



Classes and properties

The importance of class structure in Document object type

- ▶ Class is probably the most important metadata for the document
 - **Always mandatory to be filled**
 - Views document in correct virtual folders and views
 - Requires the user to add additional **relevant** metadata:
 - Every quote is related to a customer or prospect
 - Every contract is effective on a certain date

Examples of class names

- ▶ Brochure
- ▶ Memo
- ▶ Manual or guide
- ▶ Drawing
- ▶ Report
- ▶ Contract
- ▶ Plan

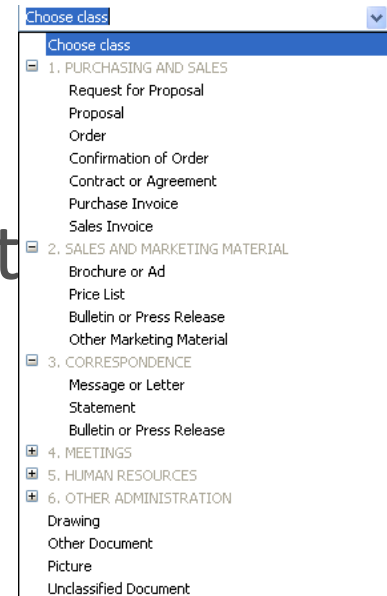
Class groups

▶ Although class groups do not affect metadata structure or finding documents, it is usually good to create 5 to 10 of them

- Eases finding the class when saving

▶ Name class groups according to function

- Meetings
 - Memo, Meeting minutes
- Production



Typical pitfalls when designing class structure

▶ Creating **too specific** class names

- specifying document should be done with properties instead of classes
 - e.g. *report type* or *meeting type*
- selecting the correct class from a long list may be difficult
- unambiguousness of class selection ruined
 - is this document *a memo* or *meeting minutes*?

▶ Naming class groups according to

Properties

- ▶ Properties are class-specific and they are used **to specify the object in more detail**
 - To which customer / project is this document related?
 - Who is responsible for this agreement?
 - When does this agreement expire?
- ▶ Typically, a good amount of properties per class is between 5 to 7, with a maximum of 5 mandatory fields

Data types of the properties

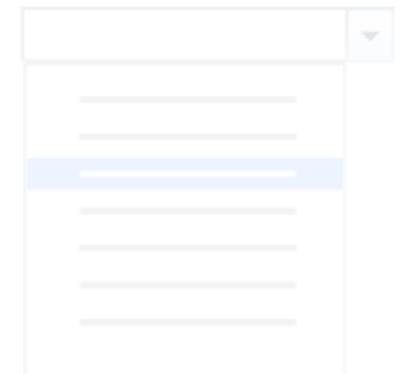
► Property data types in M-Files are the following:

- Number (integer or real)
- Text (single-line or multi-line)
 - normal text, email, hyperlink
- Lookup or Multi-select lookup (drop-down)
- Date
- Time
- Boolean (yes / no)



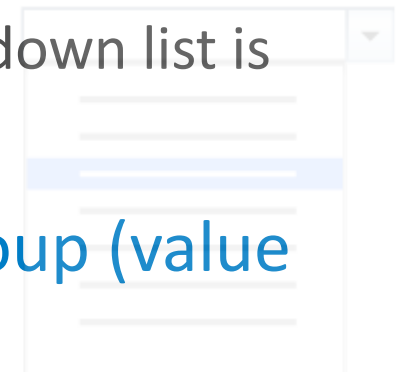
Value lists 1/2

- ▶ Using value lists whenever possible is recommended
 - The likelihood of typing errors and spelling mistakes decrease
 - Eases the use, quick to use
 - Improves abilities to create views
- ▶ Example value lists
 - Postal codes and cities
 - Meeting types



Value lists 2/2

- ▶ Value lists can be hierarchical with another value list or object type:
 - **Postal code and city**
 - City property is automatically filled after selecting postal code
 - If city is selected first, postal code drop-down list is filtered accordingly
 - **Product (object type) and product group (value list)**



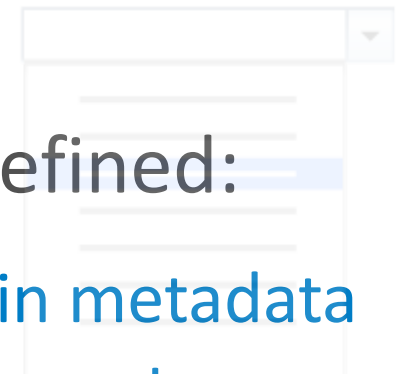
Typical pitfalls with class-specific property definitions

▶ **Too many** or **too few** properties per class

- Too few: difficult to sort and find documents as the number of documents in the vault increases
- Too many: users think of M-Files as a complex system to use and / or refuse to fill in **relevant** metadata for objects

