MI:Explore version 5.3

MI:Explore Configuration Guide

GRANTA
MATERIAL INTELLIGENCE

GRANTA MI™ is the leading system for materials information management in engineering organizations. It enables you to control, analyze, and securely share critical corporate data on materials and processes, managing the materials information lifecycle.

www.grantadesign.com

Release notes, documentation, and Knowledge Articles for the current and all previous GRANTA MI releases are all available on the Granta Support website. Go to www.grantadesign.com and click SIGN IN to log into your My Granta page, then click **Documentation**.

We welcome your feedback on this document. Please let us know if anything is unclear, if you spot an error, or have an idea for new content, by emailing docs@grantadesign.com

Document version: MIX5.3/01
Published: July 2018

Table of contents

1 Introduction

MI:Explore 5.3 was released with GRANTA MI version 11 update 2. This document is aimed at GRANTA MI administrators or System Administrators who need to carry out the necessary MI:Explore configuration tasks.

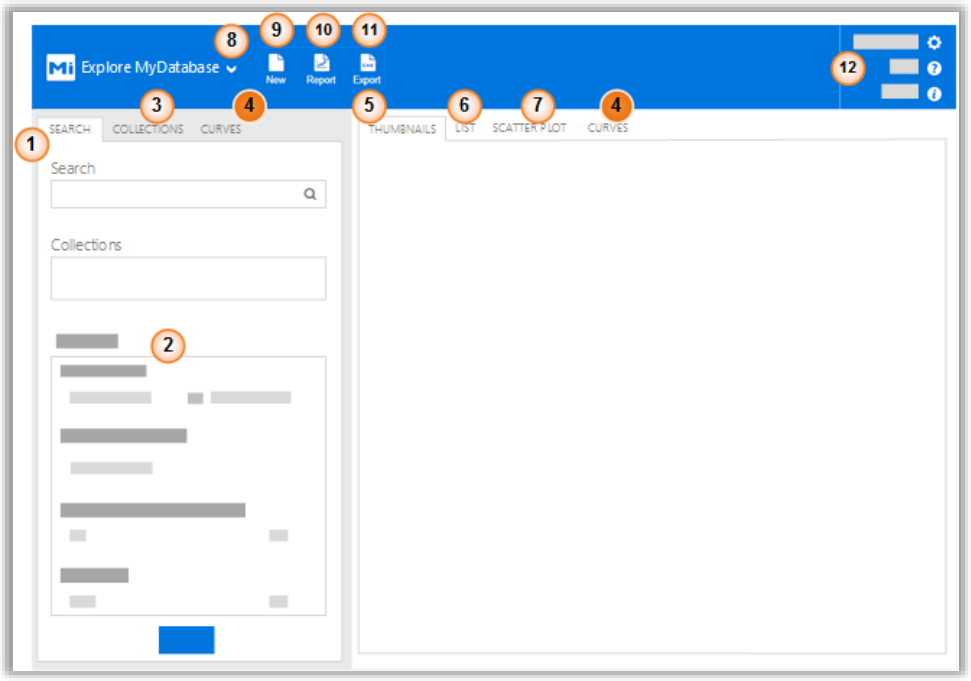
For information on how to install MI:Explore using the Installation Manager, see the *GRANTA MI Installation Guide*.

For help on how to use MI:Explore, click the Help button in the top right corner of the application browser window, or click here to open the help in your browser.

2 About MI:Explore

MI:Explore is a web application for finding, visualizing, and entering materials information in a GRANTA MI database, and quickly performing key tasks such as filtering, comparing, and plotting.

Highly configurable via JSON configuration files, different MI:Explore data views can be provided for different user groups in your company, delivering only the data and tools relevant to that user community. Users looking for the full set of tools to query, browse, report on, edit, import, and export data can use the general MI:Viewer web browser user interface.



1. **Text search** – find all records that include the specified term in their name or in any of their text attributes, and view them on THUMBNAILS, LIST, SCATTER PLOT, CURVES tabs
2. **Search filters** – narrow down search results by applying filters based on attribute values
3. **Collections** – group and quickly access 'favorite' records, or records used in a particular project, or with a particular set of properties
4. **Curves** – plot functional data curves from the data in the search results
5. **Thumbnails tab** – search results displayed in an image thumbnail gallery
6. **List tab** – view search results displayed in a list
7. **Scatter plot tab** – view search results displayed on a scatter (bubble) plot
8. **Switch data view** – change database, table, and layout, and edit configuration settings
9. **New** – create new records
10. **Report** – generate reports from the selected records
11. **Export** – export data in a range of different formats for use in CAE systems
12. **Preferences** – change the application theme (colors) and unit settings

See *Section 3, Application configuration options*, to find out how you can configure MI:Explore applications with different data and tools.

2.1 Integration with other applications

MI:Explore can be integrated with other Granta enterprise applications:

- **"Where Used" queries**: users can query instances of assignment of records to parts in Teamcenter®, where the assignments were made in the NX CAD application using MI:Materials Gateway. See Section *8*.
- **MI:Workflow management**: users can launch the Workflow Manager from a selected record within MI:Explore, and view the active workflows that are being progressed on that record. To access Workflow Manager from MI:Explore; see Section *7*.
- **MI:Viewer**: users can open datasheets in MI:Viewer from within an MI:Explore datasheet, allowing them to view and edit the full range of MI data types; see Section *6*.

2.2 Supported data types

The following GRANTA MI attribute types can be used when searching, viewing, plotting, and editing data in MI:Explore:

MI Attribute Type		Search pane	List tab	Scatter Plot tab	Curves tab	Datasheet view	Datasheet edit
Numerical	Integer	Y	Y	Y		Y	Y
	Point (single-value)	Y	Y	Y		Y	Y
	Point (multi-value)	Y	Y			Y	
	Range	Y	Y	Y		Y	Y
Text	Short Text	Y	Y			Y	Y
	Long Text	Y	Y			Y	Y
	Discrete	Y	Y			Y	Y
Functional	Float Functional (point)	Exists	Exists		Y	Exists	
	Float Functional (range)	Exists	Exists		Y	Exists	
	Equations & Logic						
	Discrete Functional						
Media	Picture					Y	Y
	Hyperlink	Exists	Y			Y	Y
	File (embedded media)	Exists	Y			Y	Y
Other	Date	Y	Y	Y		Y	Y
	Logical	Y	Y			Y	Y
	Tabular					Y	Y

In the preceding table:

- "Y" means you can search on, filter, plot, view, or edit values of this attribute type
- "Exists" means you can search and filter on whether the attribute exists on the Search pane, but not on its value; on datasheets, you can see that the attribute exists and view the curves, but you can't edit the data points.
- A blank cell means that attributes of this type are not supported for the specified functionality

3 Application configuration options

The process of configuring MI:Explore will require input from a GRANTA MI expert to decide appropriate values for some settings, and it may also be necessary to modify your GRANTA MI database, for example, to create new layouts and/or subsets for use in MI:Explore.

Application configuration settings for MI:Explore are specified in a number of configuration files in JSON (JavaScript Object Notation) format.

preferences.json specifies system options, and the URLs for other GRANTA MI applications that can be integrated with MI:Explore, allowing users to:

- set the application unit system; see section *3.16*
- change the formatting of number separators; see section *3.17*
- change the language of the user interface; see section *3.18*
- set the path for custom configuration files; see section *5*
- open MI:Viewer datasheets from MI:Explore; see section *6*
- access Workflow Manager from MI:Explore; see section *7*

preferences.json is located in the settings folder within the MI:Explore installation folder.

Other MI:Explore configuration settings are set in the JSON file stored with each database. These settings determine the data view visible in MI:Explore, and are described in the rest of this section. A number of alternative configurations may be defined in these files, allowing different databases, tables, and attributes to be searched. Users can be permitted to change between data views using the Switch data view dialog (see section *4.1*), or by entering the key in the URL.

See Section *5, Managing configuration files with MI:Admin* for information about loading configuration files into MI:Admin. See Section *9, MI:Explore configuration file reference*, for reference information about all the supported mandatory and optional settings. See Section *10, Sample configuration file*, for a sample JSON file.

To query instances of assignment of records to parts in Teamcenter®, see section *8.*

3.1 Defining different data views

MI:Explore is highly-configurable, allowing different data and feature sets to be presented to different user groups in your company. This is done by defining alternate application front-ends, known as “data views”, in the MI:Explore configuration files.

Data views are identified by a unique key:

```
{
    "key": "DefaultSearch",
    "displayName" : "Explore Aerospace Materials",
    "default": true,
    "databaseKey": "MI_Training_11.0.0m",
    ...
},
{
    "key": "polymers",
    "displayName" : "Explore Polymers",
```

```
    "default": false,
    "databaseKey": " MI_Training_11.0.0m ",
    ...
}
```

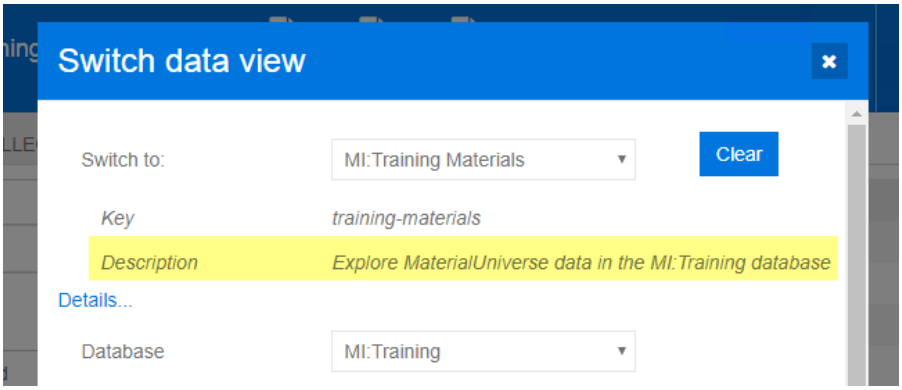
To open MI:Explore with a specific data view configuration, the key must be specified using a `search-config` query string in the URL, for example:

http://<mi_server>/explore/?search-config=polymers

If the search configuration is incorrectly specified in the URL (for example, misspelled), or if the specified search configuration is not found in any configuration files, MI:Explore will open with the default configuration, identified with `"default": true` ; see the first configuration extract in the example above.

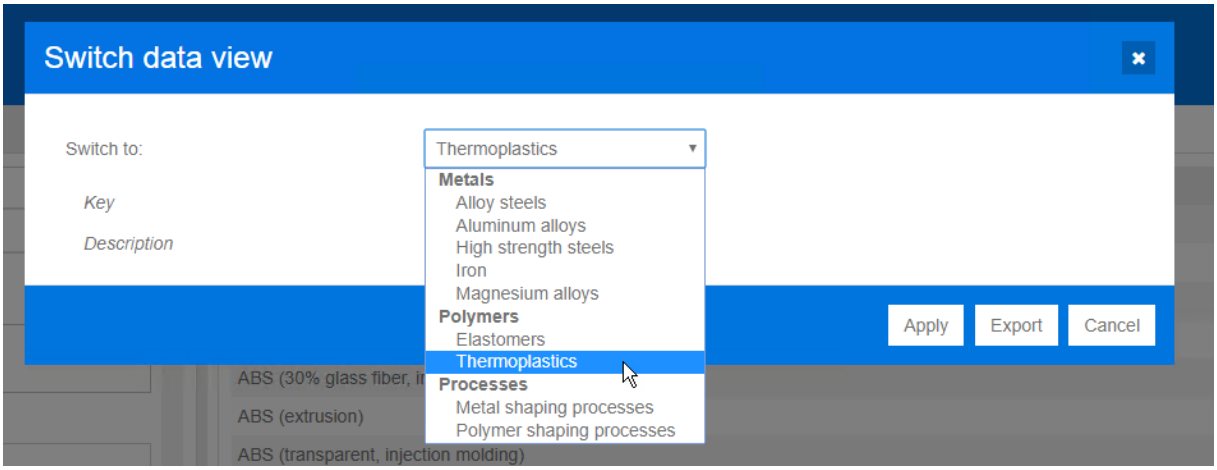
Data views may optionally be given a description.. This can be used to record useful information about the view, or to help users identify each data view in the Switch data view dialog.

```
"description": "Explore MaterialUniverse data in the MI:Training
database"
```



Data views may be designated a group, to further assist the user in selecting a data view from the Switch data view dialog.

```
"groups": ["Polymers", "On Demand"]
```



Each data view may appear in one group, many groups, or no groups. You can limit the data views that a user sees to only those views within a certain group; see section 4.

3.2 Application title

The name that appears in the application header is specified using the displayName setting in the configuration files. For example:

```
"key": "MI_MaterialUniverse_5.40.0m_Dec2016",
"displayName": "Explore Material Universe",
```

```
"key": "testdb",
"displayName": "Explore Test Records",
```

3.3 Database, table, and subset to search

The database, table, and subset from which MI:Explore will retrieve records are specified using databaseKey, table, and subset. For example:

```
"databaseKey": "MI_Training_11.0.0m",
"table": "MaterialUniverse",
"subset": "Aerospace materials",
```

The database key is optional; if it is not specified, the value defaults to the key of the database that the configuration file is stored with.

Specifying a subset allows the set of records presented in MI:Explore to be narrowed down to only records that are relevant to users using a particular MI:Explore data view. When a subset is specified, search and filter operations in MI:Explore will be limited to only the records in this subset; if no

subset is specified, search and filter operations will be applied to all records in the table. For example, the previous configuration defines the data of interest as metals and alloys records in the *Aerospace materials* subset in the *MaterialUniverse* table of the MI:Training database.

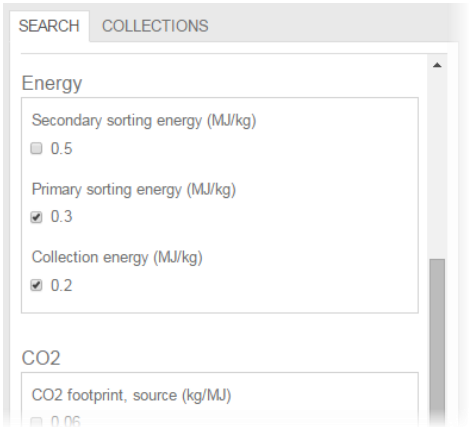
Any new records created in MI:Explore will be placed in the specified subset; see section *3.10, Data add and edit functionality*.

3.4 Search panel

When performing searches and filtering search results in MI:Explore, users can specify in the Search panel the values of interest for Range, Point, Integer, Short text, Long text, Discrete, Picture, Logical, and Date attributes. For Functional attributes (point and range) and File attributes, the search/filter criteria may specify whether or not a record includes the attribute (*exists*), but not its value.

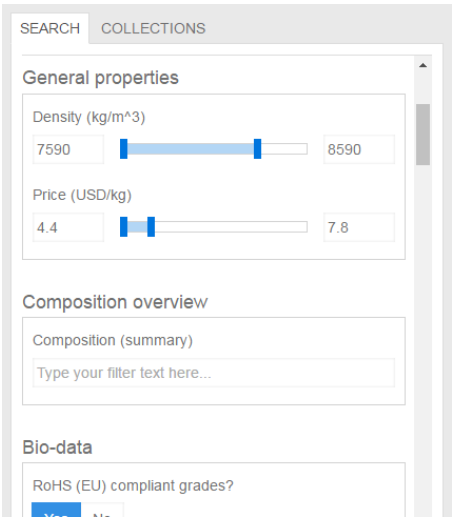
The attributes available in the Search panel in MI:Explore, and the headings under which they are grouped, are determined by the layout specified by searchLayout in the configuration files. An additional setting, logSliders, allows you to configure the use of logarithmic scales on slider controls in the Search panel.

3.4.1 Available attributes for searching




```
"searchLayout": "End of life",
```

In this example, a layout called *End of life* defines the headings *Energy* and *CO2* and the attributes that appear under them – *Secondary sorting energy, CO2 footprint,* and so on.




```
"searchLayout": "Materials search",
```

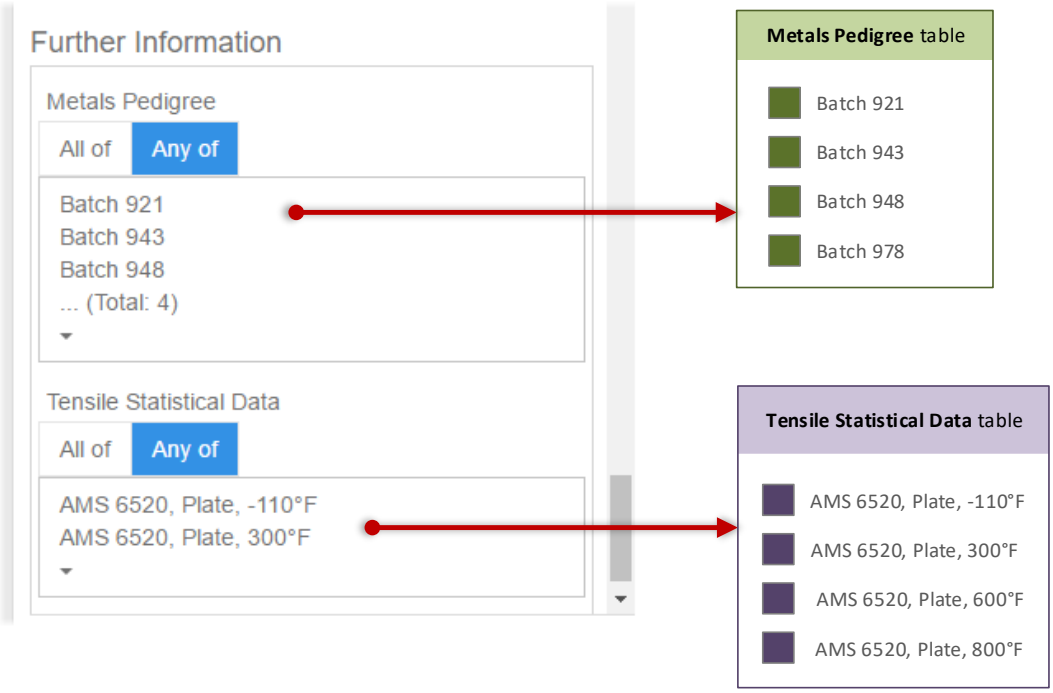
Here, a layout called *Materials search* defines the headings including *General properties* and *Composition overview*, and the attributes that appear beneath them – *Density, Price,* and so on.

