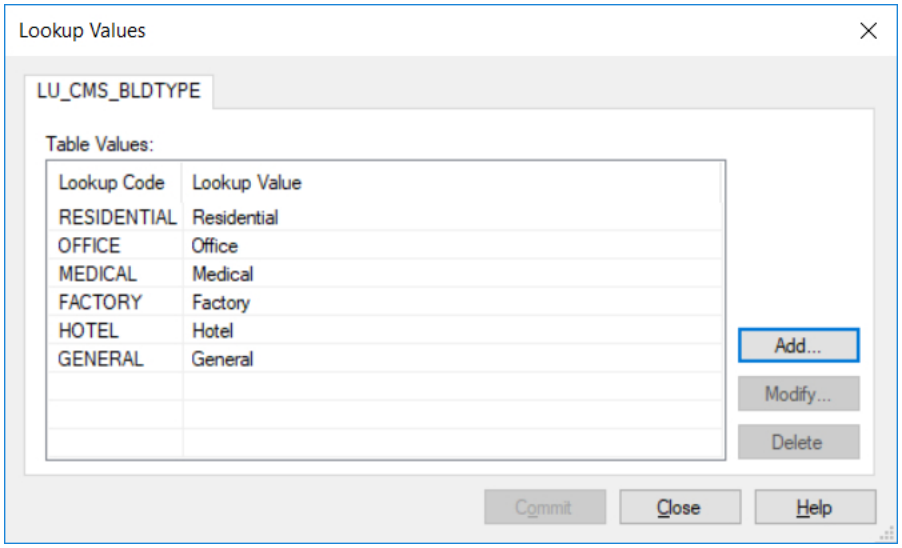


Figure 35 The Lookup Values dialog box

2 Click the **Add**... button.

The New Lookup Value dialog box is displayed.

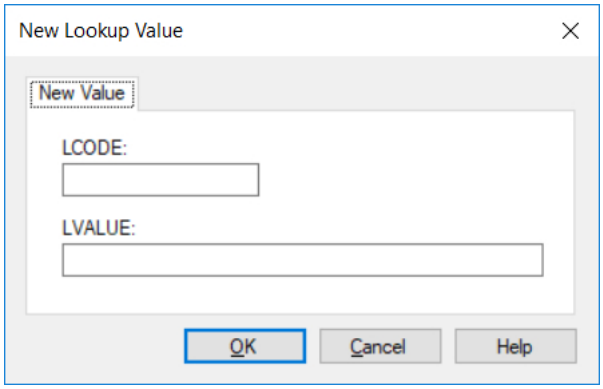


Figure 36 The New Lookup Value dialog box

3 Enter a new lookup code (LCODE).

4 Enter a new lookup value (LVALUE).

5 Click **OK** to add the lookup value to the table.

6 Click **Commit** on the **Lookup Values** dialog box to save the changes to the database.

Modifying a lookup value in a lookup table

1 Select the **Lookup Values...** option on the **Action** menu or the context-sensitive (right-click) menu that is displayed when you select the appropriate **Lookup Table** on the **Tables/Views** tree.

The Lookup Values dialog box is displayed:

2 Select the value that you want to change, and then click the **Modify**... button.

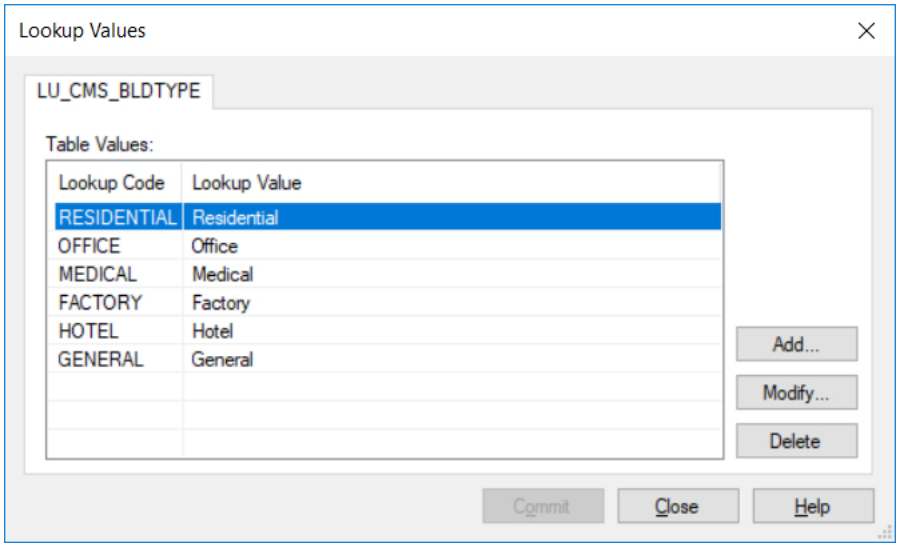


Figure 37 The Lookup Values dialog box

The Modify Lookup Value dialog box is displayed.

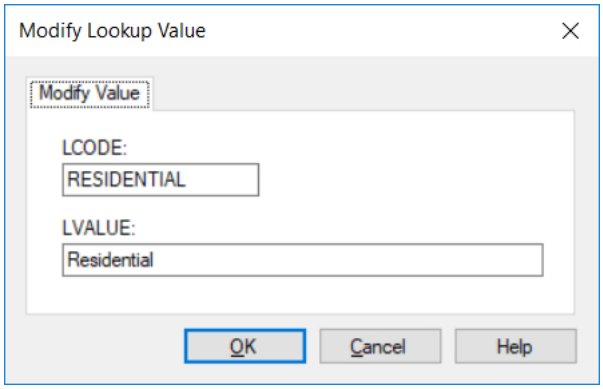


Figure 38 The Modify Lookup Value dialog box

3 Change the lookup value as required, and then click **OK**.

4 Click **Commit** on the **Lookup Values** dialog box to save the changes to the database.

Deleting a lookup value from a lookup table

1 Select the **Values...** option on the **Action** menu or the context-sensitive (right-click) menu that is displayed when you select the appropriate **Lookup Table** on the **Tables/Views** tree.

The Lookup Values dialog box is displayed:

2 Select the value that you want to delete, and then click the **Delete**... button.

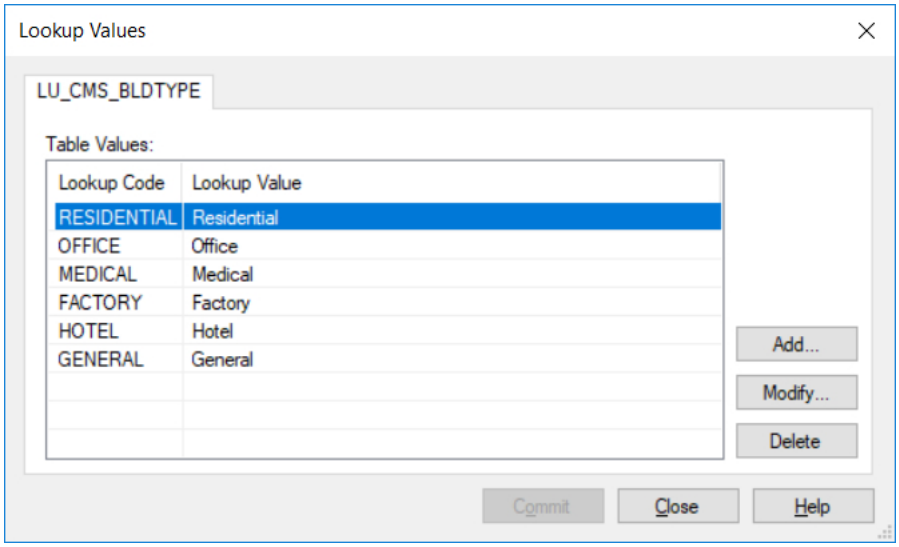


Figure 39 The Lookup Values dialog box

The lookup value is deleted from the lookup table.

3 Click **Commit** on the **Lookup Values** dialog box to save the changes to the database.

Editing lookup columns

1 Select the **Edit Columns**... option on the **Action** menu or the **context-sensitive** (right-click) menu that is displayed when you select the appropriate Lookup Table on the Tables/Views tree.

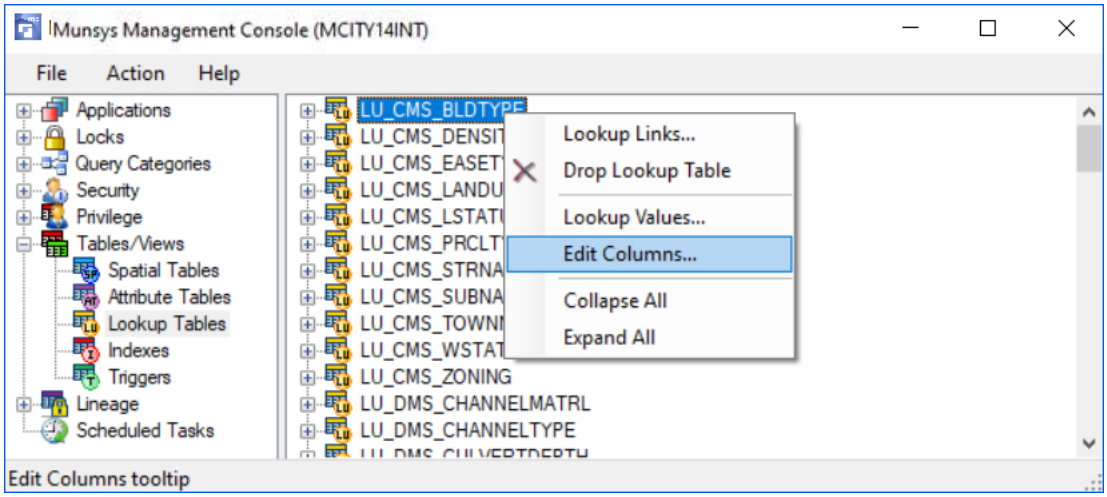


Figure 40 Edit Lookup Columns

The Edit Lookup dialog box is displayed:

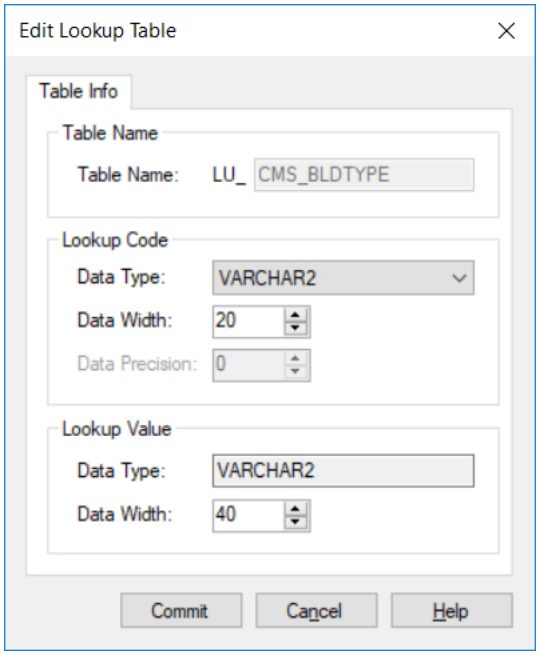


Figure 41 Edit Lookup Table

2 Set the Lookup Code column to either CHAR, VARCHAR or NUMBER. Each option displays parameters associated with that data type. Set each of the parameters as required.

3 Only the width of the Lookup Value can be changed. Set this value as required

4 If you tried to change the data type of any lookup table that contains values, and error will occur to prevent data loss.

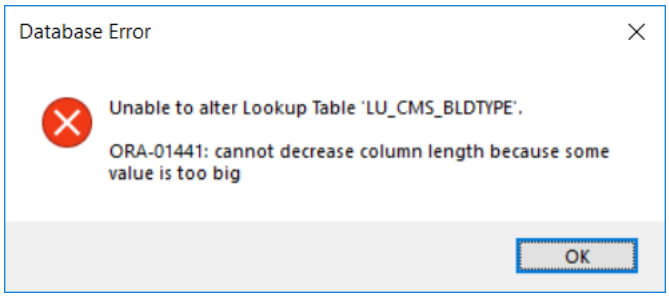


Figure 42 Data loss notification

5 Click **Commit** to update the changes in the database

Working with indexes on spatial tables and lookup tables

In the Munsys Management Console, you can create custom indexes, as well as recreate and drop indexes on Munsys spatial, attribute and lookup tables using Indexes on the Tables/Views tree.

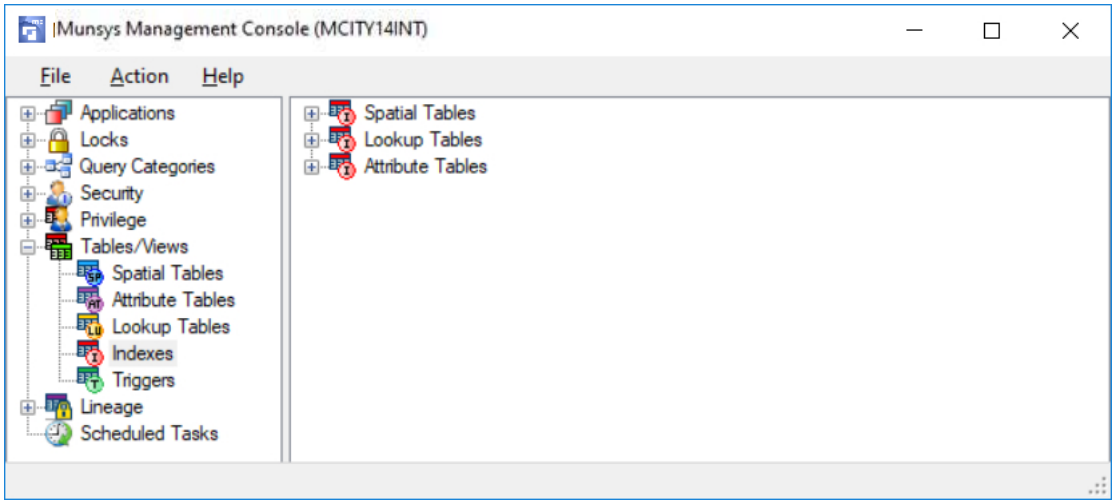


Figure 43 Munsys Management Console: Indexes

Indexing is separated into two different sections:

- Spatial Tables – All tables starting with the prefix SP_.
- Lookup Tables – All tables starting with prefix LU_.
- Attribute Tables – All tables that are being managed by the Munsys Attribute Management tools.