▶ The mandatory properties poorly defined:

- Optional property: user forgets to fill in metadata that is critical in terms of finding documents



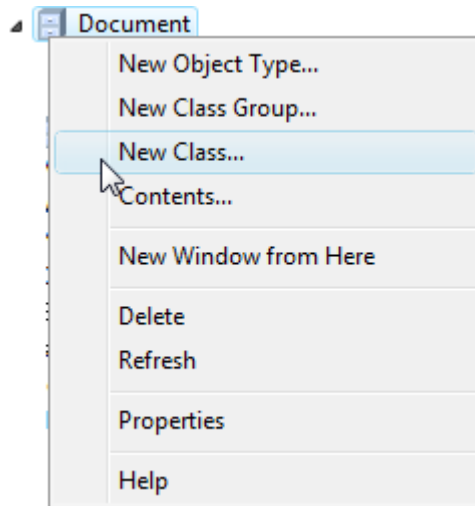
Typical pitfalls with class-specific property definitions

- ▶ Wrong data types complicate grouping documents in views
 - Lookup vs. text field
 - Virtual folders in *Customers by city* view
 - › New York
 - › NY
 - › N.Y.
 - Text vs. Number
 - If proposal sales value is of data type number, segments

Excercise 1

► Create the following classes in the Training vault:

- Meeting minutes
- Proposal
- Order
- Contract
- Memo
- Report

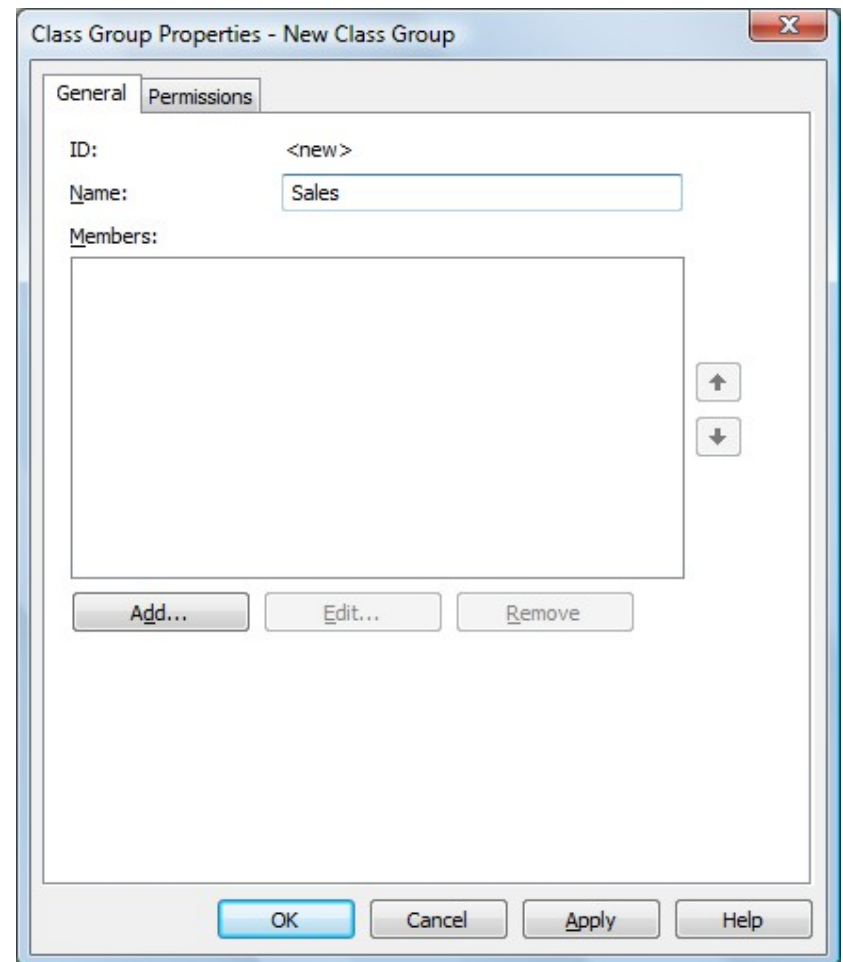
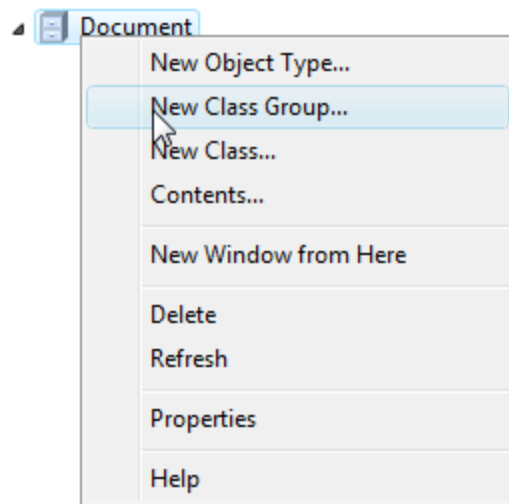
A screenshot of the 'Class Properties - New Class' dialog box. The 'General' tab is selected. The 'Name' field contains 'Memo' and the 'Object type' dropdown is set to 'Document'. The 'Properties' section contains a table with one row: 'Name or title *' with 'Text' as the data type and a checked 'Required' checkbox. Below the table are 'Add...', 'Remove', and 'Set As Name' buttons, along with up and down arrows. A note states: 'The asterisk (*) indicates the property that is used as the object name.' Below this, 'Basic properties:' are listed as 'Created; Created by; Last modified; Last modified by'. At the bottom, there is a 'Default workflow for new objects:' dropdown and 'OK', 'Cancel', 'Apply', and 'Help' buttons.