3.4.2 Link groups

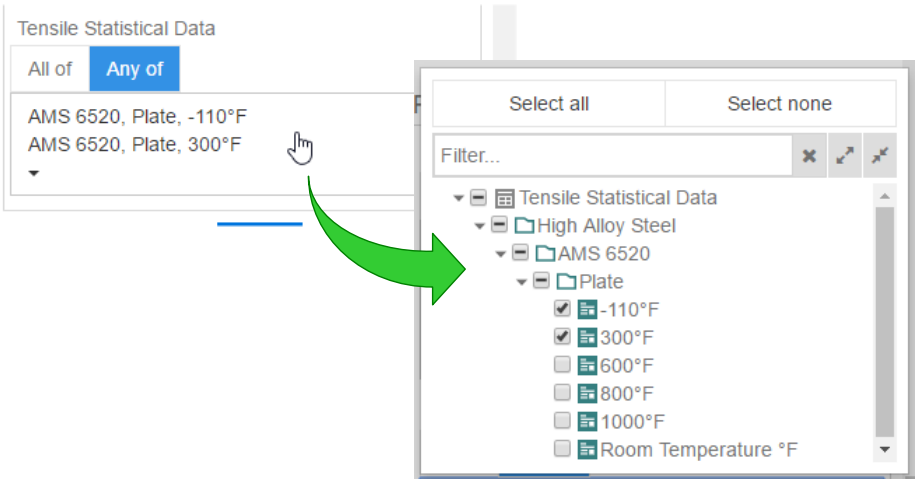
Searching across more than one database, table, and subset is not supported in MI:Explore. To filter search results, you can use link groups, allowing users to apply compatibility criteria to exclude results that are not compatible with records in other tables, for example, selected material specifications, processes, producers, coatings, or end of life options.

This is done using *Record link groups*, which capture relationships between related records in different tables within a GRANTA MI database.

For example, in the MI:Training database, records in the *Tensile Test Data* table include links to related records in the *Metals Pedigree* and *Tensile Statistical Data* tables:



To show only tensile test data records with certain batch and/or tensile statistical properties in MI:Explore, users can browse the linked table and select the relevant linked records:



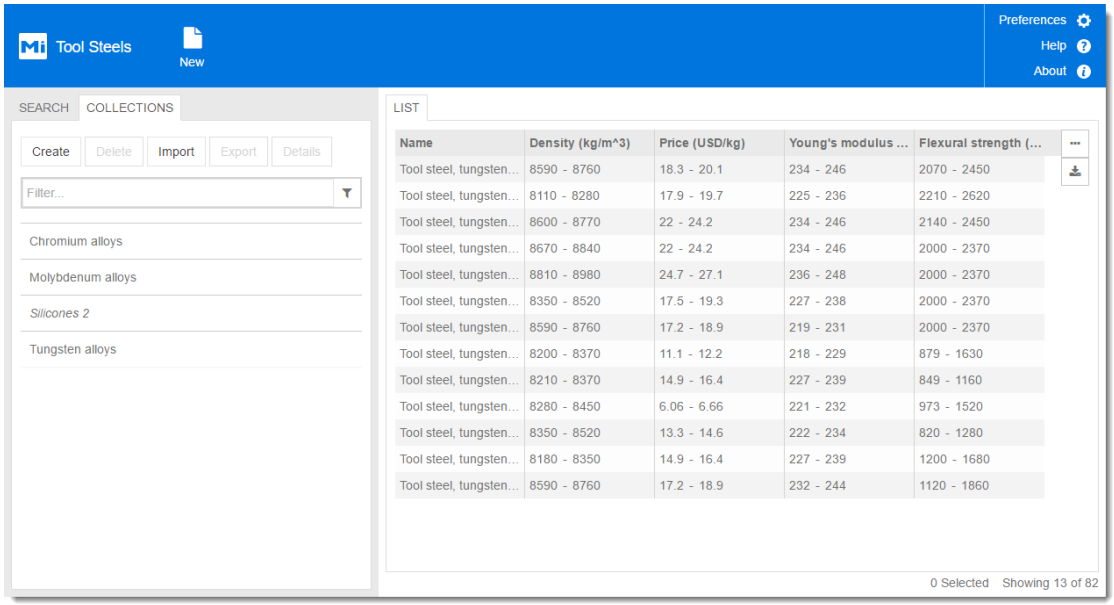
3.4.3 Logarithmic sliders

Most material properties extend over several orders of magnitude, so logarithmic scales can be used on slider controls in MI:Explore. This is configured using logSliders:

`"logSliders": true`	Slider controls will use a logarithmic scale when the data spread is larger than one order of magnitude
`"logSliders": false`	Slider controls will always use a linear scale

3.5 Collections

Collections work like “favorites”, and can be used to organize and access records that users work with frequently, or to group records that are used in a particular project, or that have a particular set of properties.



Collections can be easily created, deleted, and shared with other MI:Explore users. To add records to a collection, you select them in one of the tabs, then right-click and select **Add to collection**.

The Collections feature may be enabled and disabled in MI:Explore using the `collectionsDisabled` setting in the configuration file. When Collections are enabled, a COLLECTIONS tab appears beside the SEARCH tab, as shown in the screenshot above; when disabled, the Collections tab does not appear at all.

Note

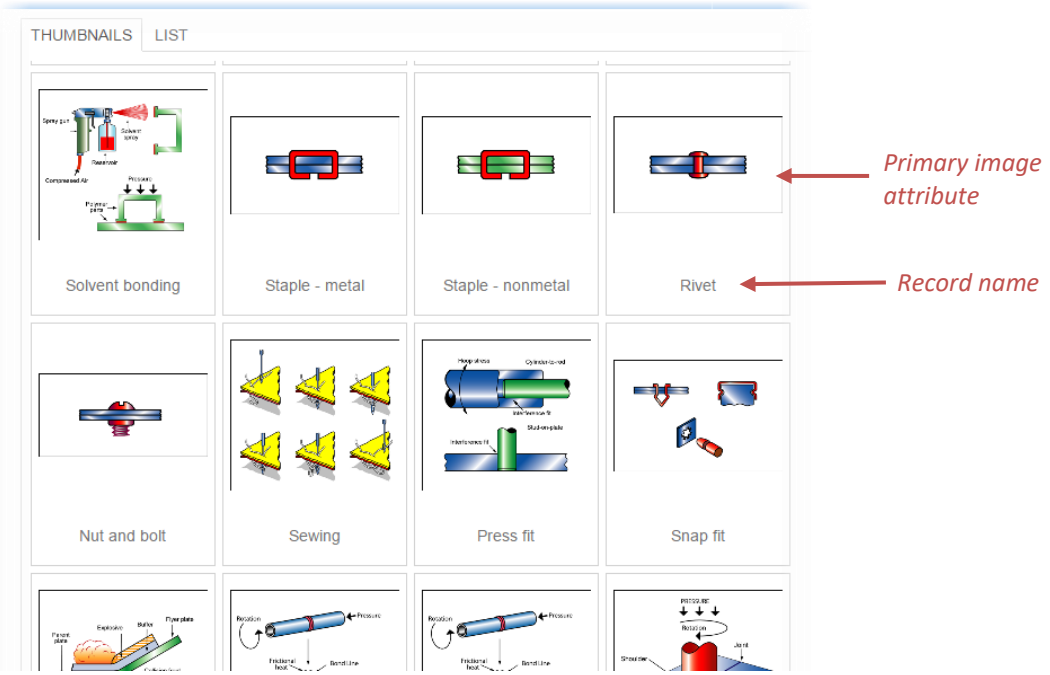
MI:Explore searches on a single table at a time (specified using the `table` key in the configuration files). However, collections may contain records from more than one table, and may also contain records that a user does not have permission to access, for example, when a collection has been imported. In these cases:

- Records in other tables (that is, not in the table that MI:Explore is currently searching) will not be visible when the collection is viewed.

- Records that a user does not have sufficient permission to access will not not be visible when the collection is viewed.
- Collections that contain only records that are not visible are shown in italics on the COLLECTIONS tab (see the *Silicones 2* collection in the example above)

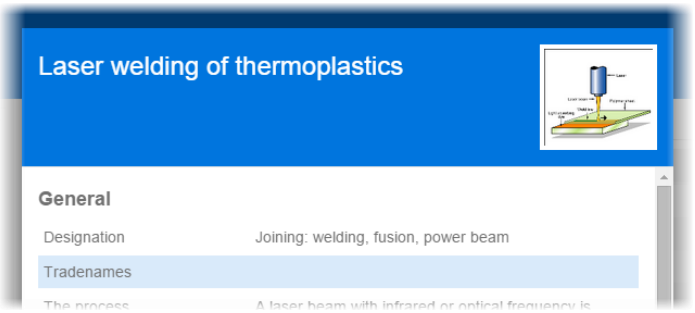
3.6 Thumbnails

The THUMBNAILS tab in MI:Explore is an optional tab that can display search results using an image attribute. Images are displayed as a thumbnail gallery:



The display of images on the THUMBNAILS tab in MI:Explore is configured using the `primaryImageAttributeName` setting, to specify the name of the primary image attribute.

The primary image is used on the thumbnails view of the search results (see image above). The primary image is also visible on the header of the record datasheet, for example:



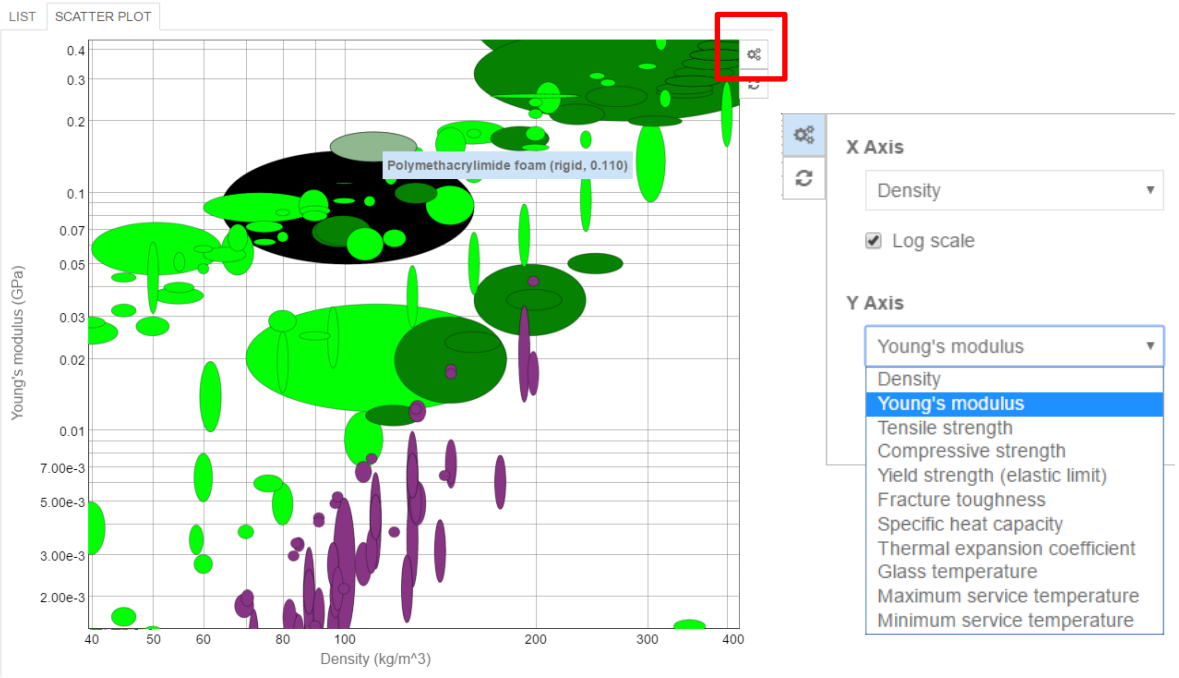
Example:

```
"primaryImageAttributeName": "Process schematic",
```

If no primary image attribute is specified in any records, then the THUMBNAILS tab will not appear in the application.

3.7 Scatter plots

Scatter (XY) plots are displayed on the SCATTER PLOT tab in MI:Explore. Users can change the x- and y- axis attributes and scale (linear or log) by clicking the **Settings** button on this tab.



The attributes that are available for selecting as x- or y-axes are those attributes defined by the search layout. Attributes of type integer, date, range, and point are permitted (point attributes may not be multi-valued). Any other attribute types in the search layout will not be available for use as a plot axis.

A number of plot settings can be configured under `xyChart` in the configuration file:

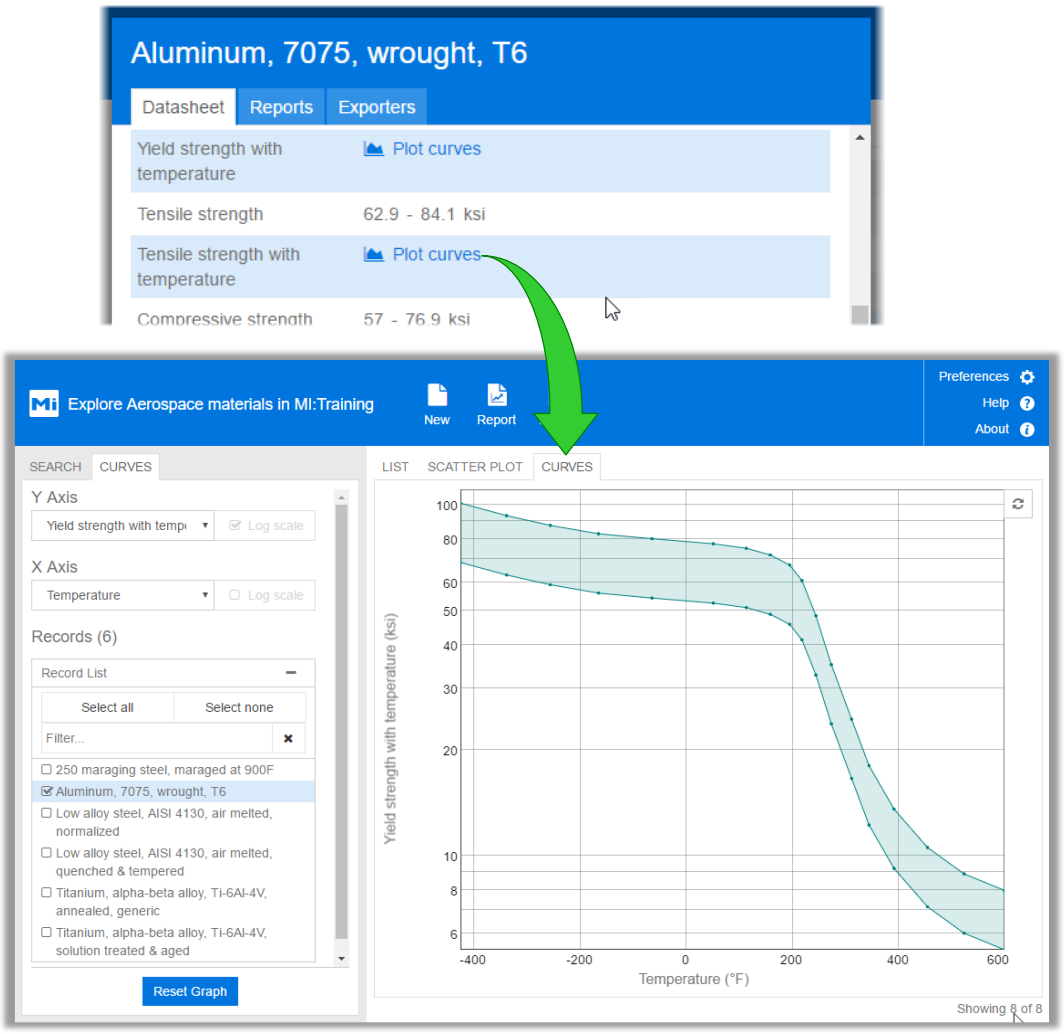
- The attributes used initially on the x- and y-axes on the SCATTER PLOT tab are specified with the `xAttribute` and `yAttribute` configuration fields.
- The default scale (linear or logarithmic) used initially for the x- and y-axes are specified with the `xAxisLogarithmic` and `yAxisLogarithmic`) configuration fields.
- The Settings button and panel can be hidden, allowing you to prevent MI:Explore users from changing any plot settings (by setting `preventAxisChange` to true).

Example:

```
"xyChart": {
  "xAttribute": "Young’s modulus",
  "yAttribute": "Tensile strength",
  "preventAxisChange": true,
  "xAxisLogarithmic": true,
  "yAxisLogarithmic": true
}
```

3.8 Curves

Where records include functional data, curves can be plotted on the CURVES tab, for example:



Users can plot curves from one or multiple records, choosing the y-axis attribute and the x-axis parameter, selecting the axis scales (linear or log, if applicable), and adjusting the parameter values to show more or less data on the plot.

The attributes available for plotting are any series functional attributes included in the search layout; gridded functional data cannot be plotted. Point data and range data can be plotted.

