Property Mapping and Smart Lookup

Simple rule configuration · https://extensionkit.unitfly.com/documentation/document-ai-property-mapping-smart-lookup/

Property Mapping and Smart Lookup allow you to control how information extracted by Document AI is mapped to M-Files properties.

You can use these settings to populate metadata, match extracted values with existing objects or value list items, handle multiple extracted values, and control whether existing property values can be overwritten.

These settings are available when configuring **Set properties** in Microsoft Foundry and Azure Document Intelligence extraction actions. They are also used when mapping properties on newly created M-Files objects.

This page describes the following configuration options:

- Property mapping: Configure the source and target of property values.
- Result formatting and arrays: Control how extracted values are converted and stored.
- Lookup mapping: Match extracted values to existing M-Files objects or value list items.
- Smart lookup: Use normalization and fuzzy matching when an exact lookup match cannot be found.
- Smart lookup filter conditions: Narrow down the objects considered during fuzzy matching.

The available fields depend on the selected M-Files property type and the Document AI service or extraction model.

Property Mapping

Use **Set properties** to map values returned by the AI model to M-Files properties.

For example, you can extract an invoice number, invoice date, supplier name, or total amount from a document and populate the corresponding M-Files metadata fields.

The specific extraction field depends on the selected service:

- **Microsoft Foundry:** Use Query field to define the information the model should extract.
- **Azure Document Intelligence:** Use the available model-specific fields, such as Property name, Query field, or Key value pair, to select the extracted information.

The following settings are shared across the supported extraction models.

Set property from response

Choose whether to set the property from the AI model response or use a static property value from existing objects in the vault.

- **Yes:** Uses the value extracted from the document by the AI model.
- **No:** Uses the Static property configuration instead of an extracted value.

Static property

*Displayed when **Set property from response** is disabled.*

Use this setting to configure properties that are not extracted from the AI model response.

Values can be static or dynamic, referencing properties from the source object.

Result property

Select the M-Files property that will receive the value.

For example, if the AI model extracts an invoice number, select the corresponding Invoice Number property in M-Files.

Result M-Files property type

Specifies what the extracted result represents.

Choose the appropriate property type:

- **Simple value:** Map the extracted result directly to an M-Files property, such as text, number, or date.
- **Object lookup:** Match the extracted value with an existing M-Files object.
- **Value list lookup:** Match the extracted value with an item in an M-Files value list.

For example, an extracted invoice total can be stored as a simple value, while an extracted supplier name can be mapped to an existing Supplier object or a value list item.

Additional mapping settings become available when using object or value list lookups.

Result Formatting and Arrays

These settings control how extracted values are formatted and how multiple values are handled when mapping them to M-Files properties.

Result culture

Specifies the culture used to interpret date and number results.

This setting is available in generic create-object property entries. Microsoft Foundry and Azure Document Intelligence prebuilt property entries automatically use `en-US`.

Result format

Specifies the format used for date results.

This setting has the same availability as Result culture.

Array to text behavior

Controls how an array of extracted values is mapped to a single-value M-Files property.

Choose one of the following options:

- **TakeFirst:** Uses only the first element of the array.
- **Concatenate:** Combines all elements into one text value, separated by the configured Items separator.

For example, if the AI model extracts the values `Finance`, `Legal`, and `Operations`:

Setting	Result
TakeFirst	Finance

Concatenate (separator: `,`)	Finance, Legal, Operations

For Microsoft Foundry, this setting is displayed when the Query field ends with `[]`, indicating that the model should return an array.

Items separator

Specifies the separator used when processing multiple values.

It is used in two situations:

- **Concatenating arrays:** When Array to text behavior is set to Concatenate, the extracted values are joined into a single text value using the specified separator.
- **Splitting lookup values:** When mapping lookup properties, a single AI response string containing the specified separator is split into individual lookup items.

For example, if the AI response contains `Finance, Legal, Operations` and the separator is `,`, the values can be processed as separate lookup items.

Lookup Mapping

Lookup mapping allows Document AI to connect extracted values with existing M-Files objects or value list items.

For example, when processing an invoice, Document AI can extract a supplier name and use it to identify the corresponding Supplier object in M-Files.

Instead of saving the extracted supplier name as plain text, the value can be stored as a lookup reference.

Select **Object lookup** or **Value list lookup** under Result M-Files property type to configure the corresponding mapping settings.

Map Object lookup by

Defines how the extracted value is mapped to an existing M-Files object.

Use this setting when the target property references an object type, such as Supplier, Customer, or Project.

Additional settings allow you to specify the object type, class, and property used for mapping.

Map Value list lookup by

Defines how the extracted value is mapped to an M-Files value list item.

Use this setting when the target property references a value list, such as Department or Document Category.

Result M-Files object type

Specifies the type of M-Files objects used for the lookup mapping.

For example, select the Supplier object type when matching an extracted supplier name with an existing Supplier object.

Result M-Files object class

Specifies the class of the M-Files objects used for the lookup mapping.

Result M-Files object property

Specifies the property on the target M-Files object used for mapping.

For example, when mapping an extracted supplier name, you can use the Name property of the Supplier object type.

Try to map empty value

Controls whether the rule attempts to resolve an empty or non-existing lookup value.

When enabled, the rule can search for objects whose configured lookup property contains an empty value.

Create if doesn't exist

Controls whether a new M-Files object or value list item can be created when the extracted value does not match an existing entry.

When enabled:

- **Value list lookup:** A new value list item can be created if the value list allows users to add new values.