Name	Data Type	Required
Name or title *	Text	<input checked="" type="checkbox"/>

Exercise 2

► Create class groups

- Sales
- Meetings

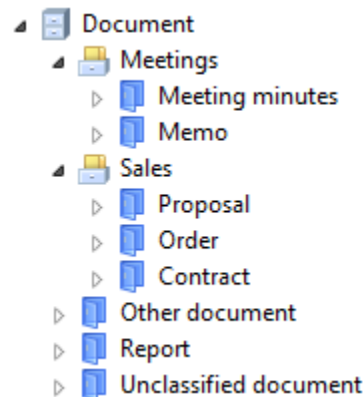


Exercise 3

- ▶ Move the classes to suitable class groups by dragging and dropping them above the class group:

- Sales
- Proposal
- Order
- Contract

- ▶ Meetings

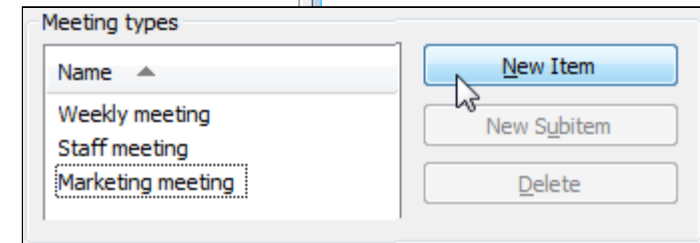
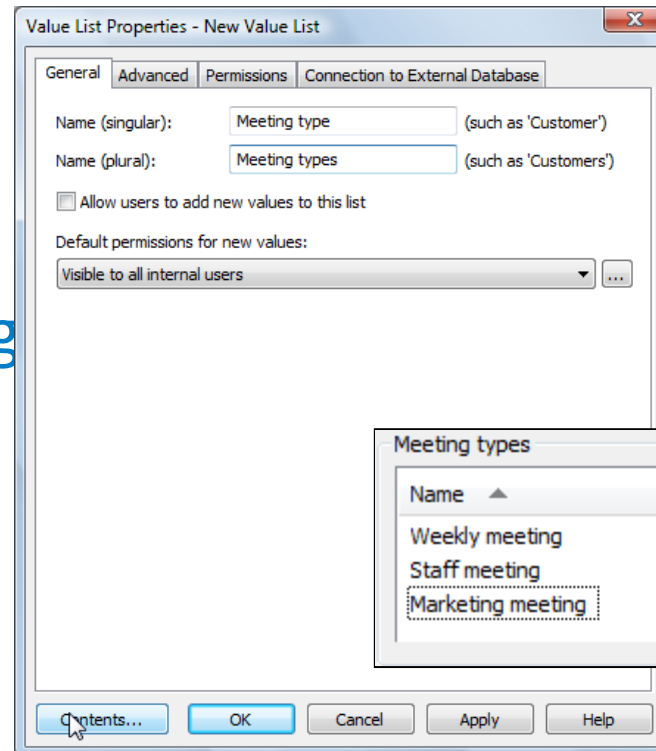
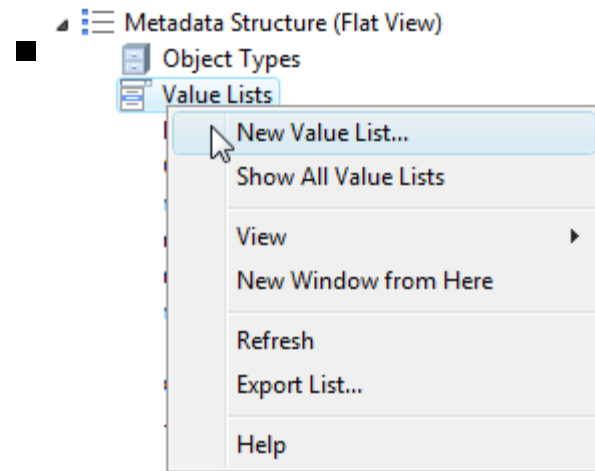