A number of default settings can be configured under curves in the configuration file:

- The attribute used initially on the y-axis on the **CURVES** tab is specified with the `yAttribute` configuration field.
- The parameter used initially on the x-axis on the **CURVES** tab is specified with the `xParameter` configuration field.
- The default scale (linear or logarithmic) used initially for the x- and y-axes are specified with the `xAxisLogarithmic` and `yAxisLogarithmic` configuration fields.

Example:

```
"curves": {
  "yAttribute": "Young's modulus",
  "xParameter": "Temperature",
  "xAxisLogarithmic": false,
  "yAxisLogarithmic": true
}
```

3.9 Datasheets

The content and functionality available in MI:Explore datasheets is configured in the configuration file:

- You can specify the layouts that define the headings and attributes that appear in datasheets; see section *3.9.1* below.
- You can specify whether or not attributes that have no value will be shown in datasheets; see section *3.9.2* below.

In addition, you can control the buttons that appear on datasheets:

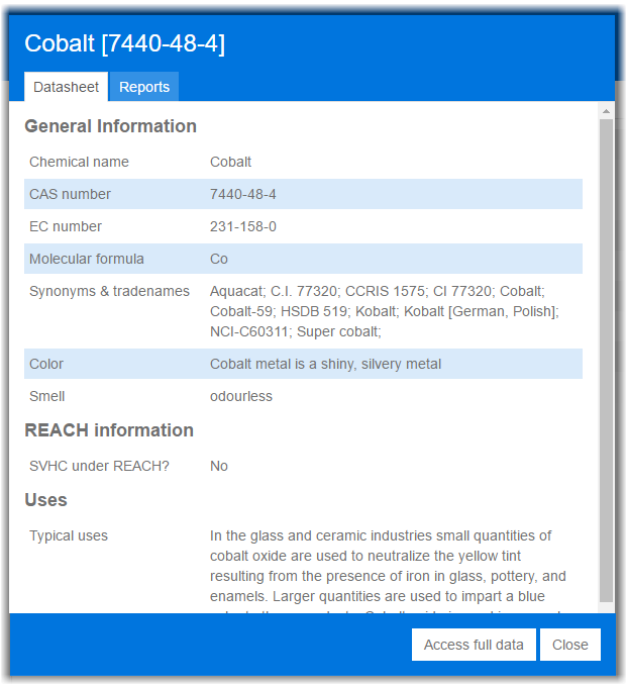
- **Edit** button; see section *3.10, Data add and edit* .
- **Access full data** button (configured in preferences.json); see section *6, MI:Viewer integration*.

3.9.1 Layouts – datasheet attributes and headings

Layouts are used to define the attributes and headings that appear in datasheets in both GRANTA MI and in MI:Explore:

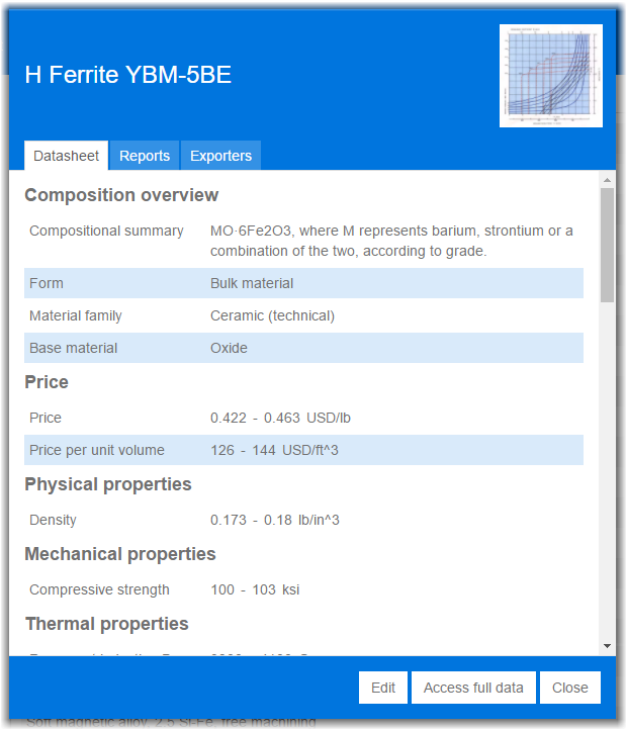
- `dataSheetLayout` specifies a layout that defines the headings and attributes shown when viewing datasheets in MI:Explore.
- `editableDatasheetLayout` specifies a layout that defines the headings and attributes shown when adding or editing data in MI:Explore; this may be a different set of attributes than shown when viewing the datasheet. The layout may also identify certain attributes as *read-only*, meaning they can be viewed but not modified, or *required*, meaning they must be completed in order to save the record.

Examples



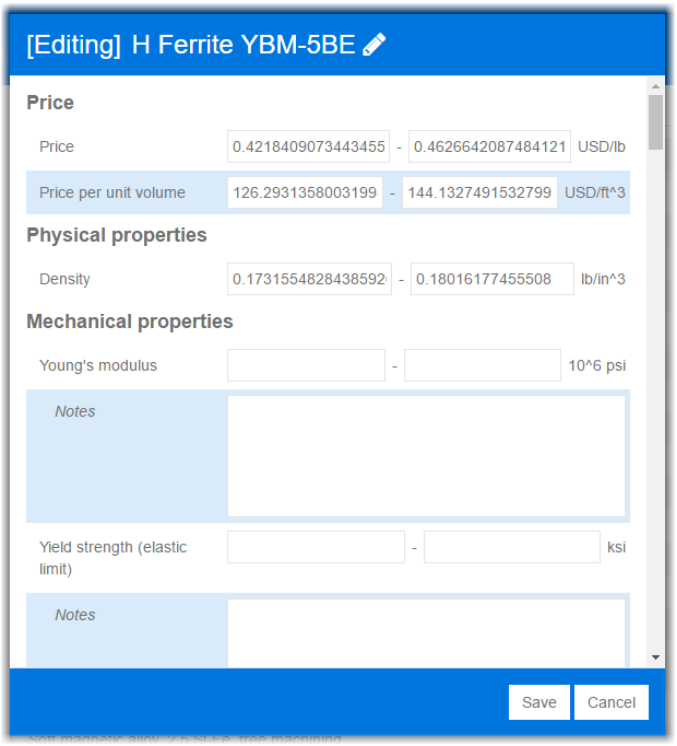
`"dataSheetLayout": "All substances"`

The layout named *All substances* defines the headings *General information*, *REACH information*, and *Uses*, and the attributes that appear beneath them.



`"dataSheetLayout": "All attributes"`

Here, a material record in the Material Universe database is viewed using the layout called *Magnetic materials*, which defines the headings *Composition overview*, *Price*, and *Physical properties,* as well as the set of attributes that appear beneath each heading – *Form*, *Base material*, *Density, Compressive strength,* and so on.



`"editableDataSheetLayout": "Magnetic materials (Edit)"`

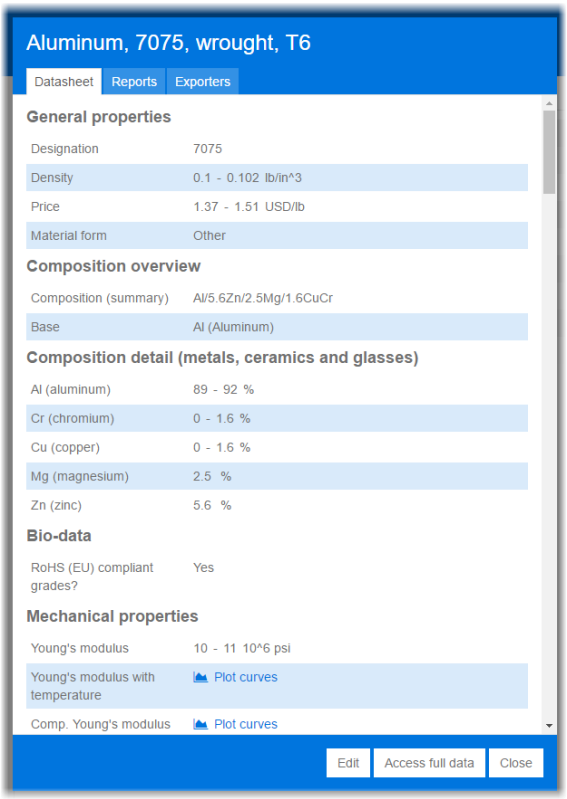
Here, the magnetic material record viewed earlier is now being edited: a different layout is used to define the properties that appear on the datasheet (compare this with the headings and attributes visible when viewing the datasheet, above).

3.9.2 Showing/hiding attributes with no value

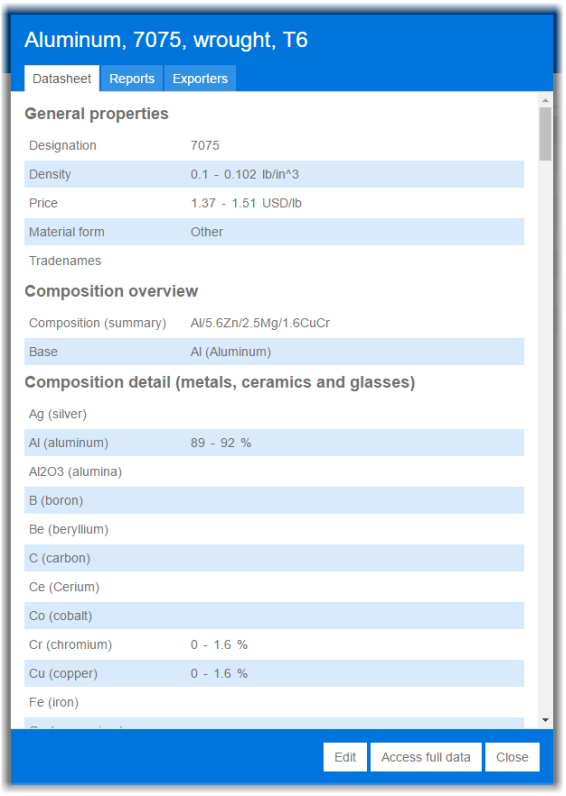
The `showBlanks` setting controls whether attributes with no value set are shown or hidden when datasheets are viewed in MI:Explore.

When `false`, attributes with no data value are hidden when viewing datasheets; when `true`, empty attributes are shown.

`"showBlanks": false,`



`"showBlanks": true,`



Note that the `showBlanks` setting only applies when *viewing* datasheets; when *editing* datasheets, all attributes in the layout are always shown.

3.10 Data add and edit functionality

3.10.1 Enabling data and edit capability

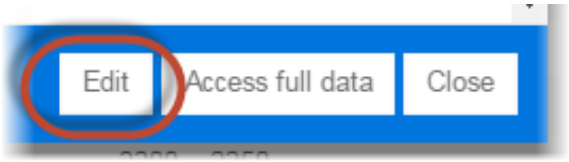
To enable MI:Explore users to add and edit data, the `editableDatasheetLayout` setting is used in the configuration file.

This setting *both* enables/disables creation of new records and editing of data, *and* specifies the layout used when entering or editing data.

When `editableDatasheetLayout` is specified:

- The **New** button appears on the toolbar; users with the necessary permissions can click this to add new records.

- An **Edit** button appears on datasheets, allowing users with the necessary permissions to edit data values.



The specified layout defines the headings and attributes shown when entering or editing data (see Section *3.9.1*) and may also identify certain attributes as ***read-only***, meaning they can be viewed but not modified, or ***required***, meaning they must be completed in order to save the record.

If `editableDatasheetLayout` is not specified, the **New** button will not appear on the toolbar and the **Edit** button will not appear in MI:Explore datasheets.

Note that, to allow users to create new records, a `subset` must also be specified in the configuration file.

Some addition configuration is required if users will need to edit tabular attributes; see section *3.10.2*.

Access controlled tables

Note that MI:Explore users who need to edit version-controlled records must be members of the GRANTA MI *Power User* system security role.

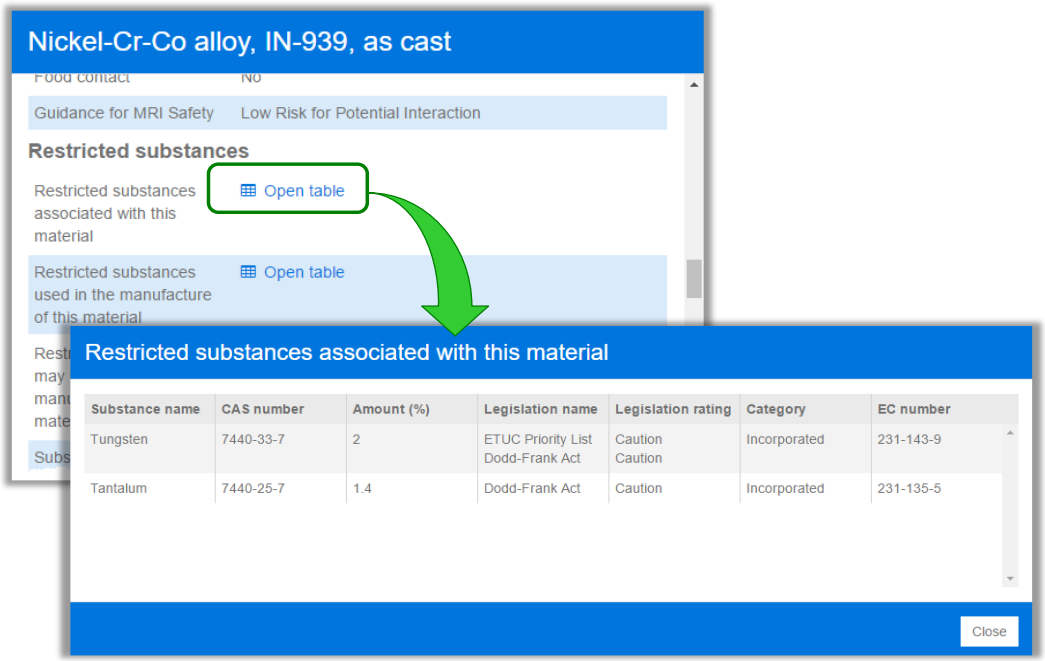
This is because MI:Explore automatically tries to release new record versions; if a user does not have sufficient privileges to be able to release records (i.e. they are not members of the *Power User* role, or higher), the record will be updated and the new record version created, but it cannot be released, and so the user will see an error.

3.10.2 Viewing and editing tabular data

Tabular attributes are used in GRANTA MI to store complex data, with data points (simple numeric, text, media, logical, or date data) organized into columns and rows; tabular attributes may include local data as well as links to data stored in other tables.

Viewing tabular data in MI:Explore

When a tabular attribute is included in the layout specified in dataSheetLayout, an **Open table** option will appear on the datasheet, enabling users to view the data. For example:

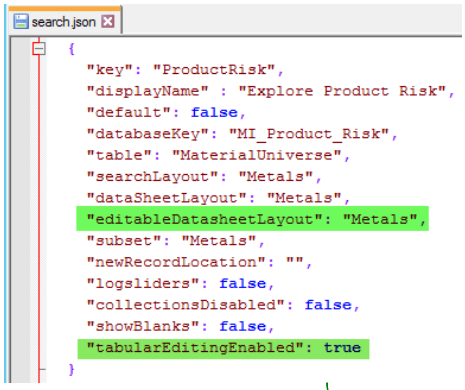


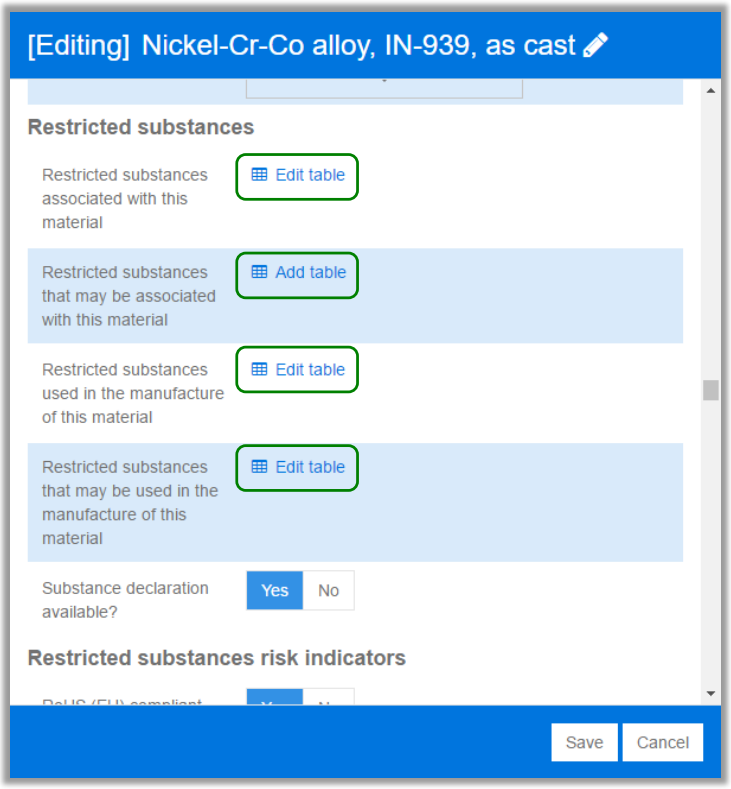
Adding and editing tabular data in MI:Explore

The ability to edit tabular data in MI:Explore is controlled with the `tabularEditingEnabled` setting in the configuration file.

`"tabularEditingEnabled": true`	Tabular data will be editable.
`"tabularEditingEnabled": false`	Tabular data will not be editable.