- **Object lookup:** A new object can be created if it does not require additional mandatory properties.

For value lists, ensure that the **Allow users to add new values to this list** option is enabled in M-Files.

For object creation, only the name and class properties are set automatically. Therefore, objects with other required properties cannot be created using this setting.

Note: This option can also be used as a fallback when Smart lookup does not find an acceptable match.

Throw if all items don't exist

Controls how lookup mapping behaves when some extracted items do not exist in the target M-Files vault.

- **Enabled:** If at least one lookup item does not exist, none of the lookup items are added to the target property.
- **Disabled:** Existing lookup items can still be added even if other items do not exist.

This setting is relevant when mapping multiple extracted values to a lookup property.

Parent property

Specifies the parent property used in the lookup configuration.

Owner property

Specifies the owner property used in the lookup configuration.

Search timeout (sec)

Specifies the maximum time, in seconds, allowed for searching M-Files objects during lookup mapping.

This setting is separate from the Fuzzy match timeout used by Smart lookup.

Smart Lookup

Smart lookup improves how extracted values are matched with existing M-Files objects and value list items.

When an extracted value does not exactly match an existing lookup item, Smart lookup can use normalization and fuzzy matching to identify a suitable match.

This is useful when documents contain different spellings, formatting variations, or slightly different names for the same object or value list item.

For example, a document might contain a supplier name that differs slightly from the name stored in M-Files. Smart lookup can attempt to identify the corresponding supplier instead of requiring an exact text match.

Enable smart lookup

Enable this option to use normalization and fuzzy matching when an exact lookup match cannot be found.

Smart lookup is available for:

- **Object Type lookups:** Match extracted values to existing M-Files objects.
- **Value List lookups:** Match extracted values to existing value list items.

When Smart lookup is enabled, additional settings become available.

Match threshold

Specifies the minimum confidence required to accept an item returned by fuzzy matching.

If the match confidence is below the configured threshold, the matched item will not be applied.

A higher threshold requires a stronger match before the extracted value is mapped to an existing M-Files item.

Fuzzy match timeout (sec)

Defines the maximum time allowed for fuzzy matching.

Default value: 60 seconds.

If matching exceeds the configured timeout, Smart lookup is aborted for that value.

If **Create if doesn't exist** is enabled, the system applies that fallback behavior.

Increase the timeout for large object sets where more extensive matching may be required. Decrease it for time-sensitive rules.

Filter conditions

Use **Filter conditions** to narrow down which M-Files objects are considered during fuzzy matching.

Instead of searching all objects of the configured type, Smart lookup evaluates only objects that match the specified filter conditions.

This can be useful when objects have similar names but differ by department, company, location, or other metadata.

Important: Filter conditions apply to Object Type lookups only.

If no filter conditions are configured, all objects of the selected object type are considered as candidates for fuzzy matching.

Smart Lookup Filter Conditions

Use filter conditions to limit the set of candidate objects before Smart lookup performs fuzzy matching.

For example, when matching an extracted supplier name, you can restrict the search to objects that have a specific property value.

Expand **Filter conditions** and add a condition.

The following settings are available for each condition.

Property

Select the M-Files property used to filter candidate objects.

For example, this could be a department, country, or another property available on the target object type.

Operator

Specifies how the candidate object's property value is compared with the filter value.

Choose one of the following options:

- **=** — The property value must equal the specified value.
- **Contains** — The property value must contain the specified text.
- **Starts with** — The property value must start with the specified text.

Value from AI response

Determines where the filter value comes from.

- **Enabled:** Uses a value from another field extracted in the AI response.
- **Disabled:** Uses a static value or an EK placeholder expression.

Response field name

Displayed when **Value from AI response** is enabled.

Specifies the name of the extracted AI response field used as the filter value.

The way this field is configured depends on the selected extraction model:

- **Microsoft Foundry:** Enter the response field name as free text.
- **Azure Document Intelligence prebuilt-layout:** Enter the response field name as free text.
- **Azure Document Intelligence prebuilt-invoice, prebuilt-receipt, and prebuilt-contract:** Select the field from the available model-specific dropdown.

Query field description

Available for Microsoft Foundry extraction.

Provides additional context to the AI model about the field being extracted, such as its expected content or format.

Value

Displayed when **Value from AI response** is disabled.

Enter a static value or an EK placeholder expression.

For example:

`%PROPERTY.123%`

This allows the filter value to be retrieved dynamically from a property on the source object.

Example: Filtering lookup candidates

Suppose Document AI extracts a supplier name from an invoice, and the M-Files vault contains suppliers with similar names.

You can configure Smart lookup to consider only suppliers matching a particular property value before attempting fuzzy matching.

For example:

Setting	Example value
Result M-Files object type	Supplier
Property	Country
Operator	=
Value from AI response	Enabled
Response field name	SupplierCountry

In this example, the filter uses the extracted country to narrow down the Supplier objects considered for matching.

Smart lookup then attempts to match the extracted supplier name against the remaining candidates.

Related Configuration

Property Mapping and Smart Lookup are used within Document AI extraction actions.

For service-specific field selection and configuration, refer to:

- Microsoft Foundry AI Action — Configure Query fields, extracted values, and property mappings using Microsoft Foundry models.
- Azure Document Intelligence AI Action — Configure property mapping for prebuilt-layout, prebuilt-invoice, prebuilt-receipt, and prebuilt-contract models.
- Document AI options — Configure general Document AI rule settings.
- Advanced rule configuration — Configure additional rule execution, searching, and error handling options.