A custom index for a spatial, attribute or lookup table is created using the Create Custom Index dialog box. This dialog box is accesses by selecting the appropriate spatial, attribute or lookup table, and then selecting the Create Custom Index option on the Action menu or context-sensitive (right-click) menu.

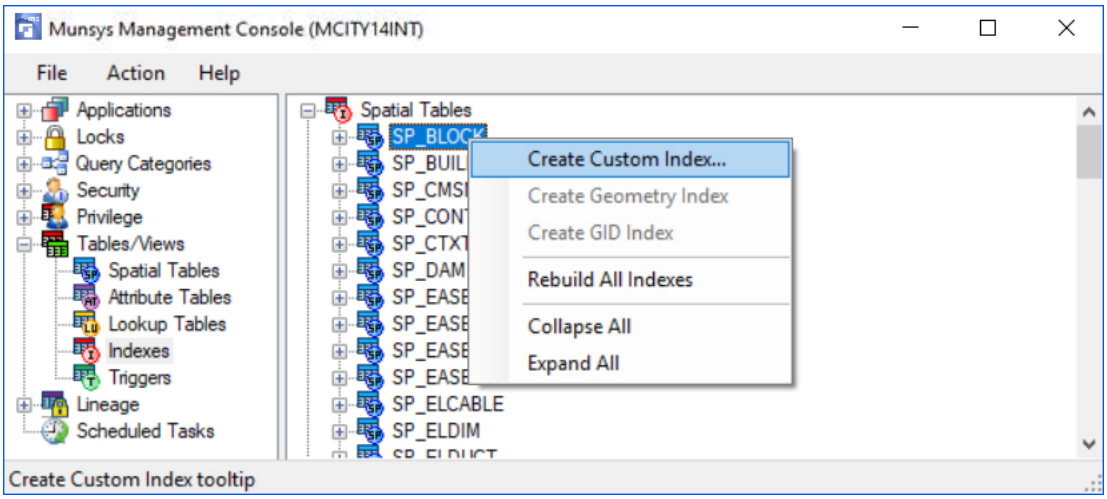
Each spatial or lookup table lists the columns on which indexes have been created. A spatial or attribute table can have multiple indexes.

Creating a custom index for a spatial or lookup table

The Create Custom Index dialog box is used to create custom indexes for columns in a selected spatial table. The GID and GEOMETRY columns have preset index names and custom indexes cannot be created on these two columns.

To create a custom index for a spatial or lookup table, do the following:

1 Select the **Create Custom Index...** option on the **Action** menu or the context-sensitive (right-click) menu that is displayed when you select the appropriate table on the **Tables/Views** tree.



The Create Custom Index dialog box is displayed:

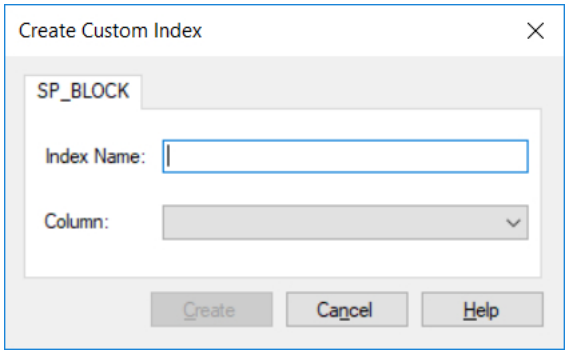


Figure 44 The Create Custom Index dialog box

2 Enter a descriptive name for the custom index that you are creating.

3 From the **Column** list, select the column that you want to create the index for (the list only displays currently non-indexed columns).

4 Click **Create** to create the new custom index.

Rebuilding all indexes on a spatial or lookup table

To rebuild an index on a table, do the following:

1 Select the **Rebuild All Indexes** option on the **Action** menu or the context-sensitive (right-click) menu that is displayed when you select the appropriate table on the **Tables/Views** tree.

The following message is displayed:

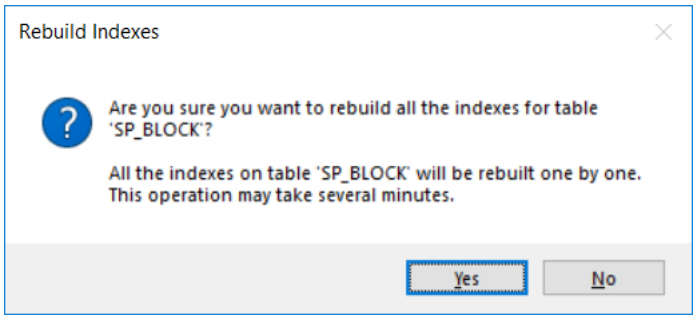


Figure 45 Rebuild indexes notification

2 Click **Yes** to rebuild the indexes.

The indexes are rebuilt.

Chapter 11
Munsys Lineage

Introduction

Munsys Lineage is primarily used as a data management tool to track and archive changes made to records in spatial tables. This is achieved through the implementation of database triggers which populate Munsys system tables based on the preferences selected when Lineage is added to a spatial table. Reports can also be generated to query the database for monitor or usage purposes.

This chapter is divided into the following sections for easy reference:

- The Lineage Tree: overview
- Working with Munsys Lineage
- Enabling or disabling Lineage
- Adding Lineage
- Disabling triggers
- Enabling triggers
- Editing Lineage
- Removing Lineage
- Recompiling triggers
- Rebuilding triggers
- Deleting monitor logs
- Running reports
- Adding reports
- Editing reports
- Deleting reports

The Lineage tree: overview

The **Lineage** tree consists of the **Spatial Tables** and **Reports** branch.

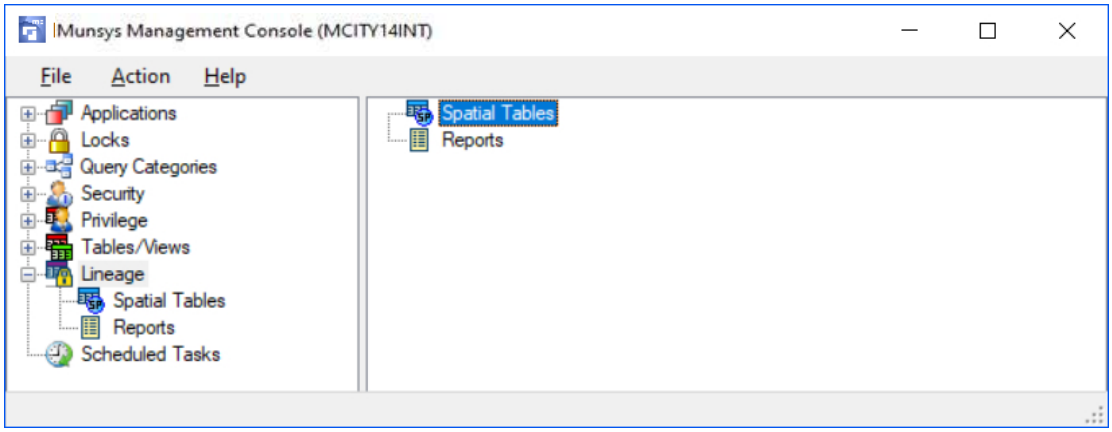


Figure 1 Munsys Management Console: Lineage Tree

- Clicking on Spatial Tables in the Lineage tree displays existing spatial tables (if any) to which triggers have being implemented to track Lineage. Refer to the specific sections in this chapter for more details on the available functions on the Spatial Tables branch.

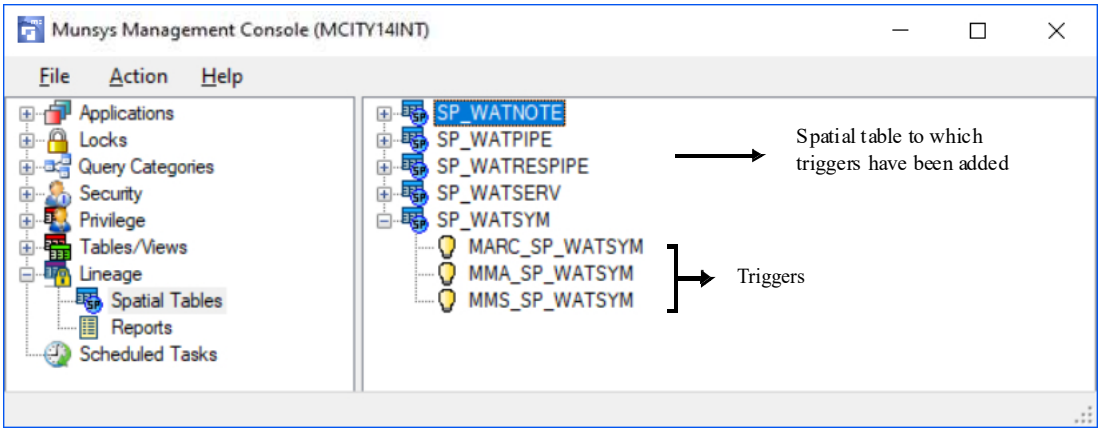


Figure 2 Lineage Tree: Spatial Tables branch

- Clicking on Reports in the Lineage tree displays the reports that can be run. Reports are created and maintained in this window. Refer to the specific sections in this chapter for more details on the available functions on the Reports branch.

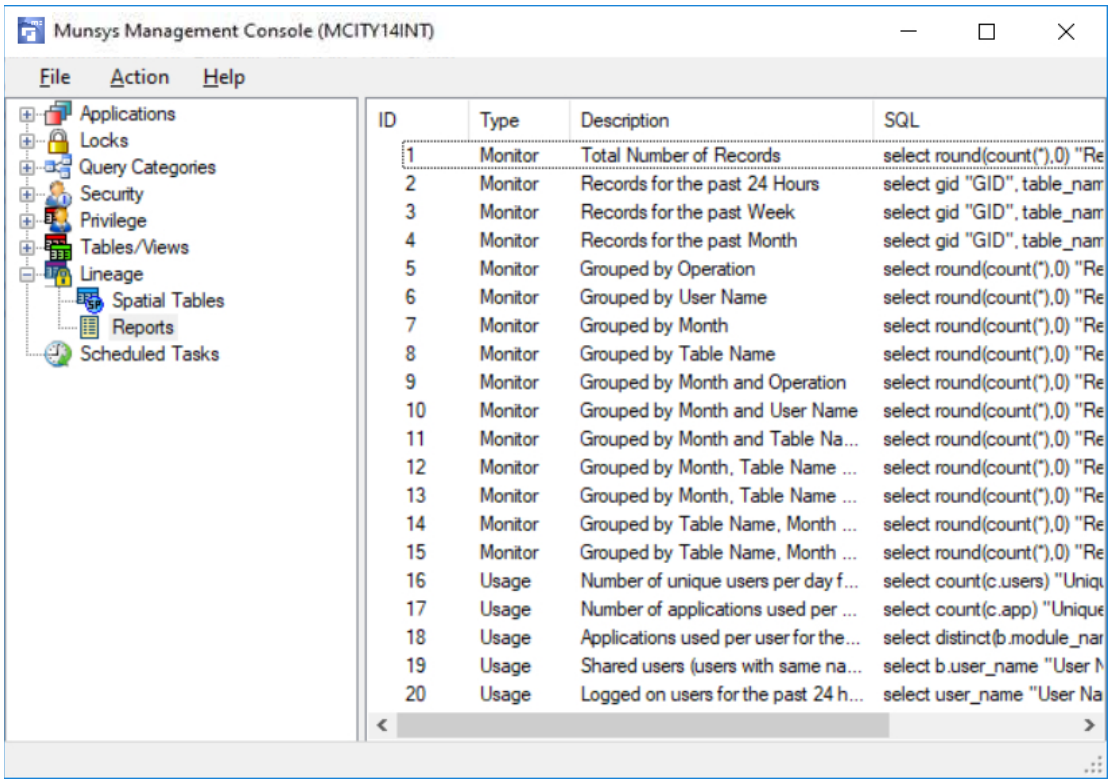


Figure 3 Munsys Management Console: Reports

Working with Munsys Lineage

Munsys Lineage can be used to:

- **Monitor** changes to records made in a specified table.
- **Archive** deleted or updated records into an archive table.
- **Report** according to specific monitor and usage requirements.

Monitor

Munsys Lineage can be used to log and monitor the following information about operations done on a table:

- the type of operation
- the date and time of the operation
- the table on which the operation was done
- the user who executed the operation

Logging is done during insert, update, and delete operations on a specified table in the Oracle schema. Updates are classified based on the columns updated, and as Spatial or Attribute, for example if the geometry of an object has been changed, it can be classified as a spatial update. Logging information is stored in the MUNSYS_MONITOR_LOG and is maintained by the administrator to prevent logs from growing too big.

Lineage date is important, especially if you want to be able to supply incremental updates to other customers, for example if the county wants to supply changes to the municipality based on a specific date.

Note The **Show Info** function in Munsys Applications shows the history of the object on the **Linked Tables** tab. This is done by inserting a record into the **MUNSYS_LNK_TABLES** table.