When tabular data editing is enabled, and the layout specified in editableDatasheetLayout includes a tabular attribute, **Edit table** or **Add table** options will appear on the editable datasheet for that attribute. For example:



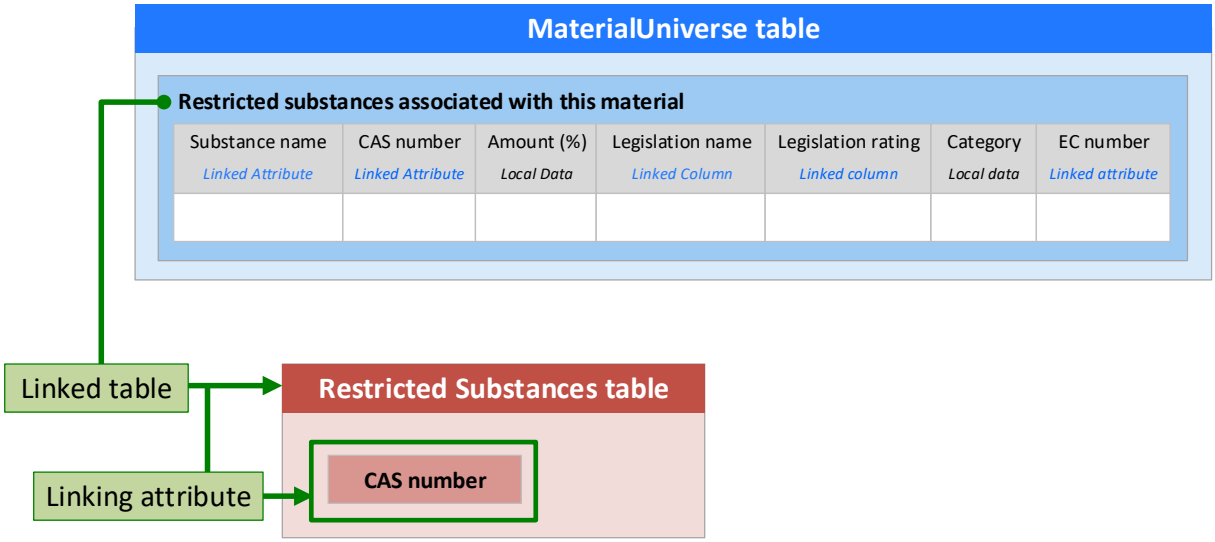


Tabular data add and edit options in MI:Explore editable datasheet

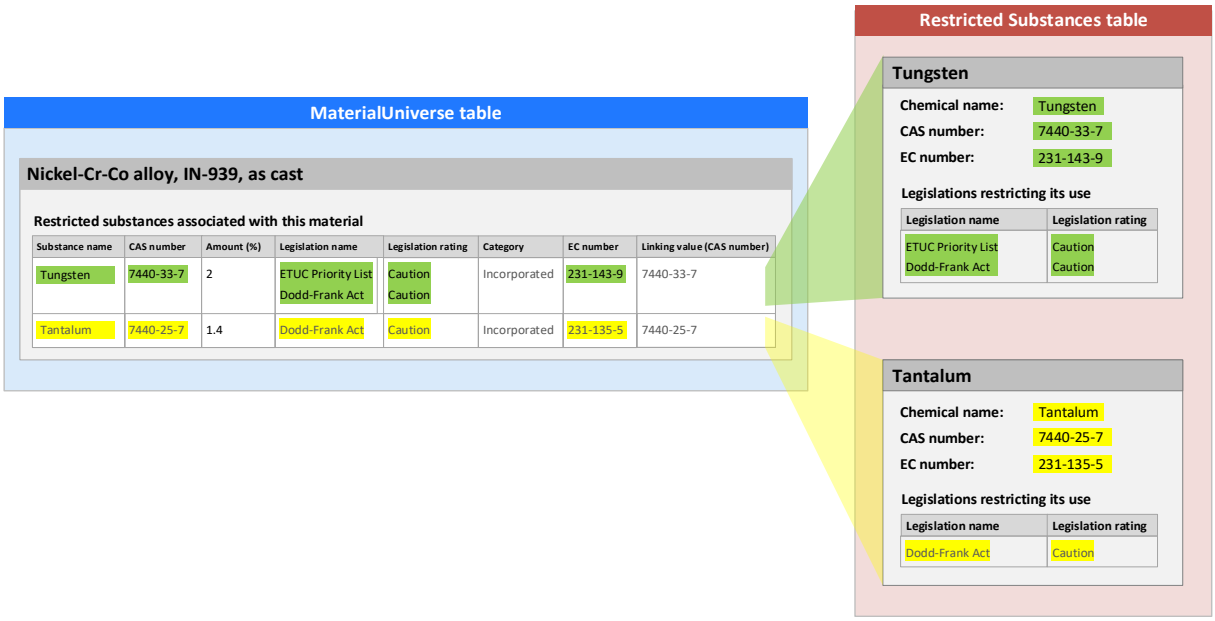
3.10.3 Editing tabular data linking values

Tabular attributes in GRANTA MI may include data from other tables, as well as local data. To include data from other records, each row of tabular data uses the value of a **linking attribute** to link to the relevant record in the **linked table**.

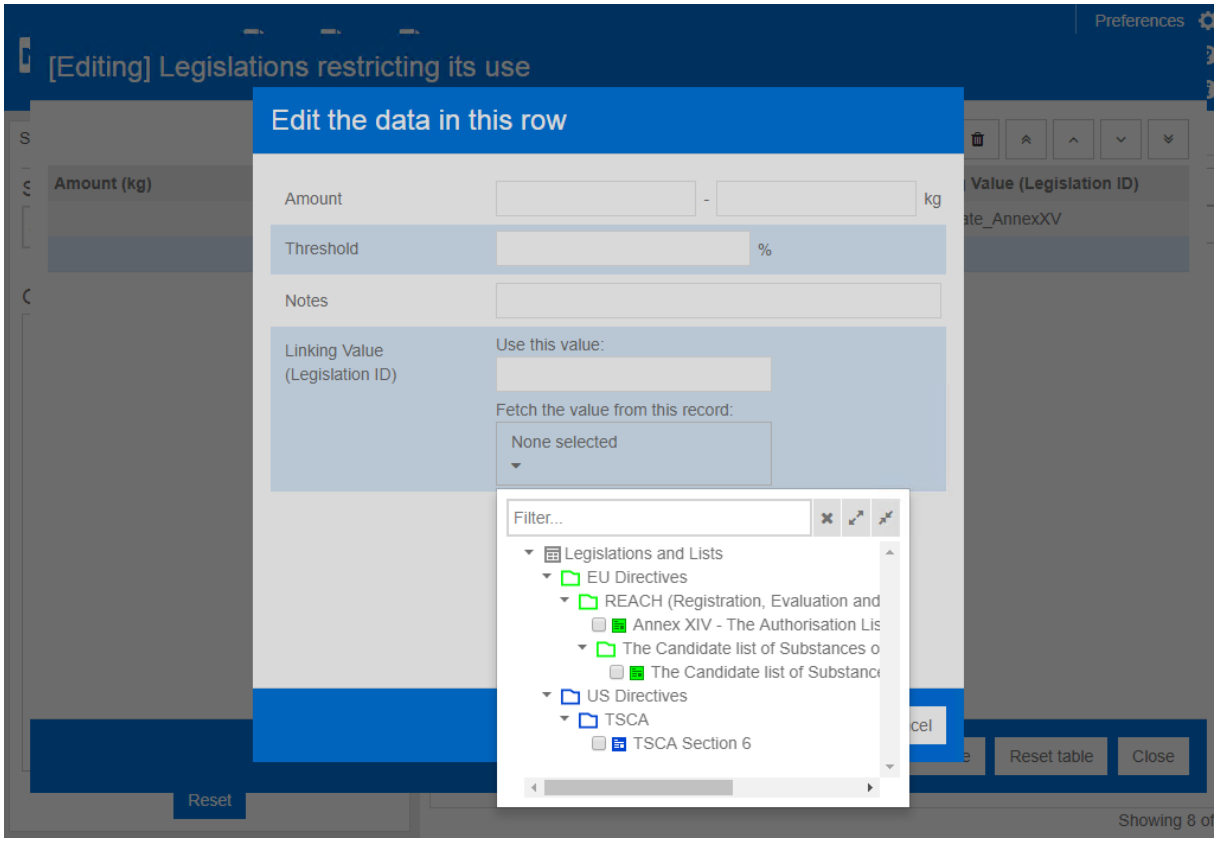
For example, in the *Product Risk* database *MaterialUniverse* table, the tabular attribute *Restricted substances associated with this material* is linked to the *Restricted Substances* table via the CAS number ***linking attribute*** value.



The linking value connects the tabular data row in the *MaterialUniverse* table to a record in the *Restricted Substances* table, allowing data values from the *Restricted Substances* record to be included in the tabular data row, for example:

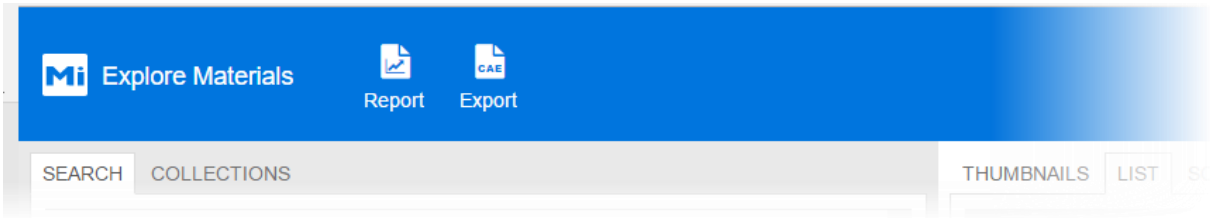


In MI:Explore, users can edit linking values in tabular data attributes by typing in the value, or by selecting a record from which the linking value can be retrieved. For example:



3.11 Reports

Where reports from the MI:Reports package are installed in your GRANTA MI environment, any reports that analyze a list of records can be run from within MI:Explore via a **Report** option on the toolbar.

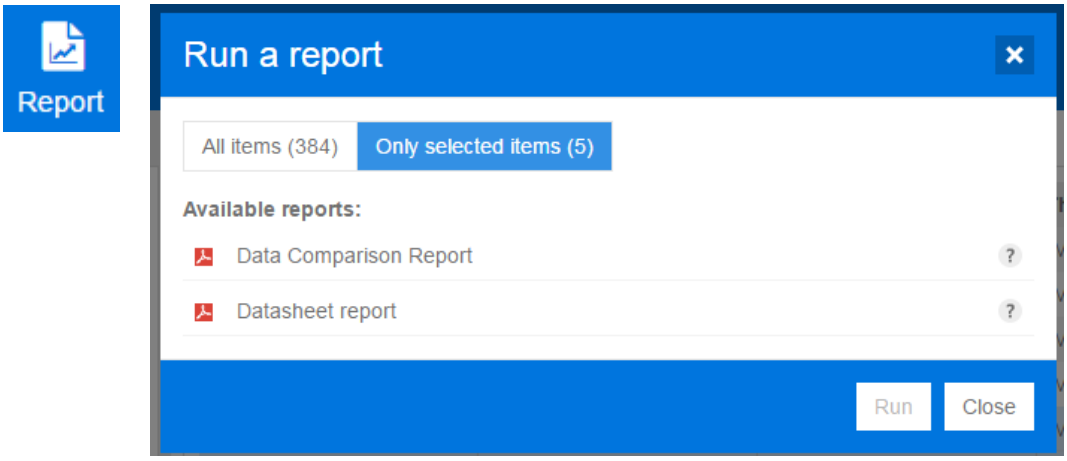


To enable or disable the Reports functionality in MI:Explore, edit the reportsDisabled option in the configuration file:

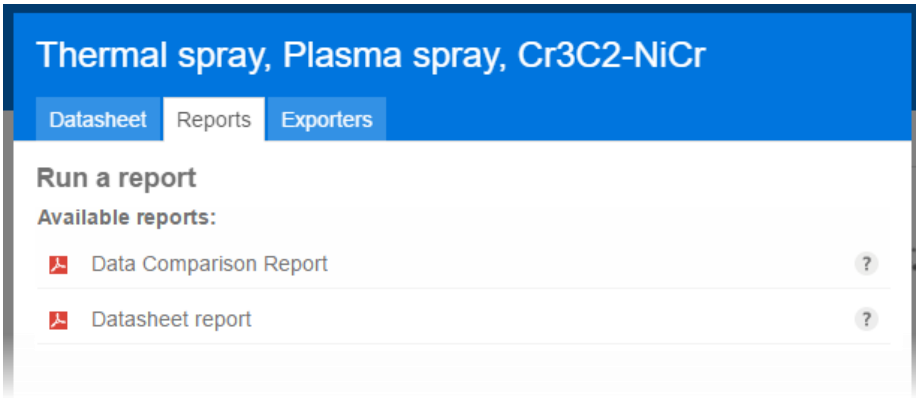
`"reportsDisabled": true`	The **Report** option will not appear in the toolbar.
`"reportsDisabled": false`	The **Report** option will appear in the toolbar.

When reports are enabled:

- A **Report** button appears on the toolbar. Users can click this to open a dialog where they can select and run a report on all records, or on only the selected records:



- A **Reports** tab appears on record datasheets showing the reports that can be run on that record:



3.12 Exporters

Materials data can be exported from GRANTA MI in a range of different formats for use in FE packages such as Abaqus, ANSYS, or NastranNX. Where FEA exporters are available in your GRANTA MI database, these can be run from within MI:Explore via the **Export** option on the toolbar.

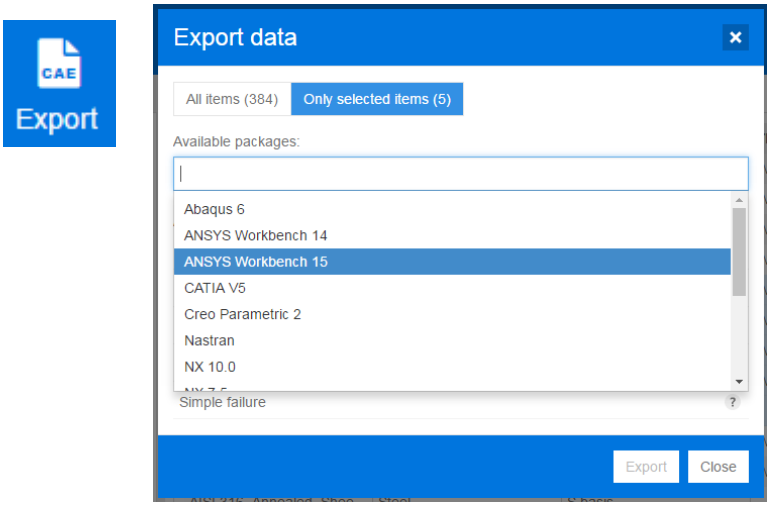
3.12.1 Enabling exporters

To enable or disable the exporters functionality in MI:Explore, edit the exportersDisabled setting in the configuration file:

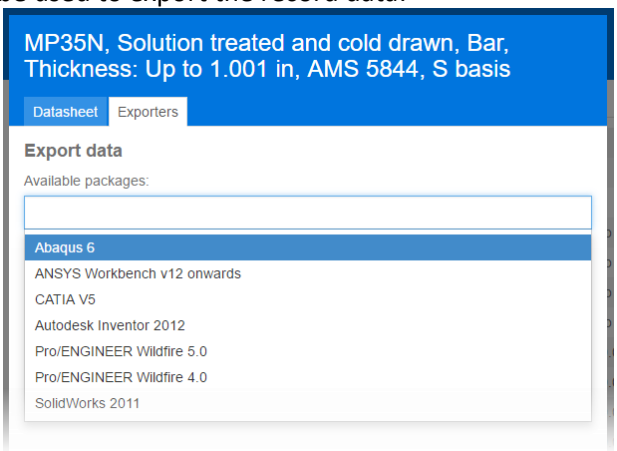
`"exportersDisabled": true`	The **Export** option will not appear in the toolbar.
`"exportersDisabled": false`	The **Export** option will appear in the toolbar.

When the exporters functionality is enabled:

- An **Export** button appears on the toolbar. Users can click this to open a dialog where they can select one of the available exporters, and export data from the selected records:



- An **Exporters** tab appears on record datasheets showing the packages and exporters that can be used to export the record data:



3.12.2 Export units

MI:Explore will select a unit system to use when exporting data, either:

- the unit system specified in the exporter configuration file, or
- the current unit system specified in the user's MI:Explore preferences.

To specify the required exporter unit system option, edit the exportersUseCurrentUnitSystem setting in the configuration file:

`"exportersUseCurrentUnitSystem": true`	The exporter will use the current unit system specified in the MI:Explore preferences (but note that this unit system may not necessarily be supported by the exporter).
`"exportersUseCurrentUnitSystem": false`	The exporter will use the unit system specified in its own .exp file definition.

3.12.3 Hiding application-specific exporters

FEA exporters in a GRANTA MI database may be configured for use with specific applications via a setting in the exporter configuration (.exp) file, for example:

This `<Applicability>` tag in the exporter config file is used to ensure that only relevant exporters are shown when users are exporting data in MI:Materials Gateway and MI:Viewer: exporters configured for use with Gateway do not appear in MI:Viewer and so cannot be selected, and vice versa.

Exporters can also be filtered in MI:Explore, ensuring that users only ever see relevant exporters. To do this, the value of the `exporterApplicability` setting in the configuration file must match the value of the <Applicability> tag in the exporter configuration file. For example, with this configuration:

```
"exporterApplicability": "MIViewer"
```

only exporters that have an `<Applicability>` tag value of *MIViewer* (as in the Abaqus example on the right above) will appear in the list of available exporters in MI:Explore.

Note:

- If the exporter .exp config file does not include an `<Applicability>` tag, the exporter will always be shown in MI:Explore.

- If the `exporterApplicability` setting is missing from the configuration file, all exporters in the database are shown.