- Memo

Exercise 4

- ▶ Create a new value list **Meeting types** and add the following values in it:

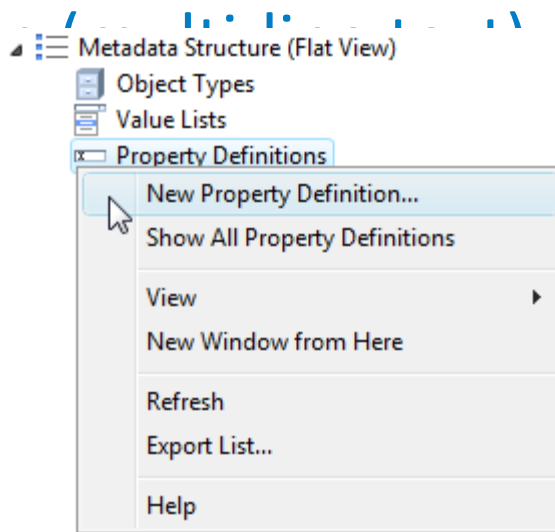
- Weekly meeting
- Staff meeting



Exercise 5

► Create the following properties

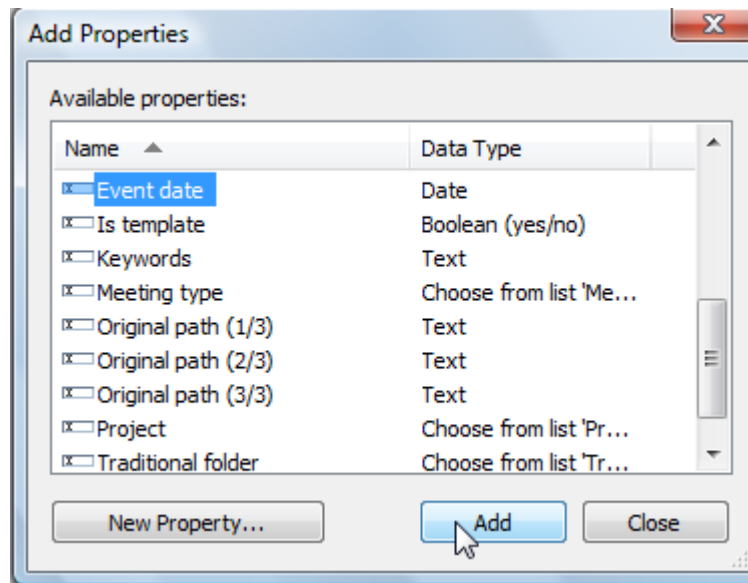
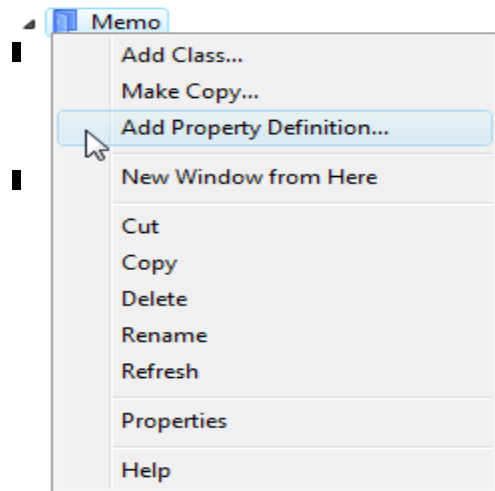
- Event date (date)
- Document date (date)
- Description

A screenshot of the 'Property Definition Properties - New Property Definition' dialog box. The 'General' tab is active. The 'Name' field contains 'Event date'. The 'Data type' dropdown is set to 'Date'. The 'Content' dropdown is empty. The 'Show values from the following list:' dropdown is empty. The 'Filter the list by using the value of the following property:' dropdown is empty. The 'Sort values in the list in the following order:' dropdown is empty. The 'Allow using this property with the following object type:' dropdown is set to 'All object types'. At the bottom, there are 'OK', 'Cancel', 'Apply', and 'Help' buttons.

Exercise 6

► Add the following properties into the **Memo** class:

- Meeting type
- Document date



Exercise 7

- ▶ Select **Class Properties of Memo** and mark the following properties **as required**:
 - Event date
 - Meeting type
- ▶ Change the order of properties to as follows:
 - Name or title
 - Meeting type
 - Event date