Archive

Munsys Lineage can be used to archive deleted or updated records into an archive table associated with a specified spatial table. This requires the creation of an archive table (SP_AD_tablename) and an associated archive trigger for a selected spatial table in the database. The archive trigger is implemented in such a way that all the columns in a specified spatial table are archived. When Lineage is added to a spatial table, the structure of that spatial table is duplicated to create an archive table. The archive table is populated with records that meet archive requirements as specified when Lineage is added or edited. Three extra columns are added to the archive table to keep track of the following information:

- the type of operation performed which resulted in the record being archived: An operation type column (ARCHIVE_OPERATION) is added to the archive table, which can be populated with the operation types DEL (delete) and UPD (update).
- the date and time of the archive: A date stamp column (ARCHIVE_DATE) is added to the archive table, which is populated with the date and time that the archive operation occurred.
- the archived ID: A unique ID column (ARCHIVE_ID) of the archived record is added.

The archive table is also inserted into the Munsys system tables, enabling users to query deleted records from an archive table using Munsys Applications.

The database administrator maintains the records in the archive tables to prevent the tables from growing too big. Records are deleted from the archive table based on the time stamp of the deleted record.

Reports

Reports can also be generated from spatial tables to which Lineage is applied. Administrator can generate monitor and usage reports to provide basic or detailed reporting based on predefined or customised requirements.

Munsys Lineage also reports on user logon/logoff and license usage, based on the entries in the MUNSYS_USAGE_LOG table.

Enabling or disabling Lineage

Lineage can be enabled and disabled from within the schema Properties dialog box. The Lineage tree branch only becomes active and functional once it has been enabled for a schema. Alternatively, when Lineage is not enabled, all Lineage functionality is disabled and cannot be modified in the Lineage tree branch of the Munsys Management Console.

To enable / disable Lineage and for more information, please refer to the chapter on Schema management: The Lineage Tab.

Adding Lineage

Adding Lineage to a spatial table will create the necessary tables and triggers in the database to monitor and archive operations for the selected spatial table. The administrator will also be able to customise how and when Lineage should monitor and archive records.

Three types of triggers are created when Lineage is added to a spatial table, two for logging purposes and one for archiving. These triggers populate Munsys system tables based on the preference selected when Lineage is added or edited.

- MMA_tablename (for example MMA_SP_PARCEL) – this trigger name contains the prefix MMA_(Munsys Monitor Attribute columns) to indicate that the primary function is to log attribute updates. This trigger populates the MUNSYS_MONITOR_LOG table with the following operations:
  - INS – Insert operations
  - UPD – Update operations, based on the columns selected during the implementation
  - DEL – Delete operations
- MMS_tablename (for example MMS_SP_PARCEL)– this trigger name contains the prefix MMS_(Munsys Monitor Spatial update) to indicate that the primary function is to log spatial updates that occurred in the GEOMETRY column. This trigger populates the MUNSYS_MONITOR_LOG table with spatial updates only. This trigger is only created if the user selects the option to log spatial updates explicitly when the Lineage is added or edited. In this case, the SPA (spatial update) operation will be used in the case of a spatial update.
- MARC_tablename (for example MARC_tablename) – this trigger name contains the prefix MARC_(Munsys Archive columns) to indicate that the primary function is to archive records. The MARC trigger populates the archive table associated with a spatial table (for example SP_AD_WATERPIPE) with the archived records. The MARC_tablename trigger populates the SP_AD_tablename with the operations update or delete (as preferenced when Lineage is added or edited).

To add Lineage to a spatial table

1 Select the **Spatial Tables** branch in the **Lineage** tree to view spatial tables with triggers already implemented.

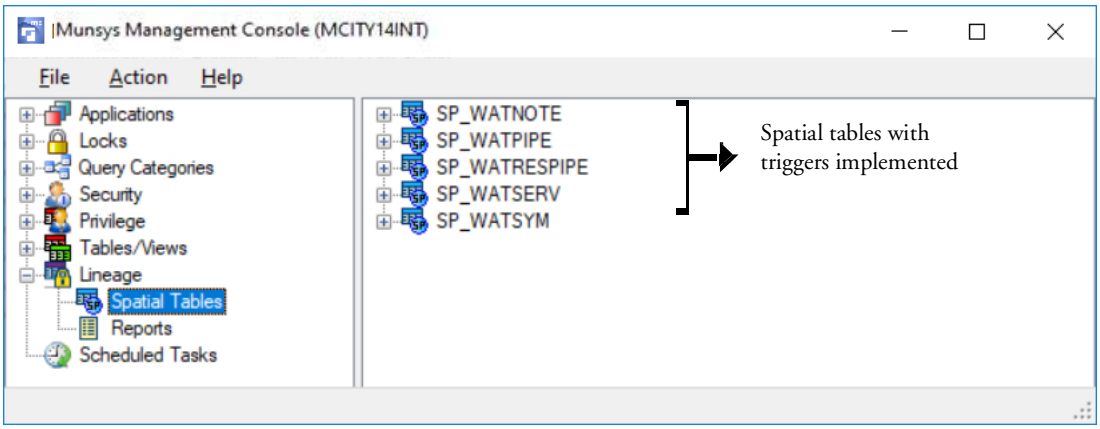


Figure 4 Lineage tree: Spatial Tables

2 To add lineage to a spatial table, select the **Add Lineage...** option that is displayed when you right click the **Spatial Tables** branch of the **Lineage** tree.

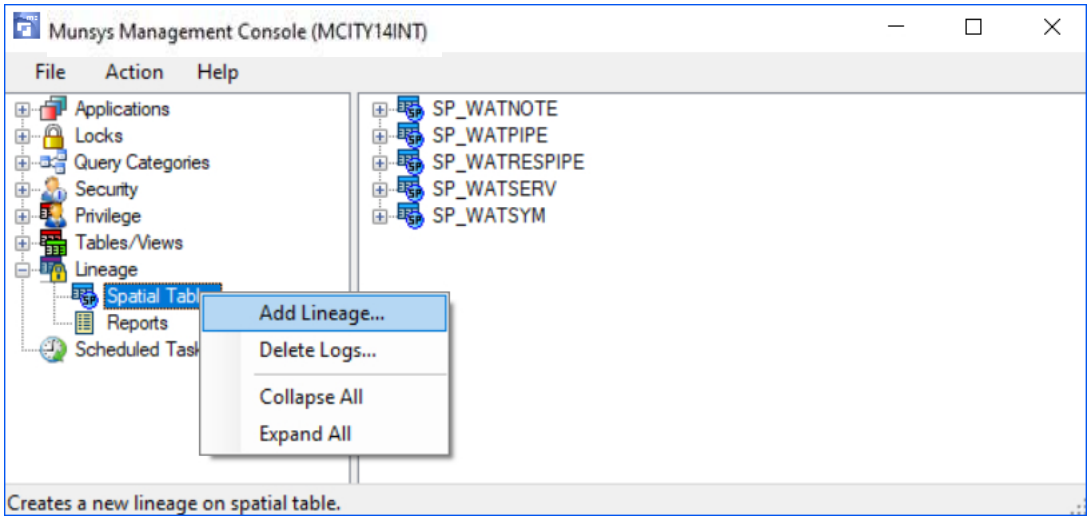


Figure 5 Add Lineage selection

The Add Lineage dialog will appear:

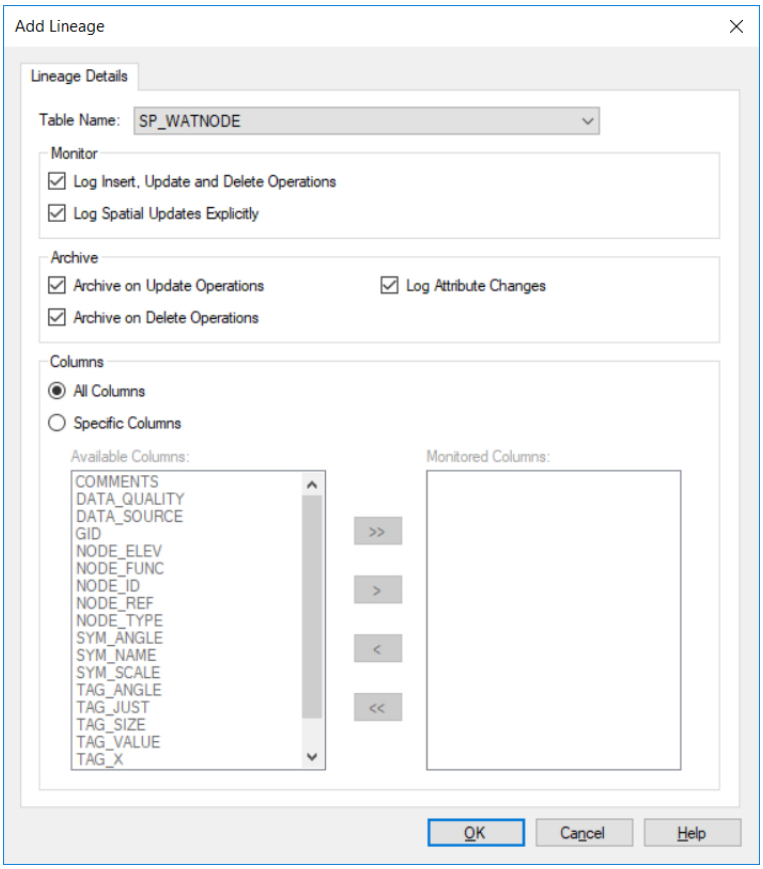


Figure 6 The Add Lineage dialog box

3 In the **Table Name** drop down of the **Add Lineage** dialog box, select a table for which Lineage will be applied. The table selection does not include tables to which Lineage is already applied.

4 In the **Monitor** area, select one or more of the following check boxes (all operations are selected by default):

- Log Insert, Delete and Update Operations – This option logs insert, update and delete operations. Selecting this option creates the MMA_tablename trigger in the database.
- Log Spatial Updates Explicitly – Log Spatial Updates this option logs updates on the GEOMETRY column as an additional operation. The MMS_tablename trigger is created, firing only on the GEOMETRY column when an update occurs. The trigger logs an operation of SPA (spatial update) in the monitor log.

5 In the **Archive** area, select one or more of the following check boxes (all operations are selected by default):

- Archive on Update Operations – this option archives records that have been updated. The MARC_tablename trigger will populate the SP_AD_tablename for update operations. The trigger logs an operation of UPD in the monitor log.
- Archive on Delete Operations – this option archives records that have been deleted. The MARC_tablename trigger will populate the SP_AD_tablename for delete operations. The trigger logs an operation of DEL in the monitor log.

6 Specify the **Columns** that will be monitored and archived in the Lineage.

- All Columns – when this option is selected, all columns will be used, which means that the trigger is not dependent on the structure of the table. The trigger will also fire on the GEOMETRY column (for spatial updates). The All Columns option is the default option.
- Specific Columns – when this option is selected, the Available Columns and Monitored Columns lists, as well as their associated options, become available. The arrow options are used to add or remove columns from the two lists. The columns on which the trigger should fire are moved from the Available Columns list to the Monitored Columns list, as seen in the example below. Note that, the GEOMETRY column is not listed as an available column but the MARC_tablename trigger will automatically fire on this column when a spatial update is made.

Note If a new column is added to a spatial table that has Lineage applied, that column will not automatically be added to the **Monitored Columns** list. It will have to be manually shifted to from **Available Columns** list to the **Monitored Columns** in the **Edit Lineage** dialog box. The new column will then be added to the specified database trigger(s). The column will also automatically be added to the associated archive table.

7 Click **OK**.

The spatial table will be added to the Spatial Tables branch of the Lineage tree. The necessary triggers will also be created in the database to log and archive operations for the selected spatial table. Triggers will be customised to track Lineage only for selected operations and columns as defined in the Add Lineage dialog box upon implementation.

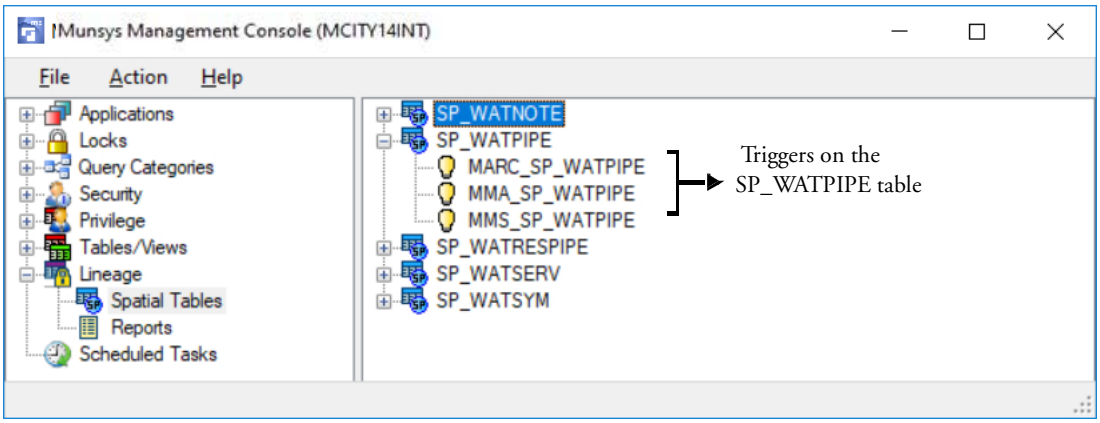


Figure 7 Spatial tables and triggers

Disabling triggers

The current operation status of a trigger is identified by the following prefix symbols:

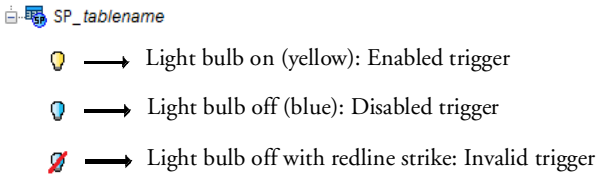


Figure 8 Trigger status symbols

Disabling all triggers on a spatial table

1 To disable all triggers on a spatial table, select the **Disable All Triggers** option on the context-sensitive (right-click) menu that is displayed when you right click on a spatial table in the **Spatial Tables** branch of the **Lineage** tree.