3.13 Options for placement of new records

When an editable datasheet is specified in the configuration file, users with sufficient permissions can create new records. Records created in MI:Explore can be placed into specific folders in the GRANTA MI database tree structure. For example:

```
"newRecordLocation": "Metals and alloys/New/"
```

This is configured by specifying the destination folder path using `newRecordLocation`.

- The folder will be created if it does not already exist.
- If not specified, new records will be placed in the top-level (root) folder.
- To create records, users must have write permission to the parent folder.

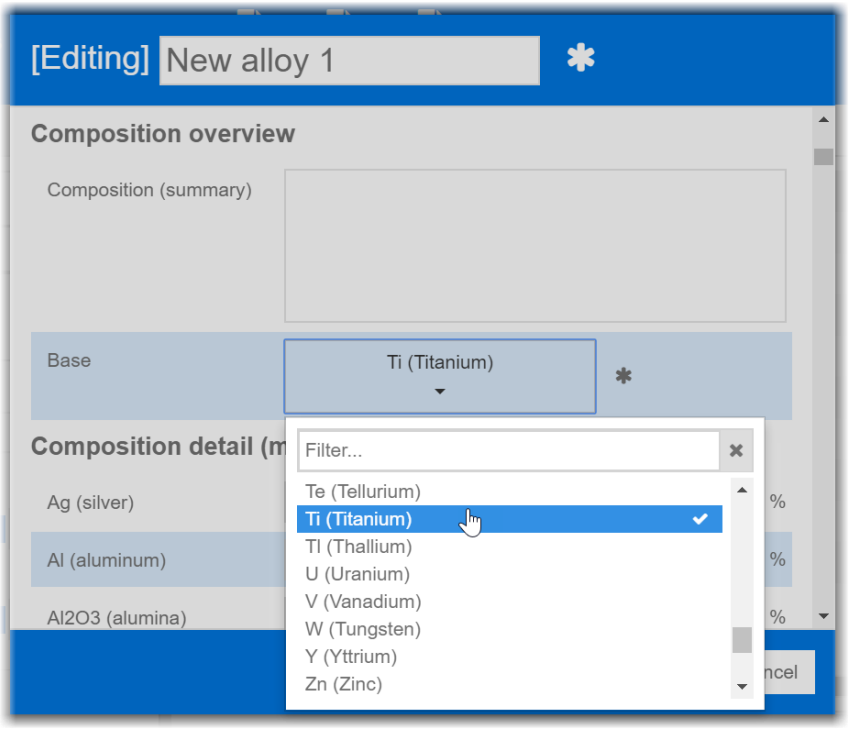
As well as literal strings, the path specified in `newRecordLocation` can include attribute names; when a new record is added to the database, the value of the attribute is then used in the record path. For example, the 2nd line below will place new records in the folder determined by the value of the Discrete attribute *Base*, as specified by the user.

Example:

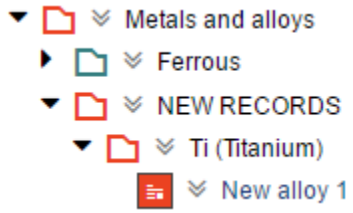
The new record location configuration:

```
"newRecordLocation": "Metals and alloys/NEW RECORDS/{Base}",
```

Value for the *Base* attribute (a required attribute) is specified when creating the new record:



The new record is placed in the specified folder structure:



For more detailed information and additional examples, see Section *9.19*.

3.14 Options for autonaming of new records

Automatic naming and numbering conventions can be configured for records created in MI:Explore using a number of configuration settings the configuration file.

A system of transforms can be used to construct the new record name, for example, to:

- Take two attribute values and concatenate them into a third attribute value;
- Increment the value of an integer attribute;
- Pad the integer attribute to a fixed length string;
- Concatenate two attributes and write them into the record name.

A number of useful transforms are supplied as part of the Granta Web Platform; the system is extensible, however, and you can write your own.

Table 1. Available transforms

Record Transform	Function	Standard Names	Mapped to attribute of type	Required?
Counter	Increments an integer attribute with every new record created	Counter	Integer	Y
Id	Concatenates two attribute values into a third attribute	Id_Prefix	Discrete	N
		Id_Suffix	Integer, Short text	N
		Id	Short text	Y
RecordNameConcatenator	Concatenates two attribute values into the record name	Name_Prefix	Integer, Short text, Discrete	Y
		Name_Suffix	Integer, Short text, Discrete	N

If a required attribute is missing, the transform will fail. If a non-required attribute is missing, the transform will succeed but the concatenation will be incomplete.

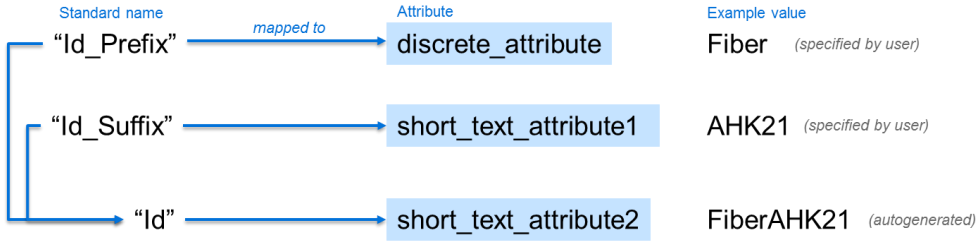
3.14.1 The **Id** transform

This transform concatenates two attribute values into a third attribute.

Record Transform	Standard Names	Attribute type	Required?
`Id`	Id_Prefix	Discrete	N
	Id_Suffix	Integer or Short text	N
	Id	Short text	**Y**

The **Id_Prefix** and **Id_Suffix** standard named attributes are both optional. If neither is supplied, the result will be that the **Id** standard named attribute will be empty.

Example



To set this up:

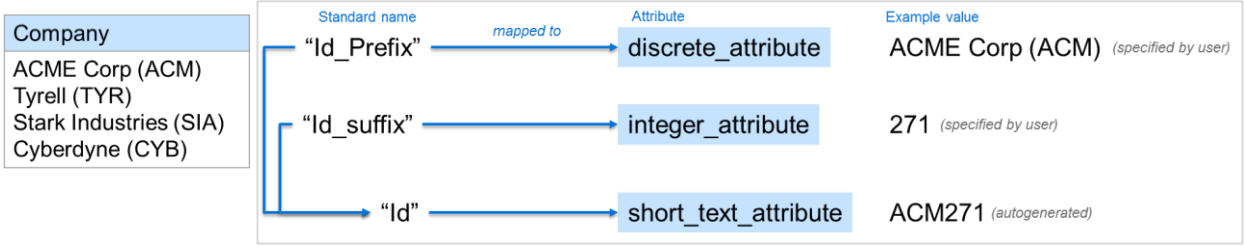
1. Set up three standard named attributes: **Id**, **Id_Prefix,** and **Id_Suffix**.
2. Add the **Id** transform to the list of transforms in the configuration file:

   `"newRecordTransformation": ["Id"]`
3. When a new record is created, the **Id** attribute in the record is populated automatically from the values specified in the other two attributes.

Text in parentheses

If the value of the **Id_Prefix** standard named attribute is a string containing a set of parentheses ( ), then only the substring within the brackets is concatenated with the **Id_Suffix** standard named attribute to produce the value written into the **Id** standard named attribute.

For example, for an **Id_Prefix** standard named attribute “Company”, with a value of “ACME Corp (ACM)”, only the string within parentheses, “ACM”, is included in the **Id**.

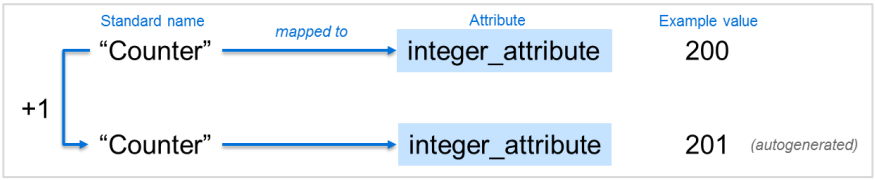


3.14.2 The **Counter** transform

This transform increments an integer attribute with every new record created.

Transform	Standard Names	Mapped to attribute of type	Required?
`Counter`	Counter	Integer	Y

Example:



To set this up:

1. Set up one standard named attribute, **Counter**, of type integer.
2. Add `Counter` to the list of transforms in the configuration file:

   `"newRecordTransformation": ["Counter"]`

Each time a new record is created, the value of the **Counter** attribute is incremented.

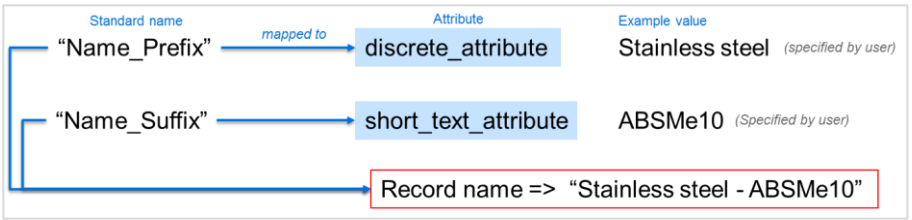
3.14.3 The RecordNameConcatenator transform

This transform concatenates two attribute values into the record name.

Record Transform	Standard Names	Attribute type	Required?
`RecordNameConcatenator`	Name_Prefix	Integer, Short text, Discrete	Y
	Name_Suffix	Integer, Short text, Discrete	N

The **Name_Suffix** standard named attribute is optional. If it is not supplied, the resulting record name will include only the **Name_Prefix** value.

Example:



To set this up:

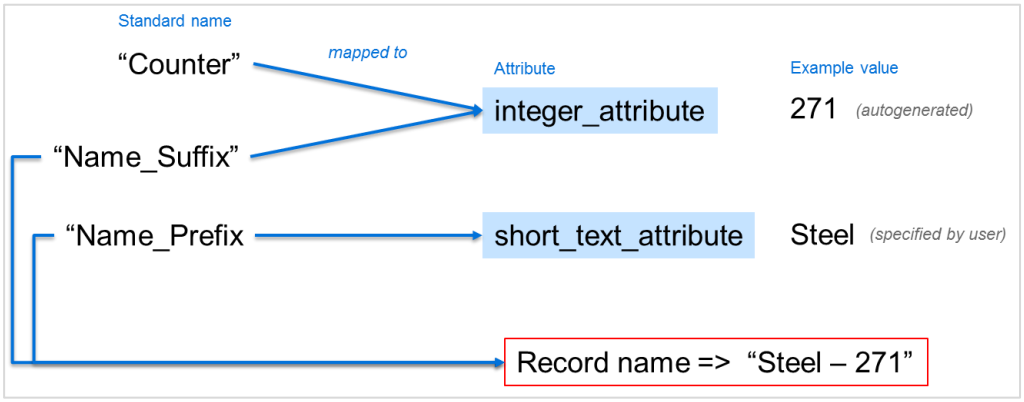
1. Set up two standard named attributes (Integer, Short text, or Discrete) to use for the first and second part of the record name; for example, **Name_Prefix** and **Name_Suffix**.
2. Add `RecordNameConcatenator` to the list of transforms in the configuration file:

   `"newRecordTransformation": ["RecordNameConcatenator"]`
3. When a new record is created, it is automatically named by concatenating the two attribute values, separated by a hyphen (-).

3.14.4 Combining two transforms

Multiple transforms may be specified in the configuration file; they are applied in order. For example, here, the `Counter` and `RecordNameConcatenator` transforms are combined:

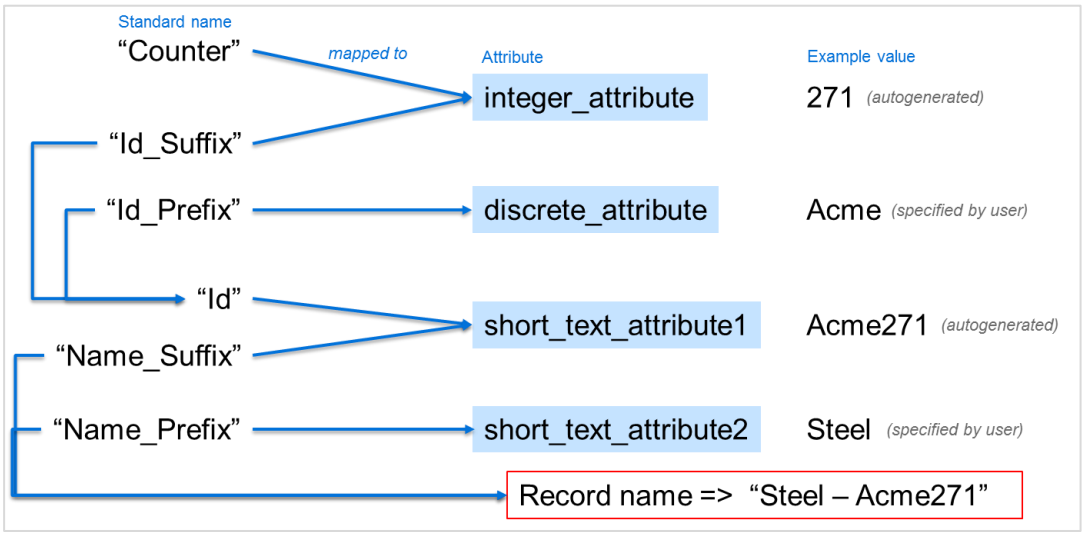
```
"newRecordTransformation": ["Counter", "RecordNameConcatenator"]
```



3.14.5 Combining three transforms

All three Transforms may be specified in the configuration file. For example, here, the `Counter`, `ID`, and `RecordNameConcatenator` transforms are used:

```
"newRecordTransformation": ["Counter", "Id",
"RecordNameConcatenator"]
```

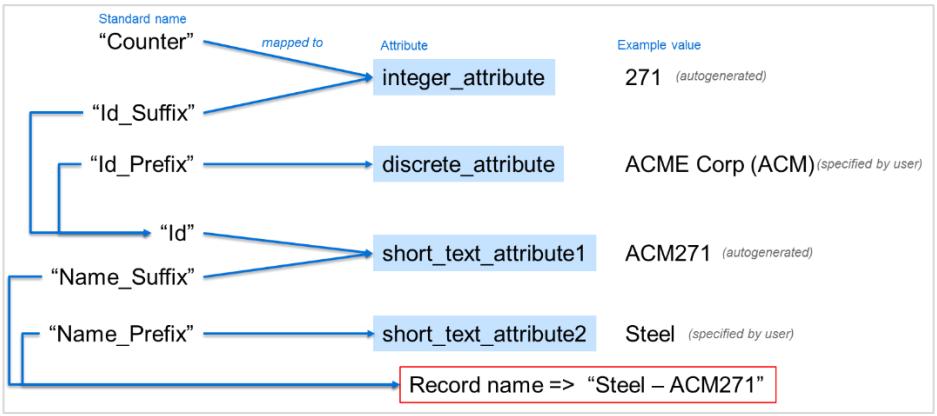


3.14.6 Additional transform examples

Text in parentheses example 2

If the value of the **Id_Prefix** includes text within parentheses ( ), that text will be used in the record name and not the whole string. For example:

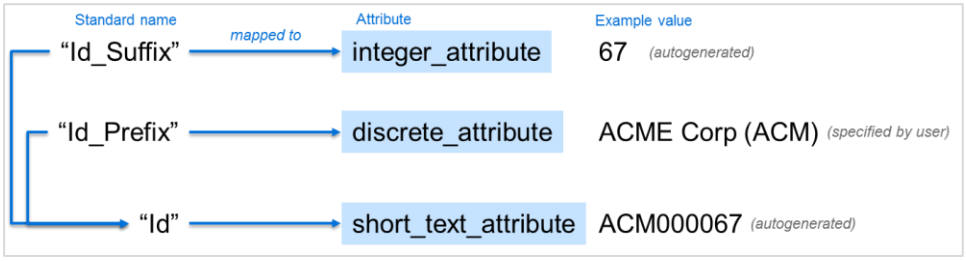
Company
ACME Corp (ACM)
Tyrell (TYR)
Stark Industries (SIA)
Cyberdyne (CYB)



Padding characters

The **Id** transform may include padding on integer attributes. For example, in this configuration, the numeric counter part of the record name is padded with zeroes to a length of 6 characters:

```
"newRecordPaddingOptions": {
      "padLength": 6,
      "padString": "0"
}
```



3.14.7 More information

See Section *9.20* for details of how to specify these configuration options in the configuration file.

The **Counter**, **Id**, and **RecordNameConcatenator** transforms are available in the Granta Web Platform. Additional custom transforms may be developed using the `IRecordTransformer` interface; see the Web Platform SDK documentation for details.

3.15 Load data on demand

Data in MI:Explore can be loaded on demand, or on startup:

- **On demand**: data is loaded when required. This reduces the startup time of the application when using large datasets.
- **On startup**: all data is loaded when the application is started, or when switching to a different search configuration URL. This is recommended for small datasets.

To enable or disable the data on demand mode in MI:Explore, edit the `loadDataOnDemand` setting in the configuration file:

`"loadDataOnDemand": true`	Data will load into the application when required
`"loadDataOnDemand": false`	All data will load on startup

Some UI differences occur when data on demand is enabled. When the loadDataOnDemand functionality is enabled:

- A **Search** button appears on the search panel. Users set their search criteria, and then click Search to load the results from the server.