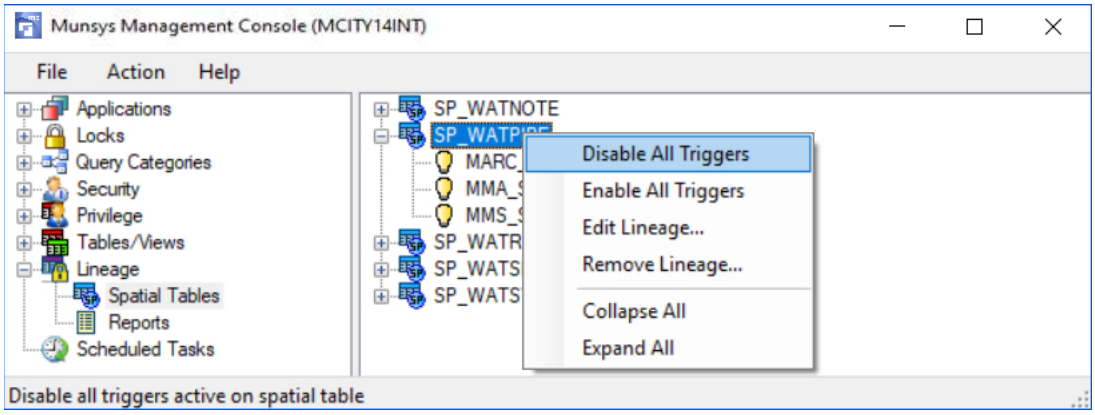


Figure 9 Disable All Triggers

All triggers for the selected spatial table will be disabled.

Disabling a selected trigger

1 To disable only a selected trigger on a spatial table, right click the trigger and select **Disable Trigger** on the context-sensitive (right-click) menu.

The selected trigger will be disabled.

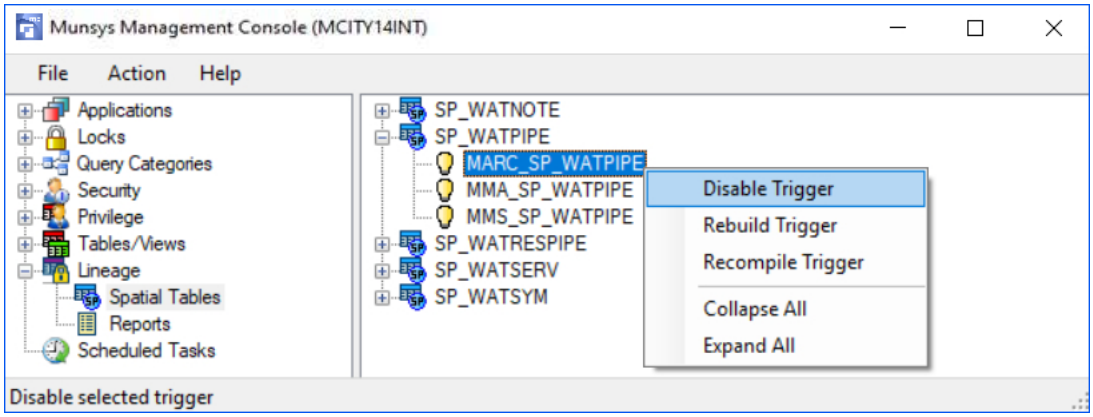


Figure 10 Disable selected trigger

Enabling triggers

The current operation status of a trigger is identified by the following prefix symbols:

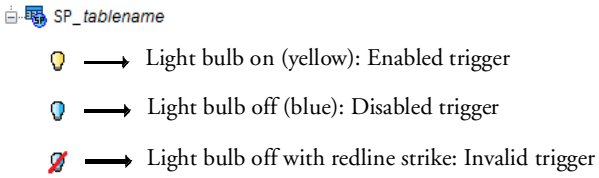


Figure 11 Trigger status symbols

Enabling all triggers on a spatial table

1 To enable ALL triggers on a spatial table, select the **Enable All Triggers** option on the context-sensitive (right-click) menu that is displayed when you right click on a spatial table in the **Spatial Tables** branch of the **Lineage** tree.

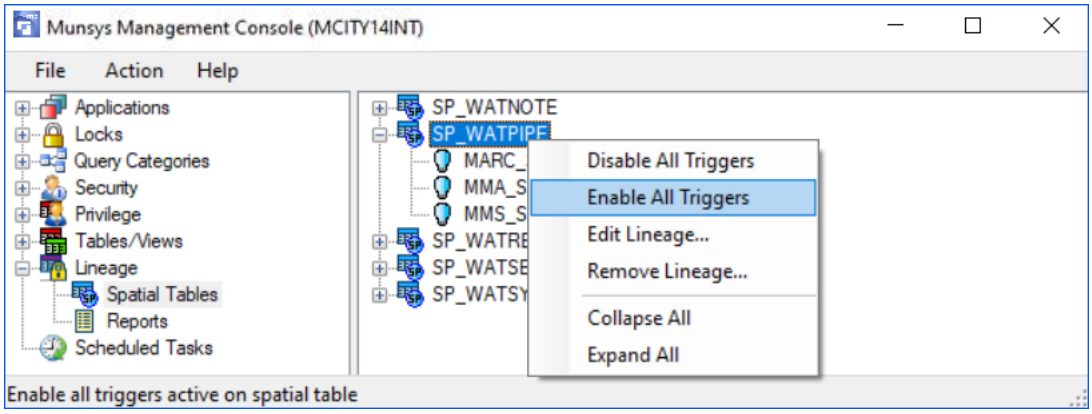


Figure 12 Enable All Triggers

All triggers for the selected spatial table will be enabled.

Enabling a selected trigger

1 To enable only a selected trigger on a spatial table, select the **Enable Trigger** option on the context-sensitive (right-click) menu that is displayed when you right click on a spatial table in the **Spatial Tables** branch of the **Lineage** tree.

The selected trigger will be enabled:

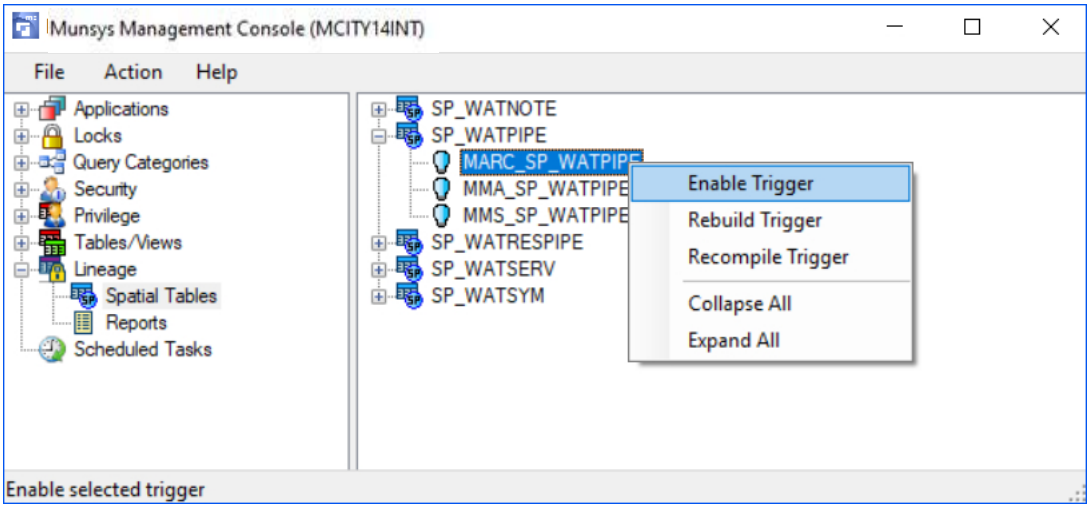


Figure 13 Enable selected trigger

Editing Lineage

1 To edit the Lineage of a spatial table, select the **Edit Lineage...** option on the context-sensitive (right-click) menu that is displayed when you right click on a spatial table in the **Spatial Tables** branch of the **Lineage** tree.

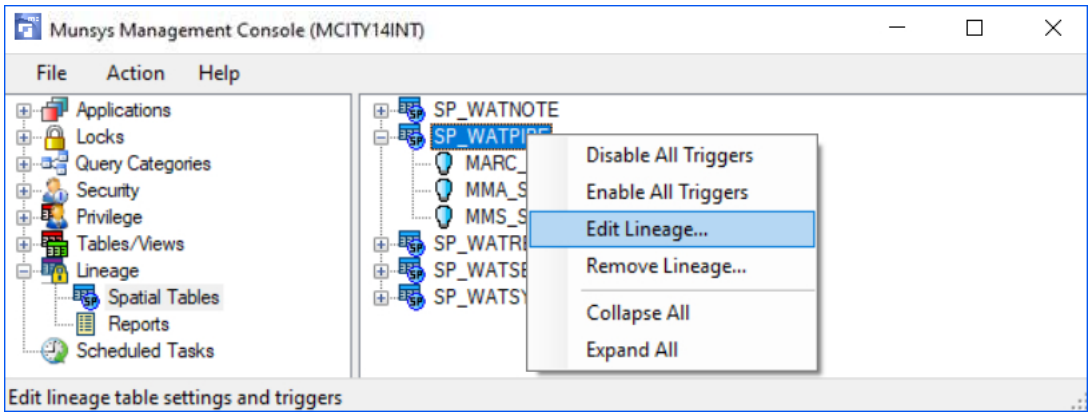


Figure 14 Edit Lineage selection

The Edit Lineage dialog box will appear:

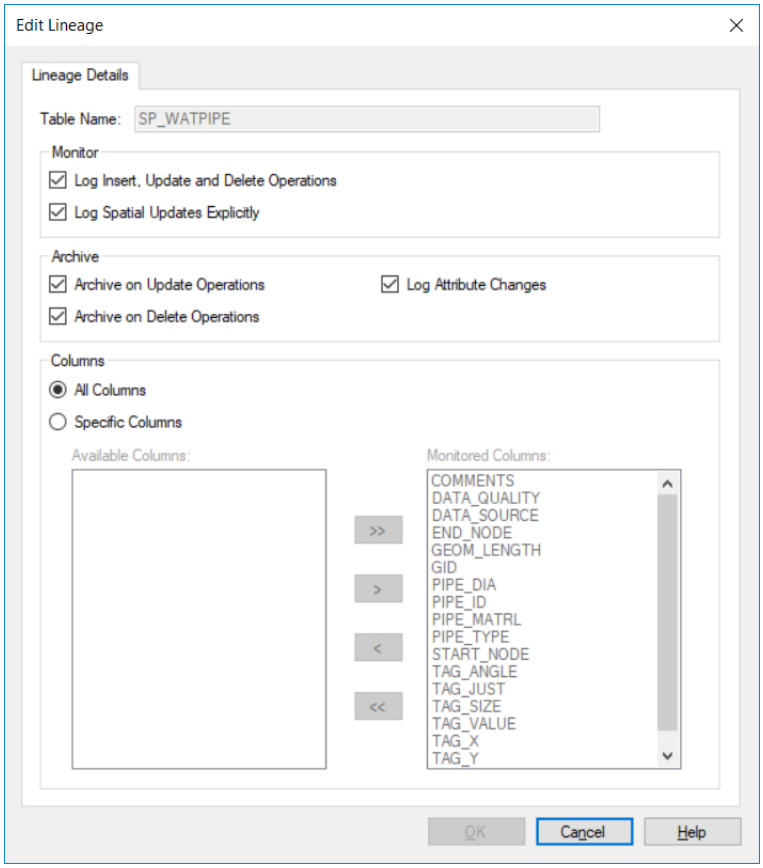


Figure 15 Edit Lineage dialog box

2 Lineage preferences can be edited for the selected spatial table by selecting the **Monitor** and **Archive** check boxes and specifying the **Columns** to which Lineage will apply.

3 Click **OK** to apply any changes.

Removing Lineage

1 To remove the Lineage of a spatial table, select the **Remove Lineage...** option on the context-sensitive (right-click) menu that is displayed when you right click on a spatial table in the **Spatial Tables** branch of the **Lineage** tree.

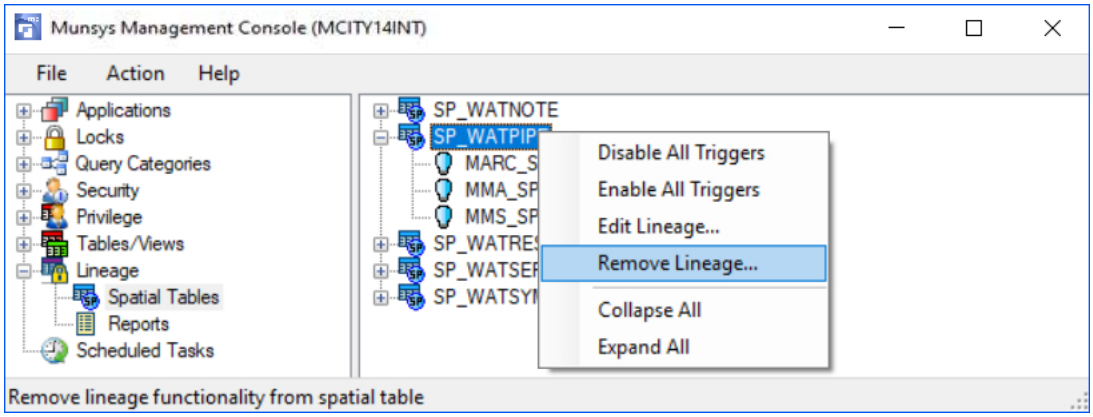


Figure 16 Remove Lineage

The Remove Lineage Confirmation dialog box will appear.

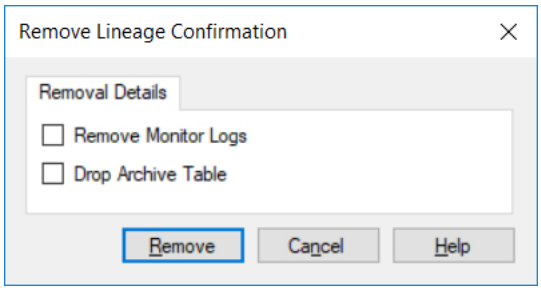


Figure 17 Remove Lineage Confirmation dialog box

2 Select the check boxes to also remove the monitor logs and associated archive table for the selected spatial table.

3 Click **Remove**.

Recompiling triggers

1 To recompile a trigger, select the **Recompile Trigger** option on the context-sensitive (right-click) menu that is displayed when you right click on a trigger that belongs to spatial table in the **Spatial Tables** branch of the **Lineage** tree.

This option will recompile the currently selected trigger. This is useful if the structure of the related table has changed and the trigger needs to be recompiled.

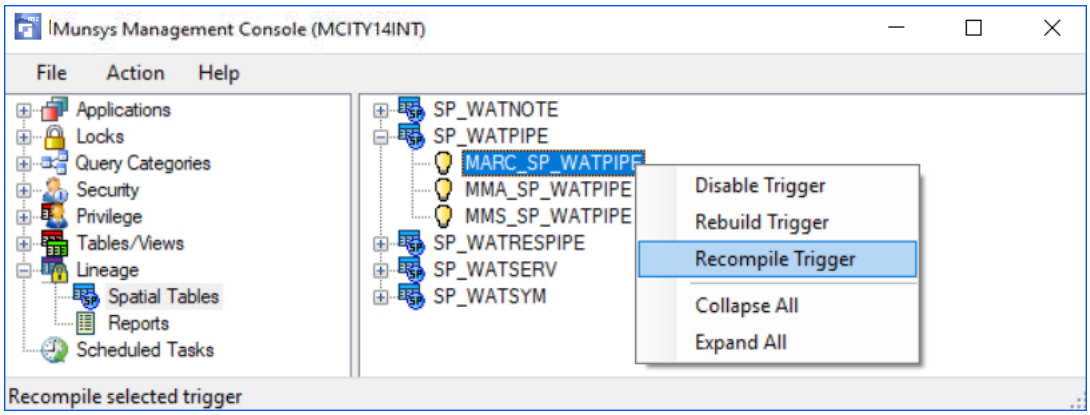


Figure 18 Recompile selected trigger

Rebuilding triggers

1 To rebuild a trigger, select the **Rebuild Trigger** option on the context-sensitive (right-click) menu that is displayed when you right click on a trigger that belongs to spatial table in the **Spatial Tables** branch of the **Lineage** tree.

Rebuild Trigger will rebuild and compile the trigger and associated views. The trigger will be rebuilt using the same process as when creating a new trigger.

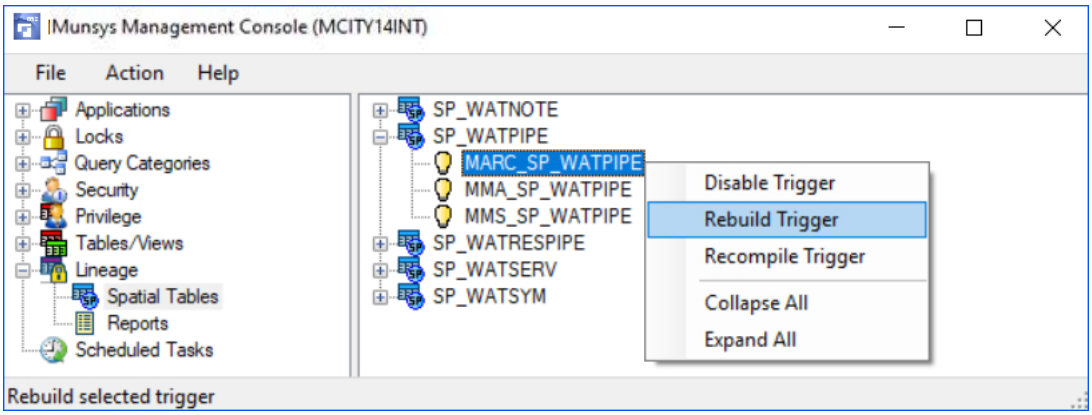


Figure 19 Rebuild selected trigger

Deleting monitor logs

The delete logs function can be used to maintain the amount of records in the MUNSYS_MONITOR_LOG.

1 To delete monitor logs, select the **Delete Logs...** option on the context-sensitive (right-click) menu that is displayed when you right click the **Spatial Tables** branch of the **Lineage** tree.

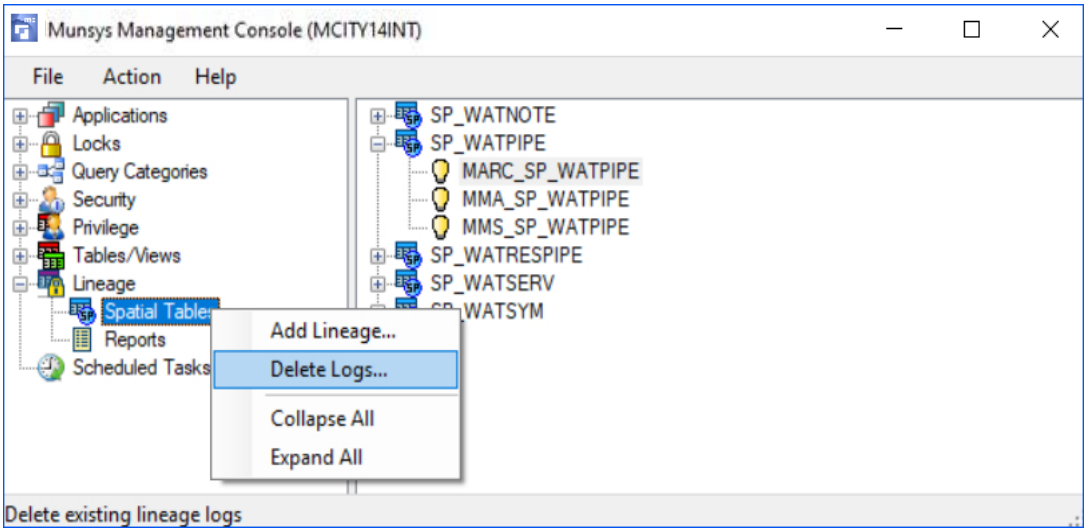


Figure 20 Spatial Tables: Delete Logs

2 The **Delete Logs** dialog box will appear:

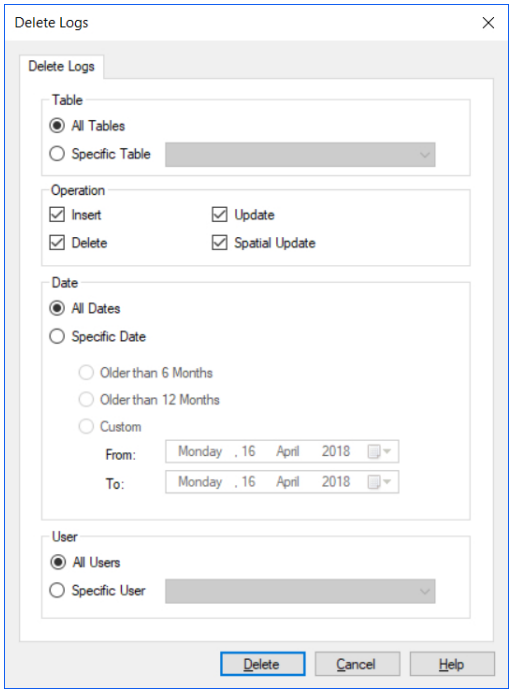


Figure 21 The Delete Logs dialog box

Note Different options are available to filter the records in this table before they are deleted. Records can be deleted from specific tables, by specific operations, according to specific dates and by specific user.

3 In the **Table** area, do one of the following:

- Select the **All Tables** option to delete all the records for all the tables
- Select a spatial table name from the **Specific Table** drop down list to delete records for a specific spatial table.

4 In the **Operation** area, select one or more of the following check boxes (all operations are selected by default):

- **Insert –** deletes all operations of the type INS (Insert)
- **Delete –** deletes all operations of the type DEL (Delete)
- **Update –** deletes all operations of the type UPD (Update)
- **Spatial Update –** deletes all operations of the type SPA (spatial update)

5 In the **Date** area, do one of the following:

- Select the **All Dates** option to delete records from all dates
- Select the **Specific Date** option to apply one of the following date filters:
  - **Older than 6 months** – will select logs older than 6 months from the current date.
  - **Older than 12 months** – will select logs older than 12 months from the current date.
  - **Custom** – specify your own dates by clicking on the calendar drop down and navigating to a desired From and To date. The default From and To dates are set to the current system date.

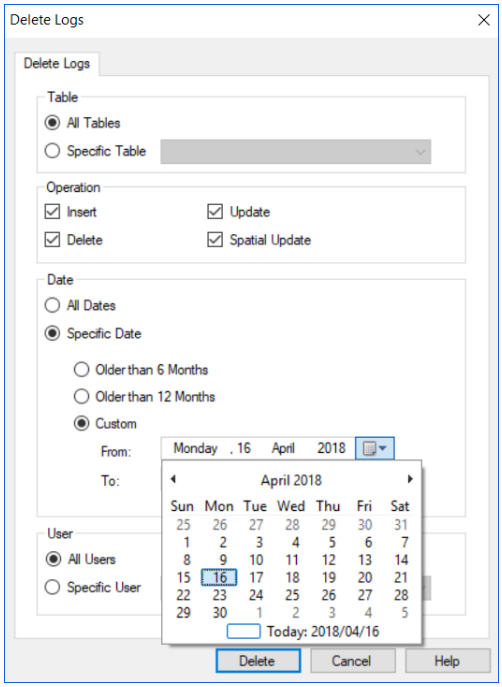


Figure 22 Delete Logs: Date Custom

6 In the **User** area, do one of the following:

- Select the **All Users** option to delete records of all users
- To delete records for a specific user, select the **Specific User** option, and then select a user from the list.

7 When you have selected all the appropriate filters, click **Delete**.

8 The Log delete confirmation dialog box is displayed, prompting for confirmation to delete the records as specified.

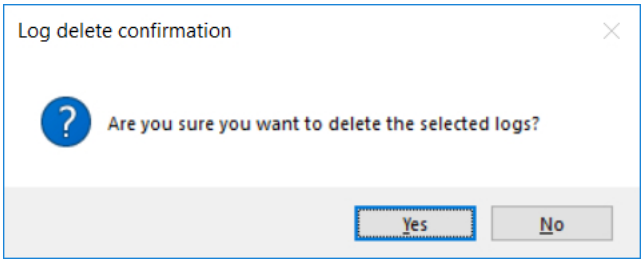


Figure 23 The Log Delete Confirmation dialog box

9 Click **Yes.**

The Logs Deleted dialog box will appear when the selected logs have been successfully deleted.

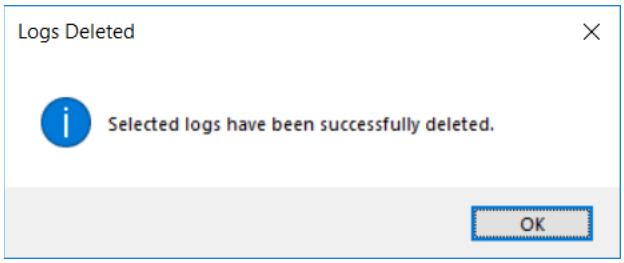


Figure 24 The Logs Deleted dialog box

Running reports

Reports can be run to query the database for monitor or usage purposes:

- Reports of type **Monitor** can be used to provide basic or detailed reporting, based on operation type, user, table name and date of the operation.
- Reports of type **Usage** can be used to show the license usage of the complete range of Munsys applications, providing the administrator with the necessary records to evaluate the license usage and detail of users logging on to the Oracle database and using any of the Munsys applications.

All report queries are stored in the MUNSYS_MONITOR_REPORT table in the database. Once a report has been run, the results can be saved to a CSV or text file.

Note The number of records output to the Munsys Management Console display for any of the monitor or usage reports is limited to 10,000. If any of the reports return more than 10,000 records, a message is produced indicating that the report process has stopped. The Administrator can continue to execute the report directly from the database from the SQL command prompt and spool it to a file.

To run a report

1 Select the **Reports** branch of the **Lineage** tree to view the available reports.

A list of **Monitor** and **Usage** reports are displayed. The **Description** field is generally descriptive of the specific query operation.