When the loadDataOnDemand functionality is disabled:

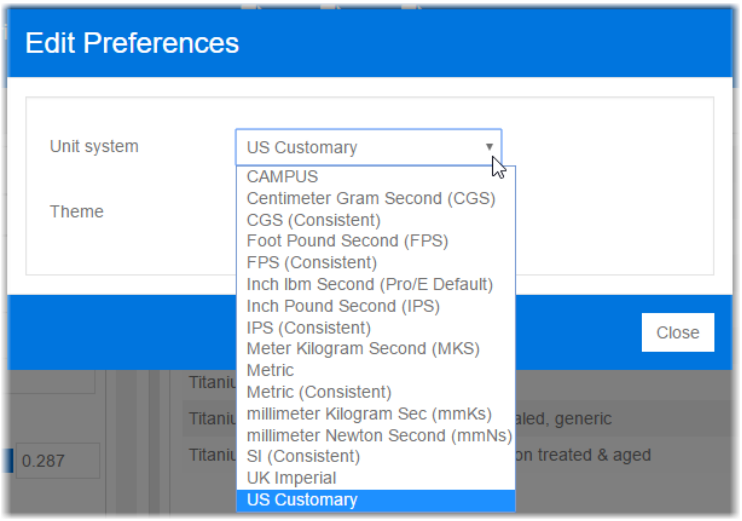
- All data is pre-loaded from the server at the startup of the application. As users change their search criteria, the results update instantaneously.

Similarly, the Curves panel is less responsive in data on demand mode, as data has to be loaded from the server before updating the curve plot.

3.16 Unit system options

Unit system options for MI:Explore are specified in preferences.json:

- The default unit system used in the MI:Explore application can be specified.
- The **Unit system** option in the **Edit Preferences** dialog can be shown, allowing users to select a different unit system, or hidden, preventing users from using a different unit system.



In this extract from a preferences.json file, the **Unit system** user interface preferences option is turned on, and *US Customary* is set as the default unit system:

```
        "unitSystem": {
             "userVisible": true,
             "defaultValue": "US Customary"
         },
```

Where the data is stored in a different unit system, MI:Explore accesses services in GRANTA MI to perform unit conversions, using information in the current database.

When exporting data, the MI:Explore unit system can be used, if the exporter permits it, or the exporter's own unit system can be used; this is configured using the exportersUseCurrentUnitSystem setting, see section *3.12.2, Export units*.

3.17 Number formatting settings

Number separator settings are specified in preferences.json:

- Recommended group separators are space, comma, period, or none.
- Recommended decimal separators are comma or period.

For example, 1,234,567.89 has comma digit group separators and a period decimal separator.

To allow users to edit their number formatting settings, use the `numberFormatting` setting in preferences.json to specify their options:

```
"preferences":  {
    ...
    "numberFormatting":  {
        "userVisible":  true,
        "defaultValue":  {
            "groupSeparator":  [ "\u2009", ",", ".", "" ],
            "decimalSeparator":  [ ",", "." ]
        }
    }
}
```

Setting `userVisible` to true enables the user to choose their group and decimal separators from the Preferences dialog . The options they see are those listed under `groupSeparator` and `decimalSeparator`. You can enforce the formatting of separators by setting `userVisible` to false and entering only one separator type in each array .

If the `numberFormatting` setting is missing, MI:Explore will use the default operating system settings.

3.18 Language options

The display language for MI:Explore is specified in preferences.json, using the `language` setting :

```
"preferences":  {
    ...
    "language":  {
        "userVisible":  true,
        "defaultValue": "en"
    }
}
```

Setting `userVisible` to true enables the user to choose their language in the Preferences dialog.

You can enforce the language by setting `userVisible` to false and specifying one `defaultValue`.

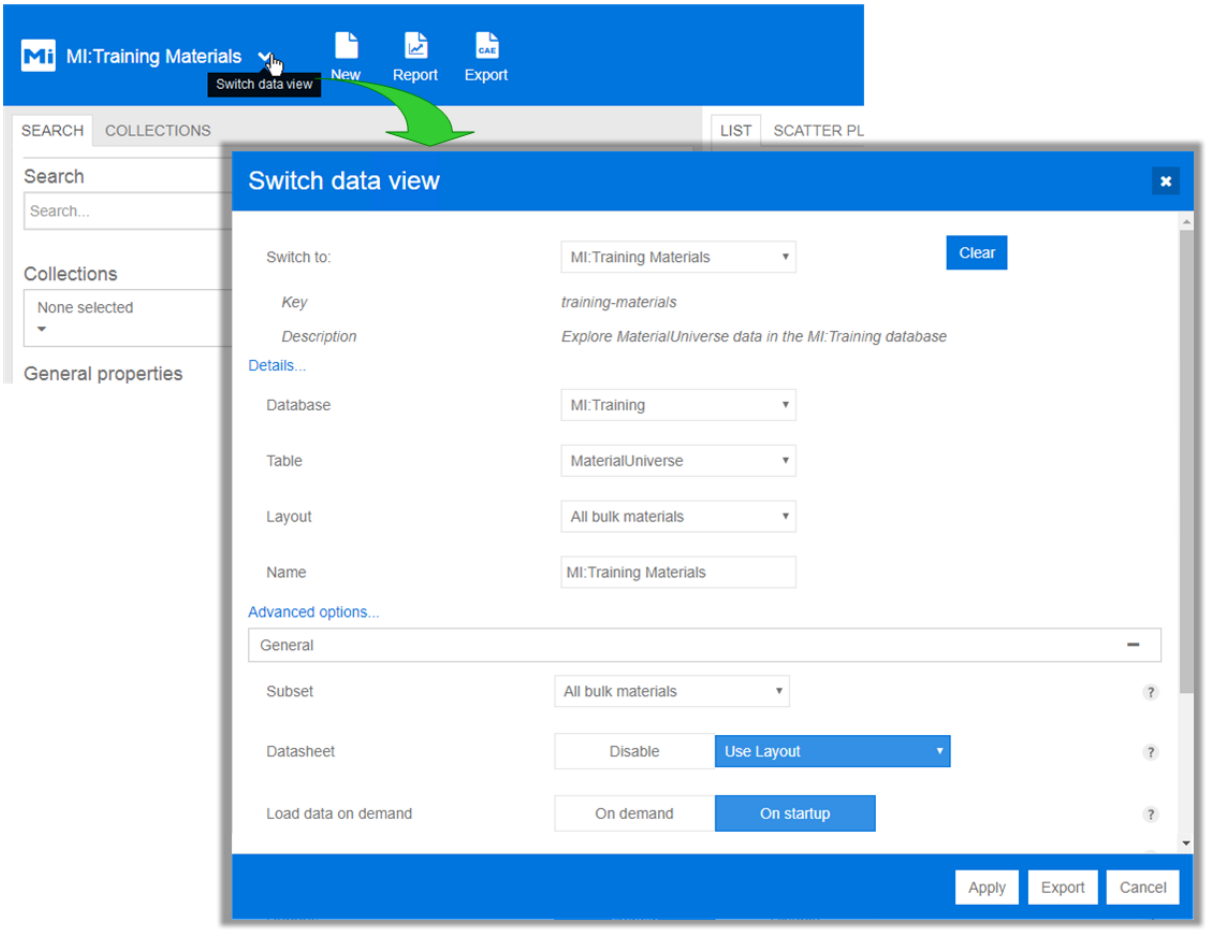
Available language options are:

- en
- fr

4 Edit data view configuration using MI:Explore

Using MI:Explore, users can select which data view to use from the Switch data view dialog. They can also use this dialog to customize their data view, and to export their customized view as a configuration file.

To launch the Switch data view dialog, click ⍰ next to the display name.



The options visible here correspond with settings in the application configuration file.

When you click Apply, the URL updates to reflect the new data view.

To save a custom data view, export it from MI:Explore and use MI:Admin to store it with the appropriate database. For further information, see section *5, Managing configuration files with MI:Admin*.

4.1 Enable Switch data view dialog for users

You can configure MI:Explore to allow users to switch data views, edit data views, or you can limit the data views a user sees, by editing the visibility of the Switch data view dialog and the options available on it. The visibility of the dialog is set using the `configSwitching` setting in the configuration file. There are five possible values:

configSwitching value	Options visible in MI:Explore to users		
	Switch data view	Edit database, table, and layout	Edit advanced options
hidden			
keyOnly	?		
keyOrData	?	?	
advanced	?	?	?
keyInGroup	?*		

*Users can only switch data view between views in a certain group.

For example:

```
"configSwitching": "keyOnly"
```

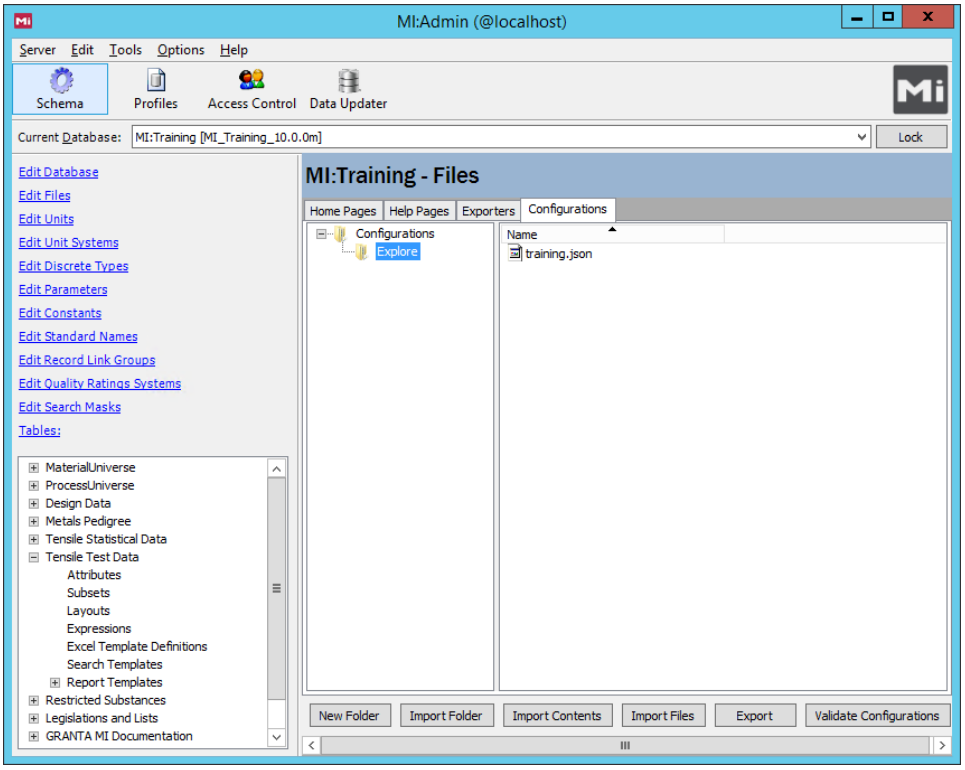
5 Managing configuration files with MI:Admin

MI:Explore configuration files are managed using MI:Admin. They are stored with the database they apply to, in a subfolder under Configurations.

To view these search configurations in MI:Explore, the location of the JSON configuration files must be specified in preferences.json, using the `configurationsPath` setting. The preferences.json file by default contains a path to “Explore”:

```
    "appName": "Explore",
    ...
    "configurationsPath": "Explore",
```

If you have custom configuration files that you want to use in MI:Explore, create a subfolder called “Explore”, and import your files into this folder:



Click **Validate Configurations** to verify the JSON syntax after importing a new configuration file.

You can specify only one configuration path in preferences.json. If you have multiple databases, create an "Explore" subfolder within each database, and store your configuration files for each database in the appropriate subfolder.

> **Note:** If you have a search.json file from a previous version of MI:Explore, you can replace the file in the installation directory with your custom file, and then delete the configurations path from preferences.json. MI:Explore will now load your configurations from search.json. This is only recommended if you upgraded to MI:Explore 5.3 from MI:Explore 4.1 or earlier.

For further information, see the *MI:Explore Help*, and the *MI:Admin Help*.

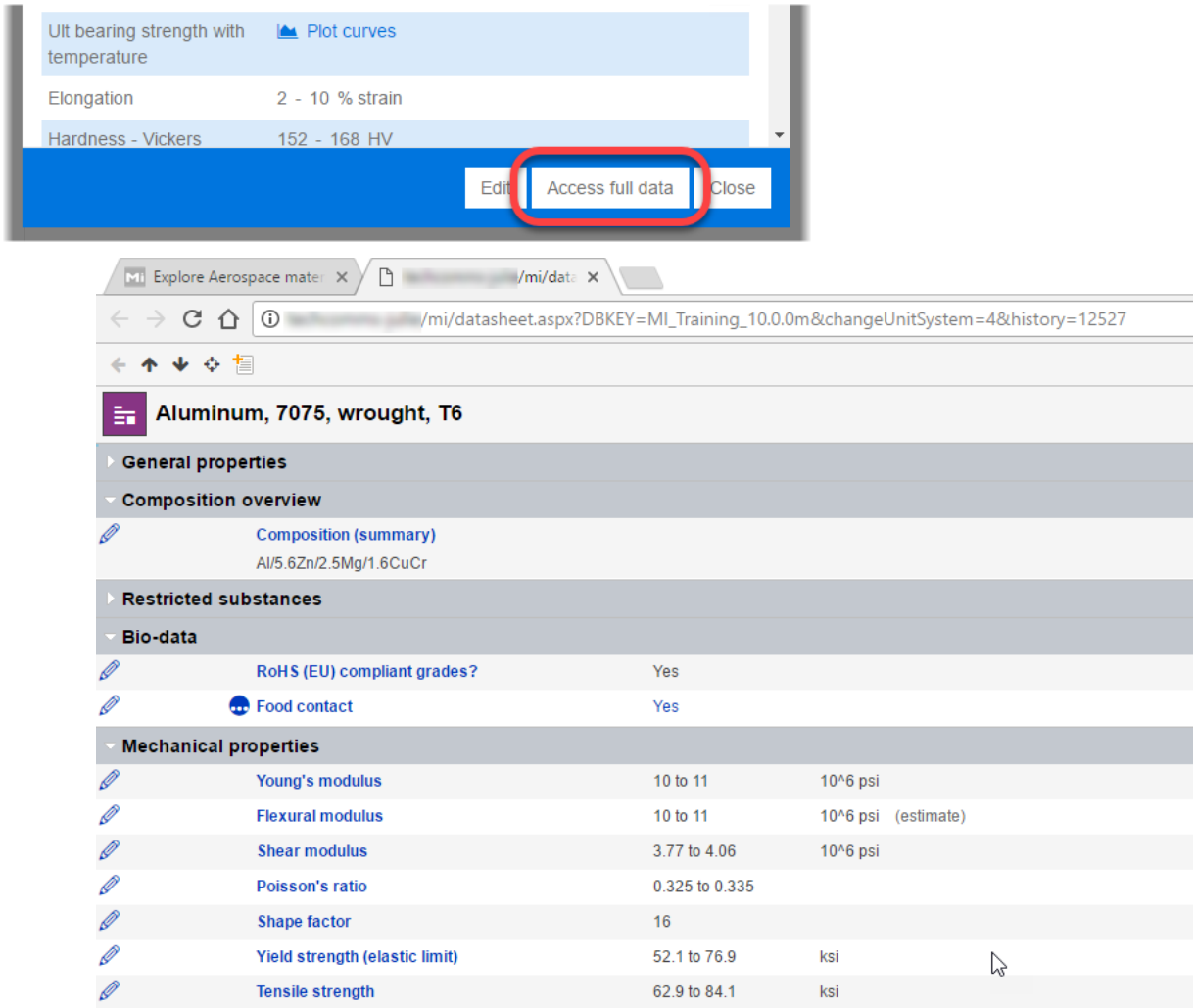
6 MI:Viewer integration

To allow users to access MI:Viewer datasheets from MI:Explore, your MI:Viewer application URL must be specified in preferences.json using the `viewerURL` setting as follows:

```
{
    "appName": "Explore",
    ...
    "viewerUrl": "http://my_miserver/mi/",
```

When specified, an **Access full data** button appears at the bottom of MI:Explore datasheets; users can click this to load the MI:Viewer datasheet for the selected record in a new browser tab.

For example:



If no MI:Viewer URL is specified in preferences.json, the **Access full data** button will not appear in the MI:Explore datasheet.

7 MI:Workflow integration

If MI:Workflow is installed in your system, MI:Explore can be integrated with it, allowing users to launch the Workflow Manager from a selected record within MI:Explore, and view the active workflows that are being progressed on that record.

Workflows can be accessed from MI:Explore if both of the following configuration steps are carried out:

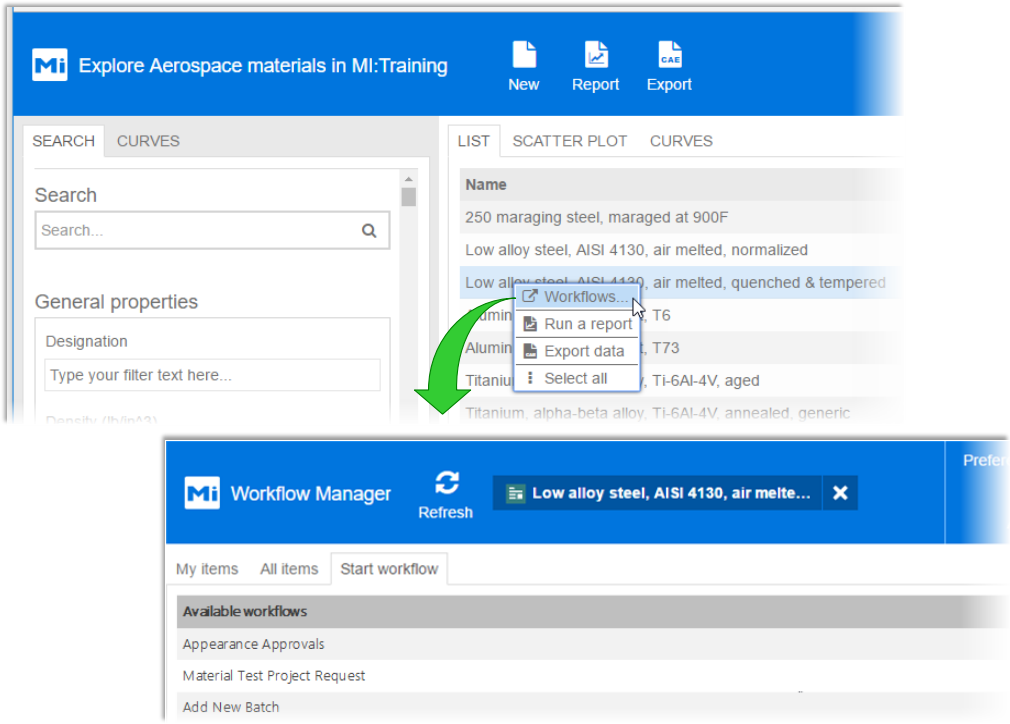
1. The MI:Workflow URL must be specified in preferences.json, as follows:

```
{
  "appName": "Explore",
  "helpUrl": "http://support.grantadesign.com/resources/miexplore/5.3/en/help/index.htm",
  "viewerUrl": "http://my_miviewer_server/mi/",
  "workflowUrl": "http://my_miworkflow_server/mi_workflow/ui/",
```

2. The **Workflows** shortcut (right-click) menu option in MI:Explore must be enabled in the configuration file using the workflowEnabled setting:

`"workflowEnabled": true`	The **Workflows** option will appear in the menu.
`"workflowEnabled": false`	The **Workflows** option will not appear in the menu.