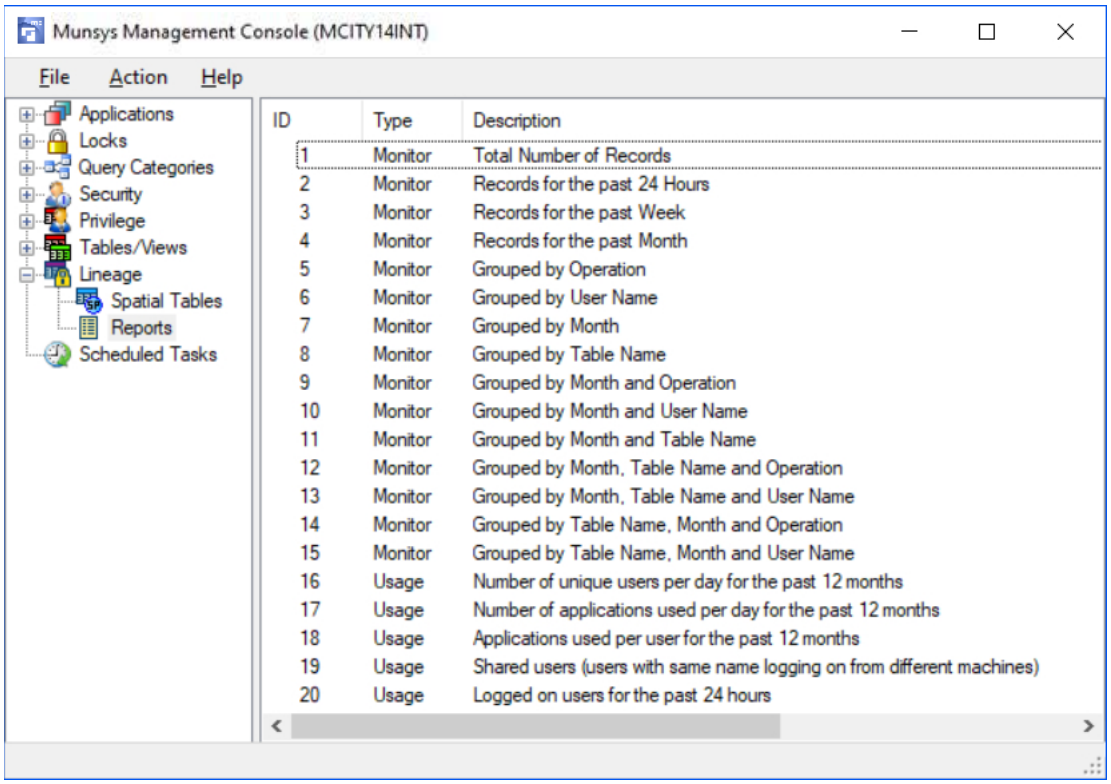


Figure 25 Lineage Tree: Reports by Type

2 To run a report, **right click** on an available report and select the **Run Report** option that appears on the context-sensitive (right-click) menu.

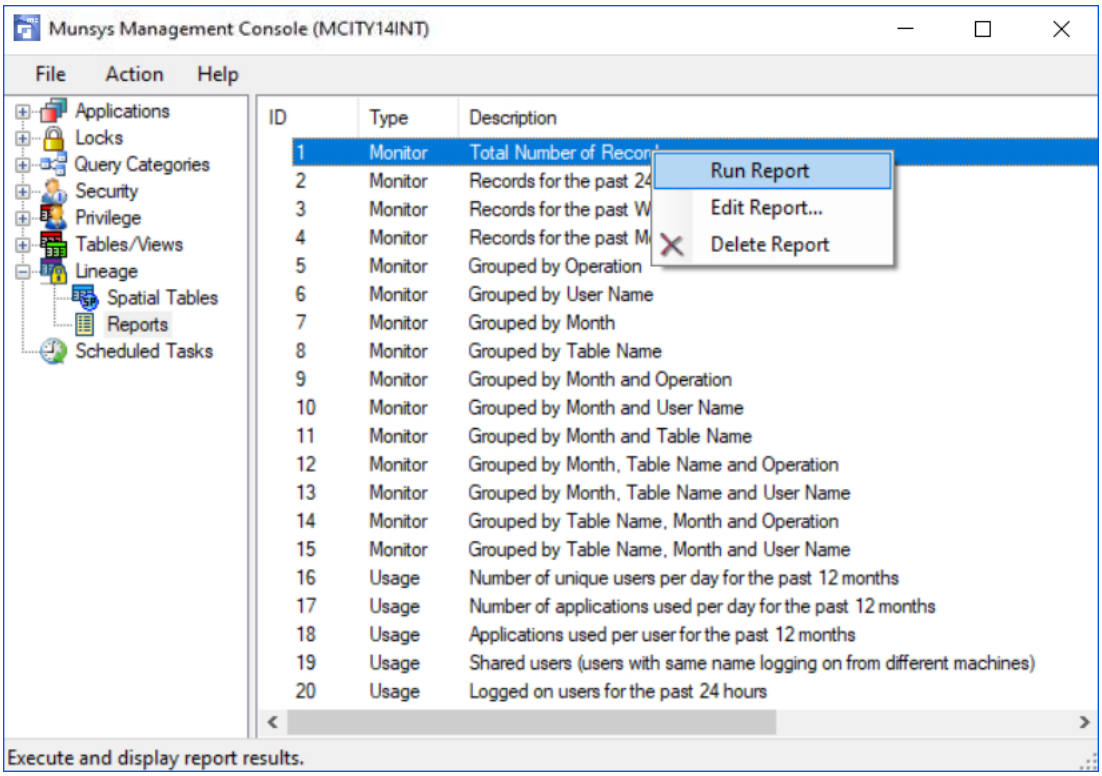


Figure 26 Run Report selection

A dialog box will appear for the selected report displaying the results of the query.

For example, the report dialog box below displays Report Details for the total number of Records recorded in the MUNSYS_MONITOR_LOG grouped by each OPEPRATION.

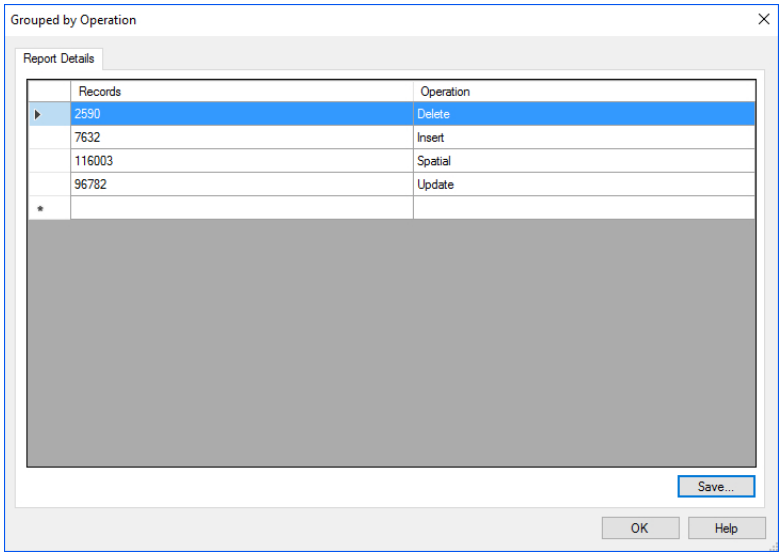


Figure 27 The run report dialog box

Once the report has been run, the results can be saved.

3 Click **Save...** to save the results to a folder of your choice. Results can be saved as a CSV or text file.

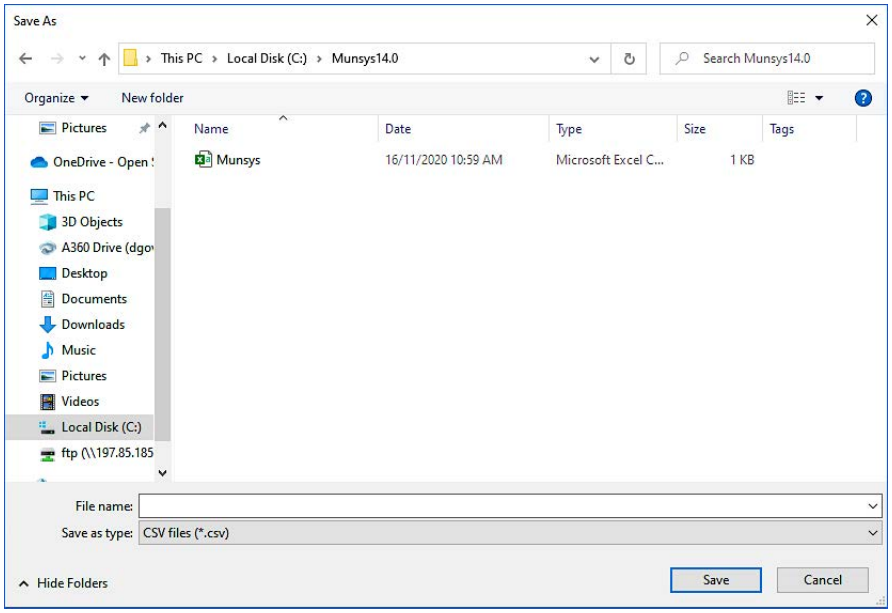


Figure 28 Saving a report

Editing reports

Reports can be edited to specific user requirements so that only the desired information is queried from the database.

1 To edit a report, **right click** on an available report and select the **Edit Report** option that appears on the **context-sensitive** (right-click) menu.

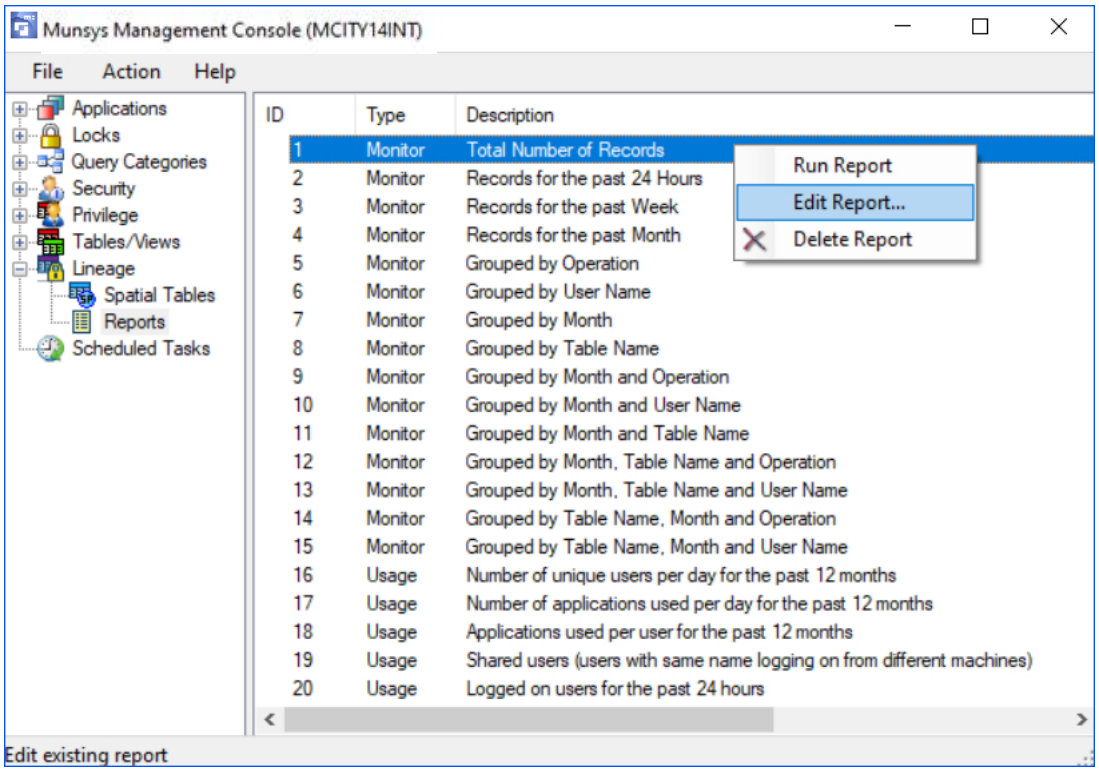


Figure 29 Edit Report selection

The Edit Report dialog box will appear displaying the Report Type, Report Description and Report SQL (statement) for the current report.

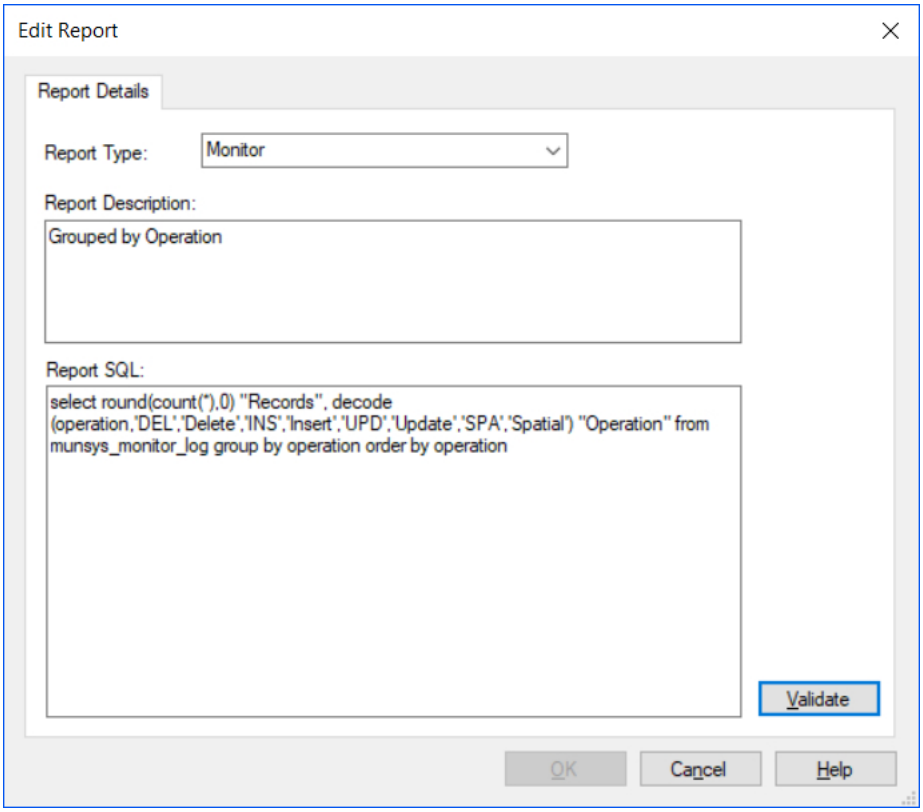


Figure 30 The Edit Report dialog box

2 In the **Edit Report** dialog box, the **Report Type, Report Description** and **Report SQL** are editable.

- **Report Type**: Click on the drop down menu to change the report type.
- **Report Description:** Click inside the text box to edit or change the report description.
- **Report SQL:** Click inside the text box to edit or change the report SQL statement.

Note If the **Report SQL** is modified the statement will need to pass the **Validate** test before the changes can be applied in the **Edit Report** dialog.

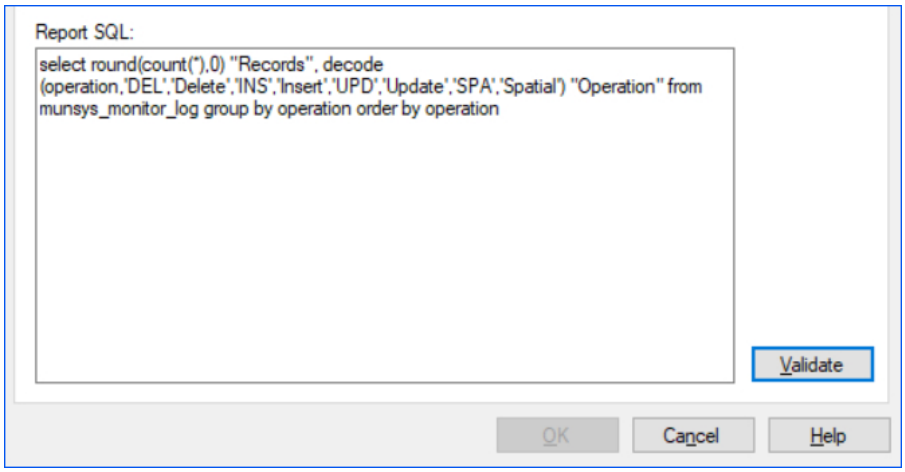


Figure 31 The Report SQL text box

- Click **Validate** to test the changes.

If the SQL is valid, the **Report Validate Success** dialog box will appear.

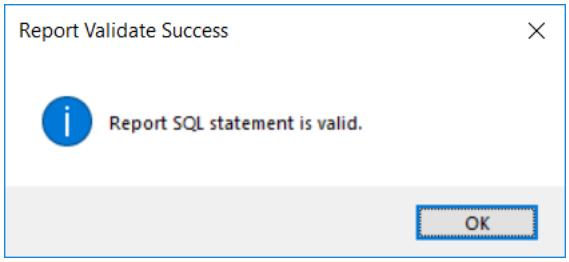


Figure 32 The Report Validate Success dialog box

- Click **OK** to return to the **Edit Report** dialog box.

The Click OK button in the Edit Report dialog box is now active again.

- Click **OK** in the **Edit Report** dialog box to apply any SQL statement changes, or continue making edits in the **Edit Report** dialog box.

Adding reports

Reports can be created to query the database according to customised requirements.

1 To add a new report, select the **Add Report...** option that is displayed when you right click the **Reports** branch of the Lineage tree.

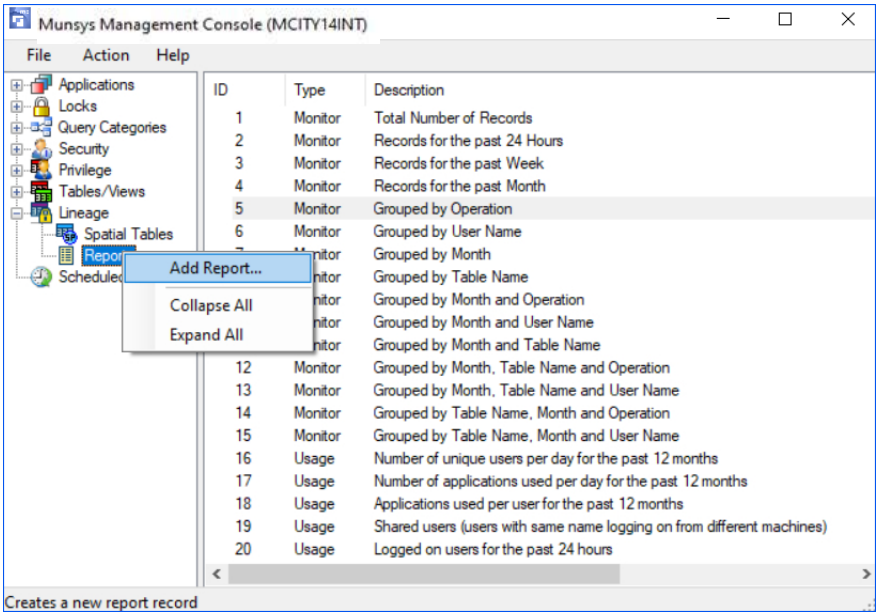


Figure 33 Add Report selection

The Add Report dialog box will appear:

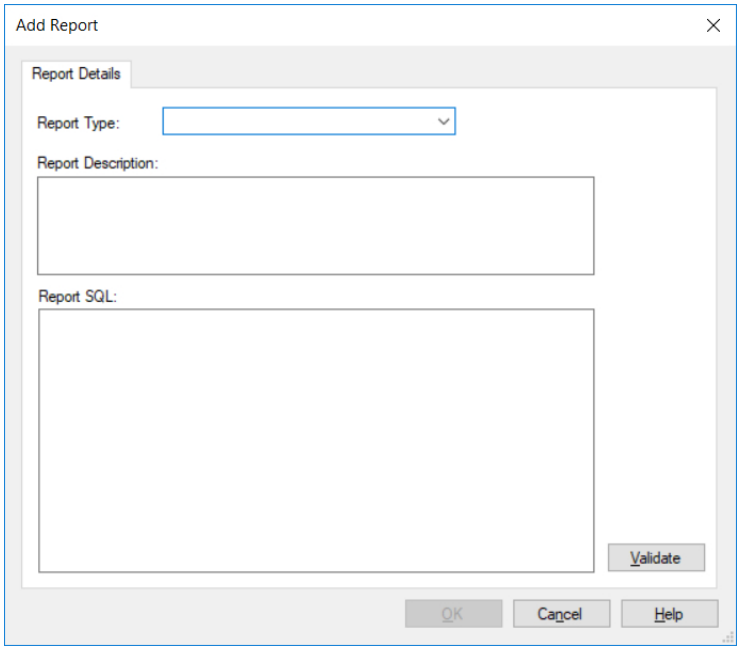


Figure 34 The Add Report dialog box

2 In the **Report Type** drop down of the **Add Report** dialog box, select a report type or create a new report type/category by typing in the drop down area.

3 In the **Report Description** area, briefly describe the report. For example, “Grouped by Month and Operation”.

4 In the **Report SQL** area, include the SQL statement that will be used to generate the report.

The OK button in the Add Report dialog box will only become available once the Report SQL statement has passed the Validate text.

5 Click on the **Validate** button to test the validity of the SQL statement in the **Report SQL** field**.**

If the SQL statement is valid, the **Report Validate Success** dialog box will appear.

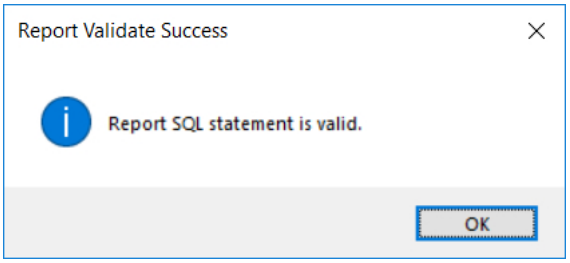


Figure 35 The Report Validate Success dialog box

6 Click **OK** in the **Report Validate Success** dialog box.

7 Click **OK** in the **Add Report** dialog box to create the report.

The report will appear in the right hand side window when the **Reports** branch of the **Lineage** tree is selected.

Deleting reports

Reports can be deleted if they are unused or no longer required.

1 To **Delete** a report, **right click** on a report and select the **Delete Report** option that appears on the context-sensitive (right-click) menu.

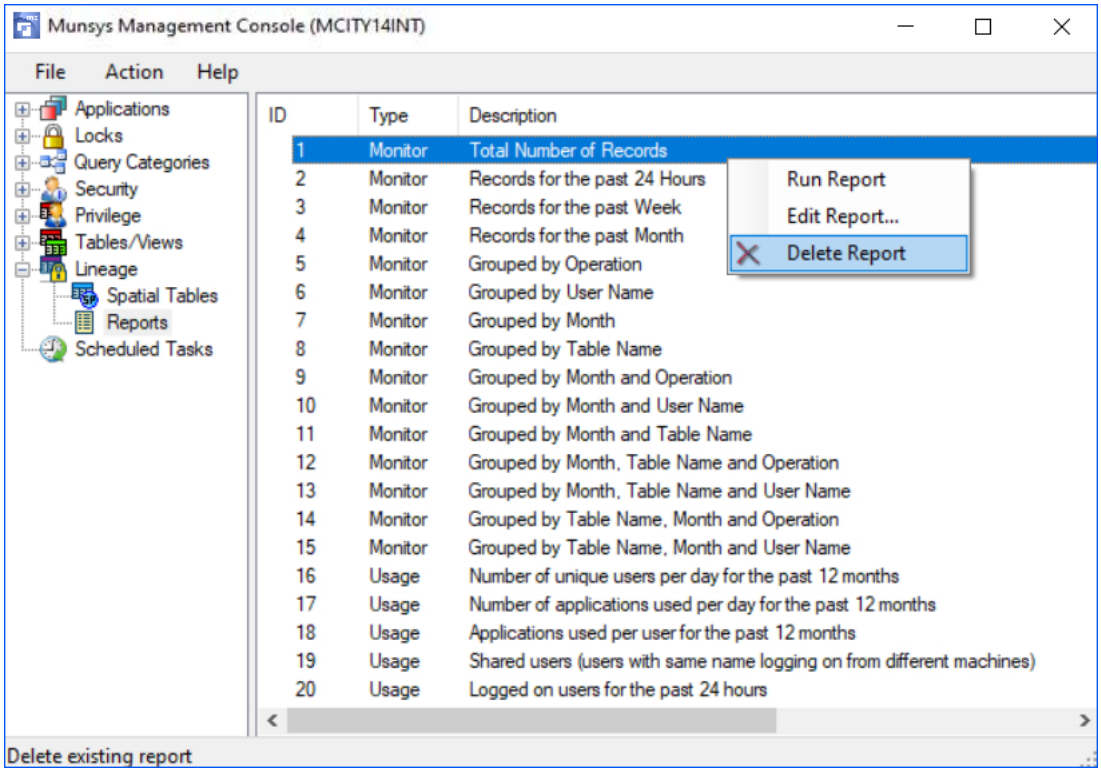
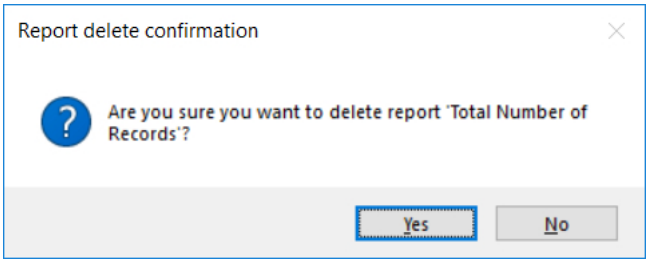


Figure 36 Delete Report selection

The **Report delete confirmation** dialog box will appear:



2 Click **Yes** in the **Report delete confirmation** dialog box to delete the selected report.

The selected report will be deleted.

Chapter 12
Munsys Scheduled Tasks

Introduction

Munsys allows users to set up and manage regular scheduled database tasks. The scheduled task will run a Procedure and can pass a range of Parameters with their values (Key / Value Pairs) to the procedure. Tasks can be scheduled to run on a Daily, Weekly or Monthly basis.

This chapter is divided into the following sections for easy reference:

- Adding Scheduled Task
- Editing Scheduled Tasks
- Deleting Scheduled Tasks

Note A Schedule Task has been setup to remove all locks on objects which are locked around 1am every morning.

Adding Scheduled Tasks

Scheduled Tasks can be created to automate the running of regularly scheduled tasks.

Clicking on Scheduled tasks in the Munsys Management Console tree displays existing scheduled tasks (if any).

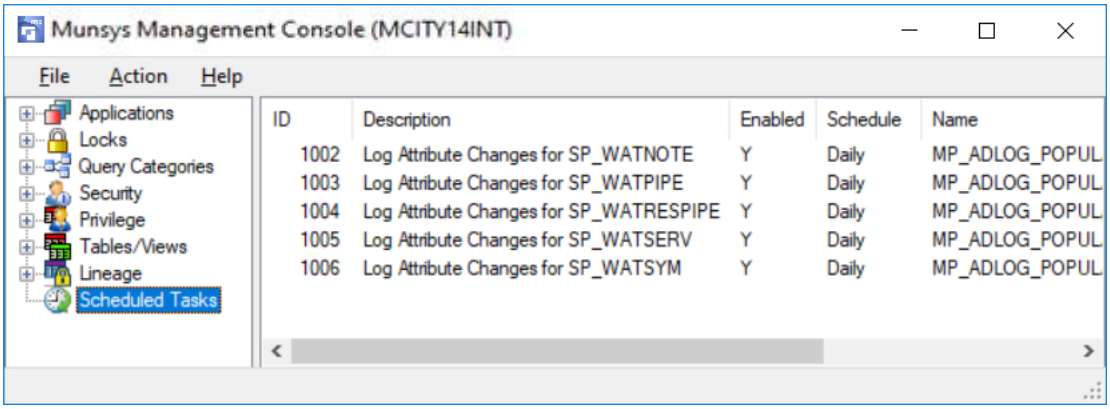


Figure 1 Munsys Management Console: Scheduled Tasks

1 To add a new Scheduled Task right-click on the **Scheduled Tasks** node in the tree or using the **Action** menu or context-sensitive (right-click) menu that is activated when an item is selected and select the **Add Scheduled Task**...option.