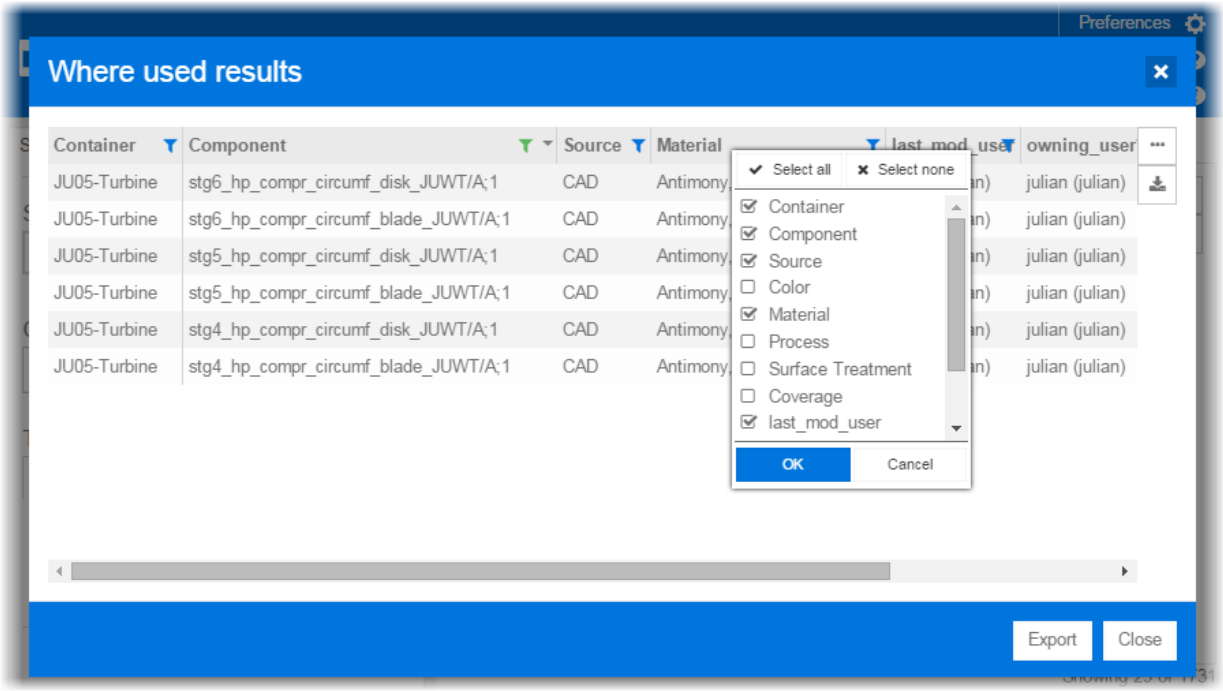
When both of these settings are configured – URL is specified and the menu item enabled – MI:Explore users will be able to launch the MI:Workflow Manager in a new browser tab from the shortcut (right-click) menu on a record in MI:Explore. For example:



8 Teamcenter® integration

MI:Explore can query instances of assignment of records to parts in Teamcenter®, where the assignments were made in the NX CAD application using MI:Materials Gateway. This is referred to as a "Where Used" query.

MI:Explore users can click **Find where used** for selected records and a "Where Used" query is sent to the Teamcenter Server; the results are displayed in MI:Explore, where they can be sorted and filtered, and exported to a file in CSV format.



The "Where Used" functionality in MI:Explore depends on MI:Connect being installed and configured to query your Teamcenter server for the assignments. For information on how to do this, see:

- *MI:Connect Installation and Configuration Guide*
- *MI:Materials Gateway for Teamcenter Integration Guide*

8.1 'Find where used' configuration in MI:Explore

Assignments in Teamcenter are of four distinct types; Material, Process, Finishing Process, and Color. The assignment type must be specified in the configuration file using `recordType`, for example:

```
"recordType": "Material",
```

If this value is missing, the *Where Used* query will not be available in the menu in MI:Explore. If present, you need to carry out the configuration in Section *8.2* below.

8.2 Where Used Teamcenter connection configuration

MI:Explore's ability to query instances of assignments of records to parts in Teamcenter uses MI:Connect to run the "Where Used" query in Teamcenter. MI:Connect is a general purpose communications service that allows clients, such as MI:Explore, to talk to 3rd party servers, such as PLM systems.

Configuration settings for MI:Connect and Teamcenter are located in the file service-settings.js in the MI:Explore installation at:

config-local/service-settings.js

8.2.1 Required MI:Connect server settings

The following settings must be configured by uncommenting and editing the values of the following constants:

`gdlConnectServerProtocol`	- Either http:/ or https:/
`gdlConnectServerAddress`	- The server name on which MI:Connect is installed. If installed on the same server as MI:Explore, this could be *localhost*. - The port number should be appended, if not default (80 for http, 443 for https).
`gdlConnectServerPluginsAddress`	- The path of plugins within MI:Connect.

8.2.2 Additional settings

"Where Used" query results from Teamcenter are returned with some optional additional information about the parts in Teamcenter. The extra columns of Teamcenter data are specified using a preference in Teamcenter; see the *MI:Materials Gateway for Teamcenter Integration Guide* for details. To see the additional column values in the "Where Used" results in MI:Explore, the name of the Teamcenter Preferences must be specified in *service-settings.js* as follows:

`gdlConnectTcAdditionalColumnsPreference`	- The name of the Preference in Teamcenter which specifies the additional columns attribute names.
`gdlConnectTcDisplayNamesPreference`	- The name of the Preference in Teamcenter which specifies the additional columns display names.

8.2.3 Sample configuration

```
WebPlatform.Mi.Connect.connect
    .value("gdlConnectServerProtocol", "https:/")
    .value("gdlConnectServerAddress", "/localhost:443")
    .value("gdlConnectServerPluginsAddress", "/miconnect/plugins");

Webplatform.Mi.Connect.connectTc
    .value("gdlConnectTcAdditionalColumnsPreference", "ADDITIONAL_PREFERENCE_NAMES")
    .value("gdlConnectTcDisplayNamesPreference", "ADDITIONAL_PREFERENCE_DISPLAY_NAMES");
```

9 MI:Explore configuration file reference

The MI:Explore configuration files are stored in the database and managed using MI:Admin. Configuration settings in these files are specified in one or more JSON objects (enclosed in braces). Each configuration object includes a set of name/value pairs consisting of field name (case-sensitive, in double quotes), followed by a colon, followed by a value. An example configuration JSON file is shown in section *10*.

Setting	Mandatory/ Optional	Summary	Switch data view dialog setting
key	Mandatory	Defines an MI:Explore data view	Key
table	Mandatory	A GRANTA MI table	Table
searchLayout	Mandatory	Layout that defines the search filter attributes	Layout
collectionsDisabled	Optional	Enable/disable the Collections feature	Collections
configSwitching	Optional	Defines user ability to change and edit data views	*N/A*
curves	Optional	Enables/disables the functional data plotting feature, and defines the default axis properties for plots	Curves
databaseKey	Optional	A GRANTA MI database key (dbkey)	*N/A*
dataSheetLayout	Optional	Layout that defines the attributes shown on datasheets	Datasheet
default	Optional	Identifies the default configuration for the application, where more than one configuration object (key) is defined	*N/A*
description	Optional	Custom information about the data view	Description
displayName	Optional	Specifies the name displayed in the application header	Name
editableDatasheetLayout	Optional	Enables and disables data add/edit functionality, and specifies the layout used when adding or editing data.	Datasheet editing
exportersDisabled	Optional	Enables/disables the Exporters feature	Exporters
exportersUseCurrentUnitSystem	Optional	Specifies the unit system use when exporting data	Unit system
exporterApplicability	Optional	Filters the exporters available in MI:Explore	Applicability
groups	Optional	Specifies which group a data view belongs to	*N/A*
loadDataOnDemand	Optional	Specifies if data is loaded on startup or on demand	Load data on demand
logSliders	Optional	Use logarithmic or linear scale in slider controls	Search slider scale

Setting	Mandatory/ Optional	Summary	Switch data view dialog setting
newRecordLocation	Optional	Specifies the folder where new records are placed	*N/A*
newRecordTransformation	Optional	Defines how new records are named	*N/A*
newRecordPaddingOptions	Optional	Defines how new records are named	*N/A*
primaryImageAttributeName	Optional	Defines the primary image used on the THUMBNAILS tab	Image
recordType	Optional	Defines Teamcenter integration options	*N/A*
reportsDisabled	Optional	Enables/disables the Reports feature	Reports
showBlanks	Optional	Show/hide attributes with no value	Show empty attributes
staticData	Optional	Options for faster application startup	*N/A*
subset	Optional*	Subset into which new records will be placed. * Mandatory if an editable datasheet layout is specified	*N/A*
tabularEditingEnabled	Optional	Enables/disables ability to edit tabular data	Tabular data editing
workflowEnabled	Optional	Enables/disables MI:Workflow integration	*N/A*
xyChart	Optional	Defines the default axis properties for scatter plots	Scatter plot, X-axis, Y-axis, Axis selection

N/A means the setting cannot be configured from the Switch data view dialog.

9.1 key

(*Mandatory*) Identifies an MI:Explore data view, that is, a set of configuration options (title, database, table, layouts, and so on) for the application.

When more than one data view is defined in the configuration files, the key can be used as a query string appended to the MI:Explore application URL, identifying which set of configuration settings (database & table, layouts, and so on) to use when opening the application; see *Defining different data views*.

The key cannot contain spaces or special characters.

Example:

```
{
    "configurations": [
        {
            "key": "materialuniverse",
            ...
            ...
        },
        {
            "key": "alloys",
            ...
            ...
        }
    ]
}
```

The corresponding URL for the *materialuniverse* `key` in the example above would be:

http://myserver/explore/?search-config=materialuniverse

while the URL corresponding to the *alloys* key above would be:

http://myserver/explore/?search-config=alloys

9.2 table

(*Mandatory*) A table in the database that the configuration file is stored with. For example:

```
"table": "MaterialUniverse"
```

See also: Database, table, and subset to search

9.3 searchLayout

(*Mandatory*) A layout that defines which attributes are available for filtering search results in MI:Explore. The layout must be stored in the table specified in `table`.

Example:

```
"searchLayout": "Stainless alloys"
```

Search layouts with a large number of attributes will affect application startup time, as data for each attribute included in the layout has to be fetched from the database. To optimize startup time, you may wish to create multiple configurations (using multiple keys) on the same database and table, each with a smaller number of targeted search attributes.

Note that search layouts should not be confused with ***search templates***, which are used to specify pre-defined search criteria (attributes and meta-attributes) to be applied when searching for data in MI:Viewer.

See also: Search panel

9.4 collectionsDisabled

(*Optional*) A Boolean that specifies whether or not the Collections feature is turned on in the application.

- `true`: the Collections feature is disabled, and no Collections tab will appear in MI:Explore.
- `false`: the Collections feature is enabled, and a Collections tab will appear alongside the Search panel in MI:Explore.

Example:

```
"collectionsDisabled": false
```

If `collectionsDisabled` is not present in the configuration file, collections will be **enabled**.

See also: Collections

9.5 configSwitching

(Optional) Specifies options for a user to change or edit a data view.

- `hidden`: disables the Switch data view dialog for users
- `keyOnly`: users can switch data view but cannot edit a data view
- `keyInGroup`: users can switch data view only between views in the same group as the default view, and cannot edit a data view
- `keyOrData`: users can switch data view and change database, table, and layout, but cannot edit advanced options
- `advanced`: users can switch and edit a data view

The `configSwitching` setting is specified in (and read from) the default data view in the configuration file. If there is no default data view, this setting should be specified in the first data view in the configuration file.

If `configSwitching` is not specified, then it defaults to `advanced`.

Example:

```
“configSwitching”: “keyInGroup”,
“groups”: [“metals”]
```

This configuration will only show data views in the metals group.

See also: Edit data view configuration using MI:Explore

9.6 curves

(*Optional*) An array of objects that specify various functional data plot properties:

`yAttribute`
: The attribute shown on the plot Y-axis by default. If missing, the first attribute found will be used.

`xParameter`
: The parameter shown on the plot X-axis by default. If missing, the first parameter found will be used.

`xAxisLogarithmic`
: A Boolean which specifies the scale used initially for the X-axis.

`yAxisLogarithmic`
: A Boolean which specifies the scale used initially for the Y-axis.

Example:

```
"curves": {
      "yAttribute": "Stress-Strain",
      "xParameter": "Strain",
      "xAxisLogarithmic": false,
      "yAxisLogarithmic": true
}
```

Note that if load data on demand is enabled:

- functional attributes will only be visible on the Curves panel if the attribute is marked as searchable in the MI:Admin database schema.
- axis scale is always displayed as linear (i.e. `"xAxisLogarithmic": false`) regardless of configuration file setting.

See also: Curves

9.7 databaseKey

(*Optional*) The database key of the GRANTA MI database containing the data to be searched in MI:Explore. If `databaseKey` is not specified, the value defaults to the key of the database that the configuration file is stored with. Specify a `databaseKey` to explicitly assign the data view to a database, or if you have configurations for different databases saved in the same configuration file.

Database keys are not case-sensitive in GRANTA MI. For example:

```
"databaseKey": "MI_Training_11.0.0m"
```

See also: Database, table, and subset to search

9.8 dataSheetLayout

(*Optional*) A layout that defines which attributes are included on MI:Explore datasheets.

The layout must be stored in the table specified in `table`. This can be a different layout from the one specified in `searchLayout`, for example, you might select a layout that included all of the attributes defined in the table.

Example:

```
"dataSheetLayout": "Polymers"
```

See also: Search panel

9.9 default

(*Optional*) Where the configuration file includes more than one configuration object (see key), this field is used to identify the default configuration for MI:Explore (`"default": true`). If no query string is specified in the URL, this will be the configuration that is used on starting the application

If set to `true`, this is the configuration that will be loaded when no `key` is specified. If more than one configuration is set to `"default": true`, the first one in the file will be used.

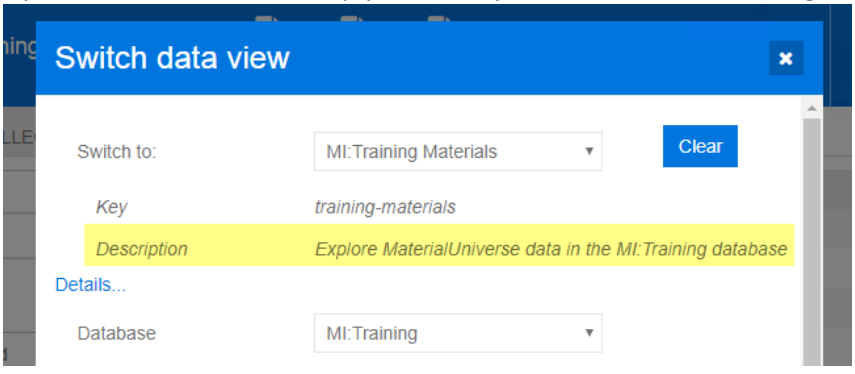
Example:

```
"key": "matuni",
"default": true,
"displayName" : "Explore Material Universe",
```

See also: Defining different data views

9.10 description

(Optional) Information to help you identify the data view when using the Switch data view dialog.



Example:

```
"description" : " Explore MaterialUniverse data in the
MI:Training database"
```

See also: Defining different data views

9.11 displayName

The name displayed in the header of the MI:Explore application.

Example:

```
"displayName" : "Explore Material Universe"
```

See also: Application title

9.12 editableDatasheetLayout

(*Optional*) Enables and disables data add/edit functionality in MI:Explore, and specifies the layout used for adding/editing data. The layout must be stored in the table specified in the `table` setting.

Example:

```
"editableDatasheetLayout": "All attributes"
```

See also: Data add and edit options, Datasheets

9.13 exportersDisabled

(*Optional*) A Boolean that specifies whether or not the **Export** option is available on the MI:Explore toolbar.

- `true`: the Export option is not available in the toolbar
- `false`: the Export option is available in the toolbar (default)

Example:

```
"exportersDisabled": false
```

See also: Export options

9.14 exportersUseCurrentUnitSystem

(*Optional*) A Boolean that specifies whether FEA exporters, if enabled, use the unit system currently specified in MI:Explore Preferences, or use the unit system specified in the exporter configuration.