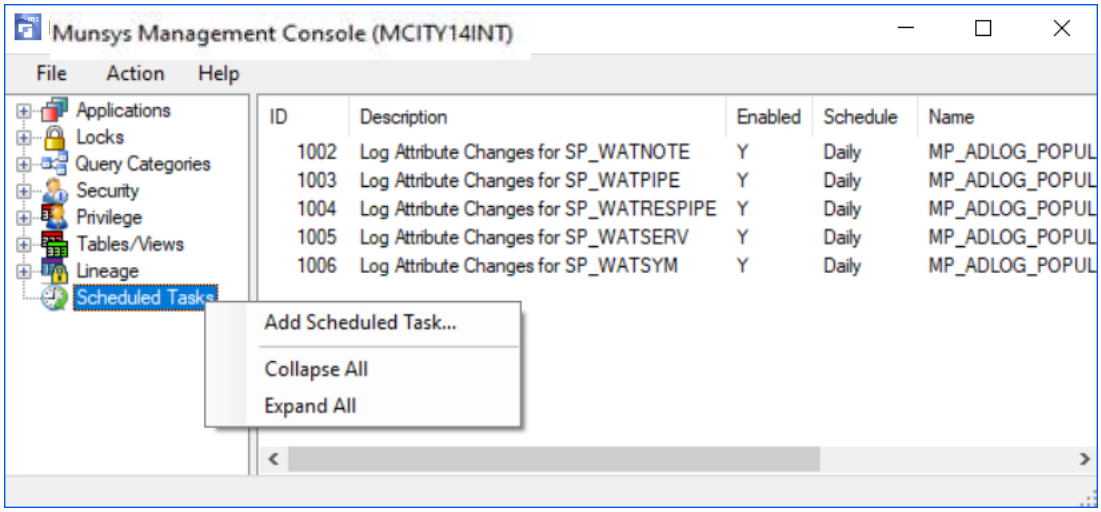


Figure 2 Add Scheduled Task

The Add Scheduled Task dialog is displayed:

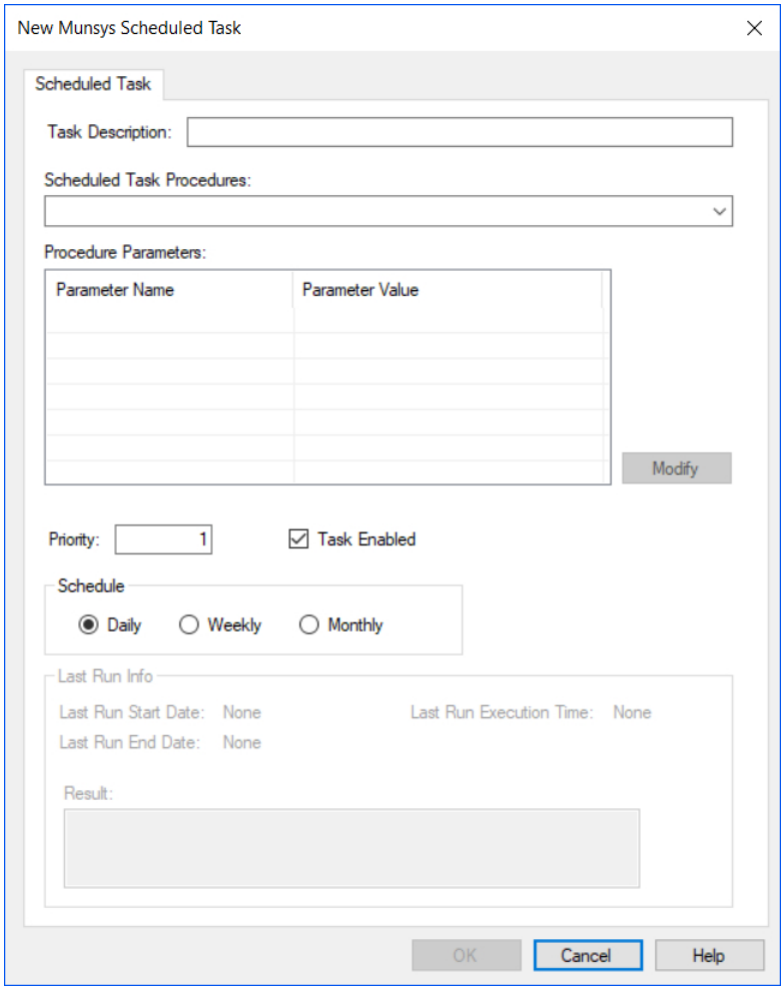


Figure 3 Add Scheduled Task dialog box

2 In the Scheduled **Task Description** area, briefly describe the Scheduled Task. For example, 'Run Asset Management Update'.

3 In the **Scheduled Task Procedures** drop down of the Add Scheduled Task dialog box, select a Scheduled Task Procedure. These are procedures that are part of the MUN_SCHEDULED_TASKS_PACKAGE. To add new Scheduled Tasks Procedures this package must be edited. Please contact Open Spatial before editing the package.

4 In the **Scheduled Task Parameters** area, a list of Parameters will be loaded depending on the selected Procedure. Modify the Parameters to obtain the desired outcome.

5 Set the **Priority** Value – 1 is the default value.

6 If all the values are correct, select the **Task Enabled** option.

7 Set the Task interval: **Daily**, **Weekly** or **Monthly**.

8 Click **OK** in the Add Scheduled Task dialog box to create the Scheduled Task.

Editing Scheduled Tasks

Scheduled Tasks can be run to facilitate easy and regular maintenance and management jobs in Oracle.

1 To edit a Scheduled Task, right click on an available Scheduled Task and select the **Edit Scheduled Task** option that appears on the context-sensitive (right-click) menu.

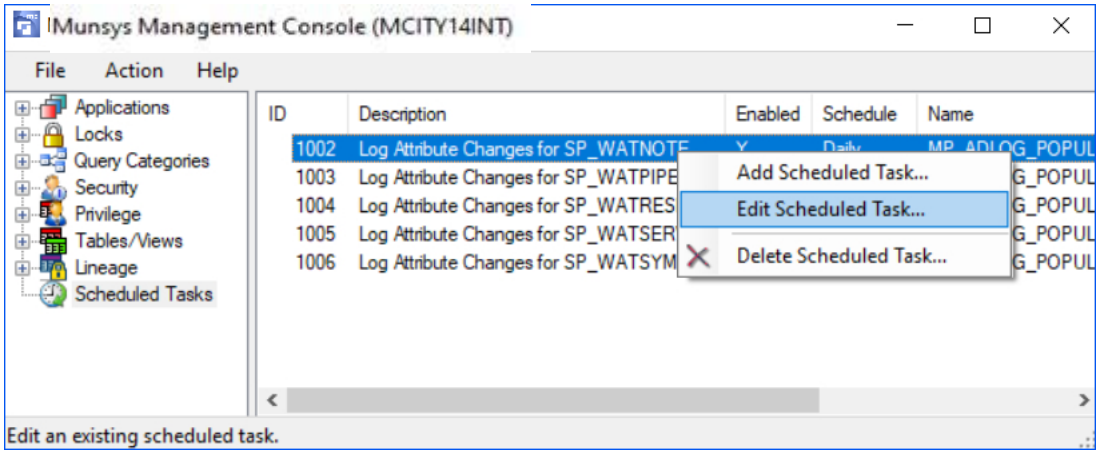


Figure 4 Edit Scheduled Task

The Edit Munsys Scheduled Task dialog is displayed:

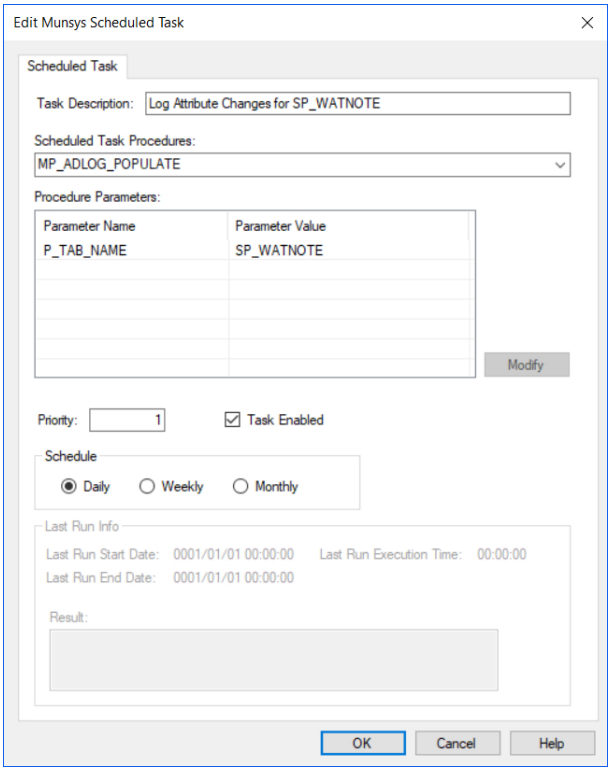


Figure 5 Edit Munsys Scheduled Task

2 In the Edit Scheduled Task dialog box, the Scheduled Task parameters are editable.

- **Task Description**: Click inside the text box to edit or change the Scheduled Task description.
- **Task Procedure**: Click on the drop-down menu to change the Scheduled Task Procedure.
- **Procedure Parameters**: Click on the parameters and then click Modify to edit or change the Scheduled Task Parameters.
- **Priority** – If more than one Scheduled task is available for a table, this will allow users to set the order in which the tasks are executed.
- **Task Enabled** – Allow users to enable or disable a scheduled task.
- **Schedule** – Allows the task to be run on a Daily, Weekly or Monthly

3 Click **OK** to save changes to the Edit Scheduled Task.

Deleting Scheduled Tasks

Custom Scheduled Tasks can be deleted if they are unused or no longer required. Some tasks are managed by other tools with the Munsys application and these cannot be deleted.

1 To Delete a Custom Scheduled Task, right click on a **Scheduled Task** and select the **Delete Scheduled Task** option that appears on the context-sensitive (right-click) menu.

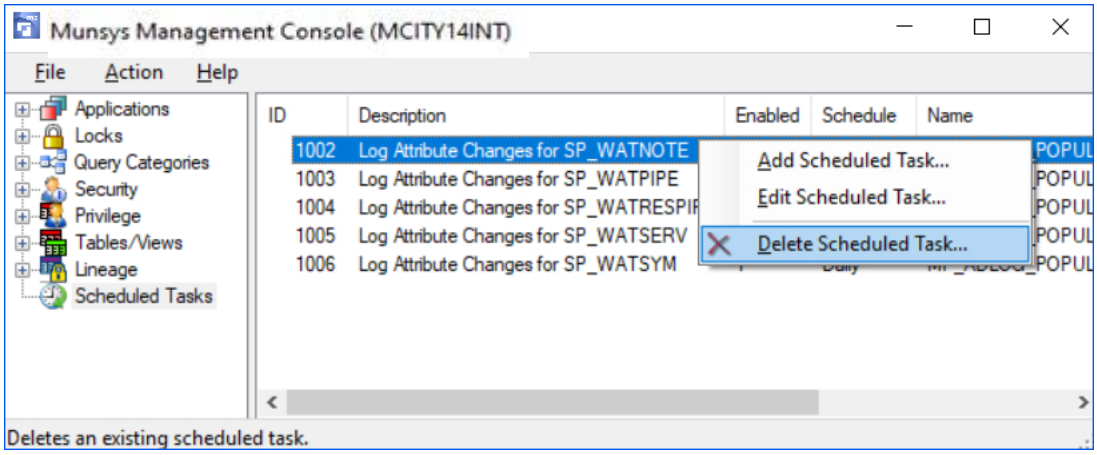


Figure 6 Delete Scheduled Task

The Delete Scheduled Task confirmation dialog box will appear:

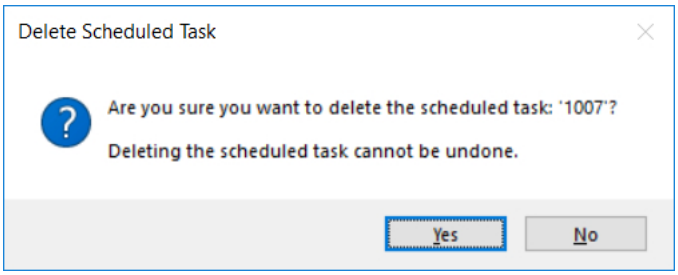


Figure 7 Delete Scheduled Task notification

If the task you selected to delete cannot be deleted from this dialog, a message will be shown indicating where to edit that scheduled task.

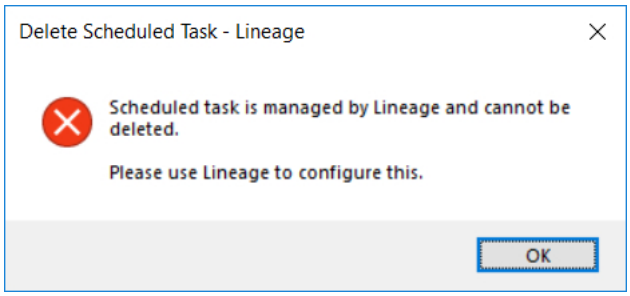


Figure 8 Cannot Delete Scheduled Task notification

Index

S

T

U

V