- `true`: the unit system specified in the MI:Explore Preferences will be used
- `false`: the unit system specified in the exporter will be used (default)

Example:

```
"exportersUseCurrentUnitSystem": false
```

See also: Units

9.15 exporterApplicability

(*Optional*) Filters the exporters which will appear in the **Available exporters** list.

When this setting is included in the configuration file, the specified value is compared with the `Applicability` tag value specified in the exporter configuration .exp file (for example, *MIViewer* or *MIMaterialsGateway*); only exporters with the same value will be listed.

Example:

```
"exporterApplicability": "MIViewer"
```

See also: Hiding application-specific exporters

9.16 groups

(*Optional*) Specifies the group or groups that the data view belongs to. Data views in the Switch data view dialog are arranged by groups. If `configSwitching` is set to `keyInGroup`, then only the data views in the same group as the default data view will be available.

A data view may belong to one group, many groups, or no groups.

Example:

```
"groups": ["Metals", "On Demand"]
```

The `groups` value must always be specified in an array.

See also: configSwitching

9.17 loadDataOnDemand

(Optional) Specifies whether data is loaded only when required (on demand), or whether all data is loaded on startup.

- `true`: data is loaded on demand
- `false`: all data is loaded on startup

Example:

```
"loadDataOnDemand": true
```

See also: Load data on demand

9.18 logSliders

(*Optional*) Specifies whether or not slider controls in the **Search** panel are logarithmic when the data spread is larger than one order of magnitude.

- `true`: slider controls will use a logarithmic scale
- `false`: slider controls will always use a linear scale

Example:

```
"logSliders": true
```

See also: Logarithmic sliders

9.19 newRecordLocation

(*Optional*) `newRecordLocation` specifies the folder in the database where new records are placed when adding records in MI:Explore.

Users can create new records in MI:Explore if

- They have write access to the database, and
- An `editableDatasheetLayout` Is specified in the configuration file.

Example:

```
"newRecordLocation": "Metals and alloys/New records/",
```

The specified folder will be created if it does not already exist.

If `newRecordLocation` is not specified, new records will be placed in the top-level (root) folder.

The path specified in `newRecordLocation` can include attribute names; the value of the attribute will then be substituted in the new record path. Attribute-based path elements are enclosed in curly brackets `{}`. For example:

```
"newRecordLocation": "Metals and alloys/New folder/",
"newRecordLocation": "Metal and alloys/New folder/{AttributeName}",
```

The data (attribute) types supported for use as path placement attributes are:

- Integer
- Point (but units are ignored)
- Range (but units are ignored)
- Discrete
- Date
- Short text. Alphanumeric characters other than underscore (_) , dash (-) , period (.), forward slash (/), space, and left/right parentheses ( ) are stripped out.

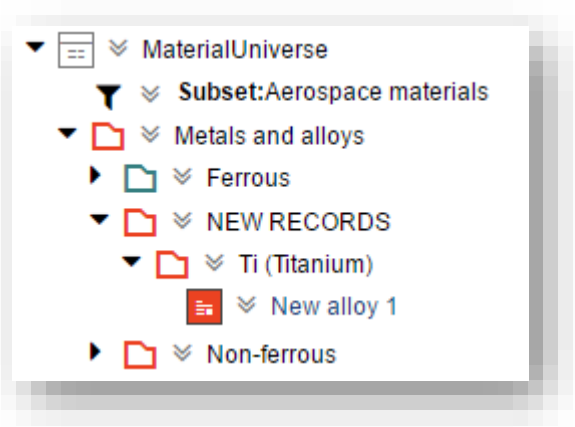
Multi-value point and multi-value discrete types cannot be used as path placement attributes.

Example:

To place new records in the folder determined by the value of the Discrete attribute *Base*, as specified by the user :

```
"newRecordLocation": "Metals and alloys/NEW RECORDS/{Base}"
```

A new titanium record named *New alloy 1*, will be created as follows:



To create new records and folders, users must have write permission to the parent folder, in the example above, *Metals and alloys*.

If an invalid attribute name is given, it will be interpreted literally, that is, it will fall back to a fixed path. For example, the following are equivalent:

```
"newRecordLocation": "Metals/New records/{invalidAttributeName}",
"newRecordLocation": "Metals/New records/invalidAttributeName",
```

See also: Placement of new records

9.20 newRecordTransformation, newRecordPaddingOptions

(*Optional*) When new records are created using MI:Explore, properties of the new record, such as its name, and/or a unique reference number attribute, can be automatically generated. This is configured using the `newRecordTransformation` and `newRecordPaddingOptions` keys in the configuration file.

See Section *3.14, Options for autonaming of new records*, for examples showing how these configuration options can be combined in different ways.

9.20.1 newRecordTransformation

(*Optional*) The value of the `newRecordTransformation` key is an array of optional record transforms, which may be used in any combination to construct the new record name.

Record Transform	Function	Standard Names	Mapped to attribute of type	Required ?
`Counter`	Increments the value of an integer attribute	Counter	Integer	Y
`Id`	Takes two attribute values and concatenates them into a third attribute value	Id_Prefix	Discrete	N
		Id_Suffix	Integer, Short text	N
		Id	Short text	Y
`RecordNameConcatenator`	Concatenates two attributes and writes them into the record name	Name_Prefix	Integer, Short text, Discrete	Y
		Name_Suffix	Integer, Short text, Discrete	N

The record transforms are applied in the order they are specified.

`Counter`

Increments an integer attribute with every new record created.

When this transform is included, an integer attribute with the standard name **Counter** is populated on the newly-created record, with a value that increments by 1 for each new record created.

`Id`

Concatenates two attribute values into a third attribute.

When this transform is included, a Short text attribute with the standard name **Id** is populated automatically when a new record is created. Its value will be a concatenation of the attributes with the standard names **Id_Prefix** and **Id_Suffix**.

`RecordNameConcatenator`

Concatenates two attribute values into the record name.

When this transform is included, the new record name is automatically populated; the user will not be able to type in a record name. The new record name will be a hyphen-separated concatenation of the attributes with the standard names **Name_Prefix** and **Name_Suffix**. If **Name_Suffix** is omitted, then there will be no hyphen.

Example:

```
"newRecordTransformation": [ "Counter", "Id",
"RecordNameConcatenator" ],
```

9.20.2 newRecordPaddingOptions

The `newRecordPaddingOptions` key contains two optional values:

`padLength`

When used in conjunction with the `Id` record transform, this value will be used to pad the number out to the specified width. For example:

- **1**, padded to a length of 6: **000001**
- **1891**, padded to a length of 6: **001891**

`padString`

If specified, this string is used as the padding character.

- If `padLength` is unspecified, there will be no padding.
- If `padLength` is specified, but `padString` is not, the **Id_Suffix** will be padded with "0".
- If both `padLength` and `padString` are specified, **Id_Suffix** will be padded to `padLength` with the `padString`.

Example:

```
"newRecordPaddingOptions": {
      "padLength": 6,
      "padString": "0" }
```

9.21 primaryImageAttributeName

(*Optional*) The attribute that defines the primary image used on the MI:Explore THUMBNAILS tab. For example:

```
 "primaryImageAttributeName":  "BH curve"
```

See also Thumbnails

9.22 recordType

(*Optional*) Defines the Teamcenter assignment type for Where Used queries, where the record type value must be one of:

- Material
- Process
- FinishingProcess
- Color

Example:

```
"recordType": "Material"
```

See also Teamcenter® integration

9.23 reportsDisabled

(*Optional*) A Boolean that specifies whether or not the **Report** option is available in MI:Explore.

- `true`: the Report option is not available in the toolbar
- `false`: the Report option is available in the toolbar (default)

Example:

```
"reportsDisabled": false
```

See also Reports

9.24 showBlanks

(*Optional*) A Boolean that specifies whether or not attributes that do not have a value are shown when users are viewing datasheets in MI:Explore.

- `true`: Show attributes with no value (*default*)
- `false`: Hide attributes with no value

If not specified, the default behavior is to show attributes with no values. Note that this option only applies when viewing datasheets; attributes that do not have a value are always shown when users are editing datasheets.

Example:

```
"showBlanks": false
```

See also: Showing/hiding attributes with no value

9.25 staticData

(*Optional*) Specifies one or more static (reference) data files. For example:

```
    "staticData": [
      "./MIL-standards.json",
      "./Patents-coatings.json"
    ],
```

This should only be used if MI:Explore is configured to load data on startup (see loadDataOnDemand).

See also: Using static data files for faster startup

9.26 subset

(*Optional*) Specifies a subset containing records of interest.

- If specified, search and filter operations in MI:Explore will be limited to only the records in this subset.
- If no subset is specified, search and filter operations will be applied to all records in the table.

The subset must be defined in the table specified in table.

Example:

```
"table": "MaterialUniverse",
"subset": "Magnetic materials",
```

Where users have the ability to add records (see editableDatasheetLayout), then a subset must be specified; all new records will be placed in this subset.

See also: Database, table, and subset to search

9.27 tabularEditingEnabled

(Optional) A Boolean that specifies whether or not tabular data is editable in MI:Explore.

- `true`: tabular data is editable
- `false`: tabular data is not editable (default)

Example:

```
"tabularEditingEnabled": true
```

See also: Viewing and editing tabular data

9.28 workflowEnabled

(*Optional*) A Boolean that specifies whether or not to launch the MI Workflow Manager from MI:Explore. If not specified, the default behavior is not to show the menu option.

- `true`: Show menu option to launch Workflow
- `false`: Hide the menu option to launch Workflow (default)

Example:

```
"workflowEnabled": false
```

See also: MI:Workflow integration

9.29 xyChart

(*Optional*) An array of objects that specify various properties for scatter plots:

`xAttribute`

The attribute shown on the plot X-axis by default.

`yAttribute`

The attribute shown on the plot Y-axis by default.

`xAxisLogarithmic`

A Boolean which specifies the scale used initially for the plot X-axis.

`yAxisLogarithmic`

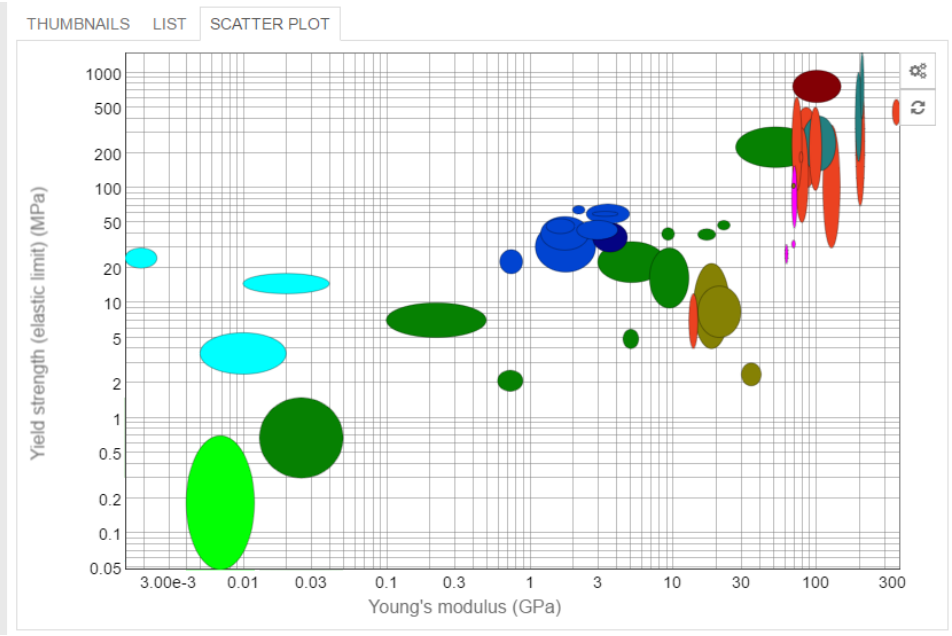
A Boolean which specifies the scale used initially for the plot Y-axis.

`preventAxisChange`

A Boolean that controls whether or not the Settings panel is available to users. If *true*, MI:Explore users will be unable to change any plot settings (X- or Y- axis attributes or scale).

Example:

```
"xyChart": {
      "xAttribute": "Young's modulus",
      "yAttribute": "Yield strength (elastic limit)",
      "preventAxisChange": false,
      "xAxisLogarithmic": true,
      "yAxisLogarithmic": true
}
```



See also: Scatter plots

10 Sample configuration file

This is a sample JSON file that can be stored and used with the MI Training database.

```
{"configurations": [
      {
            "key": "gdl-sample-aerospace",
            "displayName" : "Explore Aerospace materials",
            "description": "Explore Aerospace materials in the
MaterialUniverse database",
            "default": true,
            "table": "MaterialUniverse",
            "searchLayout": "Aerospace materials",
            "dataSheetLayout": "Aerospace materials",
            "editableDatasheetLayout": "Aerospace materials",
            "subset": "Aerospace materials",
            "newRecordLocation": "From Default",
            "logSliders": true,
            "xyChart": {
                  "xAttribute": "Young's modulus",
                  "yAttribute": "Tensile strength",
                  "preventAxisChange": false,
                  "xAxisLogarithmic": false,
                  "yAxisLogarithmic": false
            },
            "curves": {
                  "yAttribute": "Young's modulus",
                  "xParameter": "Temperature",
                  "xAxisLogarithmic": false,
                  "yAxisLogarithmic": true
            },
            "collectionsDisabled": true,
            "showBlanks": false,
            "reportsDisabled": false,
            "exportersDisabled": false,
            "exporterApplicability": "MIViewer",
            "tabularEditingEnabled": true,
            "groups": []
      },
      {
            "key": "gdl-sample-rs",
            "displayName": "Explore Restricted Substances",
            "description": "Explore Restricted Substances data ",
            "default": false,
            "table": "Restricted Substances",
            "subset": "All substances",
            "searchLayout": "All substances",
            "dataSheetLayout": "All substances",
            "logSliders": true,
            "collectionsDisabled": false,
            "reportsDisabled": false,
            "exportersDisabled": true,
            "showBlanks": false,
```

```
            "tabularEditingEnabled": true,
            "loadDataOnDemand": true,
            "groups": []
      },
      {
            "key": "gdl-sample-polymers",
            "displayName" : "Explore Polymers",
            "description": "Explore Polymers data from the
MaterialUniverse database and loading data on demand",
            "default": false,
            "table": "MaterialUniverse",
            "searchLayout": "Polymers",
            "dataSheetLayout": "Polymers",
            "editableDatasheetLayout": "Polymers",
            "subset": "Polymers",
            "logSliders": true,
            "xyChart": {
                  "xAttribute": "Compressive strength",
                  "yAttribute": "Elongation",
                  "preventAxisChange": false,
                  "xAxisLogarithmic": false,
                  "yAxisLogarithmic": true
            },
            "collectionsDisabled": true,
            "showBlanks": false,
            "reportsDisabled": false,
            "exportersDisabled": true,
            "tabularEditingEnabled": true,
            "newRecordLocation": "NEW POLYMERS",
            "loadDataOnDemand": true,
            "groups": []
      }
      ]
}
```

Appendix A. Troubleshooting

A.1 View the log file

Event information for MI:Explore, which may be useful when troubleshooting problems, is written to the Service Layer trace log file located here:

C:\inetpub\wwwroot\mi_servicelayer\logs\MIServiceLayer.TraceMessages.csv

This is a CSV file which can be opened using Microsoft® Excel® or another program that reads files in CSV format. For each event logged, the file includes information about the application connecting to the Service Layer, the event type and description, the IP address of the client, and client browser details.

A.2 Compatibility View settings in IE11

When you access MI:Explore in IE11, if Compatibility View is enabled for intranet sites, you may see a message saying “Incompatibile browser detected”.

To disable Compatibility View, on the **Tools** menu, click **Compatibility View settings**, and clear **Display intranet sites in Compatibilty View**. If this option is unavailable, you can access MI:Explore in IE11 using the full URL e.g. http://*<mi_server>*.*<networkdomain>*.local/explore/

A.3 Enable pop-ups in your browser

If your browser is configured to block pop-ups then some features in MI:Explore, such as Reports, may be blocked.

To enable pop-ups in Google Chrome:

1. Go to **Settings**.
2. Scroll down to **Advanced**, and under Privacy and security, click **Content settings**.
3. Click **Popups**. Edit the settings to either allow pop-ups globally, or add the server name on which MI:Explore is installed e.g. http://*<mi_server>*/explore/

To enable pop-ups in Firefox:

1. Go to **Options**.
2. Click **Privacy and Security**, and scroll down to **Permissions**.
3. Either clear the **Block pop-up windows** option, or click **Exceptions** and add the server name on which MI:Explore is installed e.g. http://*<mi_server>*/explore/

Appendix B. Using static data files for faster startup

MI:Explore can be configured to load data on demand, to improve the startup time for large databases. Alternatively, if you would prefer to load data on startup, you can speed up the application startup time for large databases using *static data files*.

Databases typically contain static data, such as reference data that does not need to be modified, as well as data that can be edited. To make MI:Explore startup time faster, a copy of any static data can be stored in file(s) outside of the GRANTA MI database; the data in static data files is an exact duplicate of the "live" data in the GRANTA MI database and is used only on application startup: operations that rely on the real data, such as *Where Used* and viewing full datasheets, use the source data from GRANTA MI.

Static data may be viewed, searched, and plotted in the same way as data stored in the database; it cannot be edited, and it cannot be access-controlled.

To configure MI:Explore to use static data files, you specify the file path(s) in search.json. using the `staticData` setting. For example:

```
    "staticData": [
      "./MIL-standards.json",
      "./Patents-coatings.json"
    ],
```

Contact Granta Support for more information about implementing static data files for MI:Explore.