Match and Merge Documentation

Version 3.4.1
July 2022

TIBC®

Important Information

Table of contents

Administrator Guide

Developer Guide

User Guide

Matching and Merging Walkthrough

General Tasks and Views

Release Notes

Administrator Guide

CHAPTER **1**

Overview

In an administrative role with the TIBCO EBX® Match and Merge Add-on, you can perform actions that:

- Specify the data that can participate in the add-on's matching and merging operations.

- Create a matching policy that determines *when* and *how* matching processes execute. How you enable add-on features for business users depends on use case requirements. To prevent duplication of existing data, you might want to trigger matching when a business user creates or updates a record. Alternatively, you might have existing data that you want to de-duplicate.

- Create a set of rules that determines how the add-on automatically merges data to create a *golden record*. A golden record contains the most accurate and up-to-date data.

The *Administrator Guide* aims to explain how to configure the add-on. The higher-level administrative tasks required to get up and running are described in the *Quick Start* section. Its chapters and sections are ordered in the recommended sequence of completion. The other chapters and sections in this guide drill into more detailed subjects. After you become more familiar with the add-on, feel free to use this guide as a reference for configuration-related tasks.

What's next? Familiarize yourself with the add-on configuration process using the information in . Or, feel free to jump into to begin making your data available to use with the add-on.

Quick Start

CHAPTER **2**

Configuration process map

Task 1: Make data model updates   Task 2: Register tables   Task 3: Set matching options   Task 4: Configure a merge policy   Task 5: Activate matching
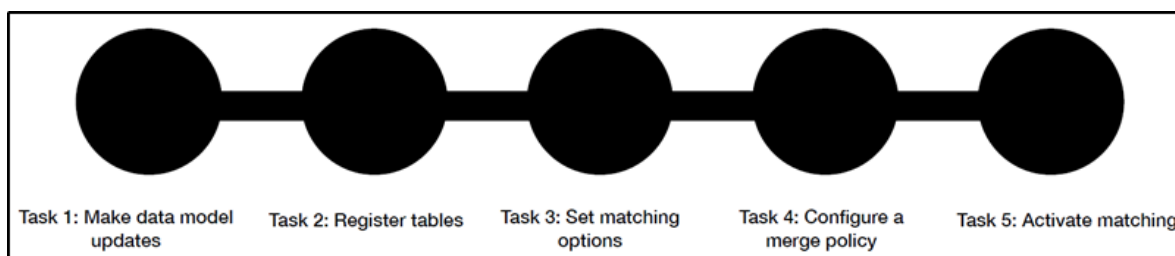
The following table gives you a high-level outline of the above configuration tasks:

1) Make data model updates	Before using add-on features, you must <u>update your data model</u> [p 13].
2) Register tables	You make data available to the add-on by <u>registering tables</u> [p 15]. You can also specify whether to pause add-on functionality on registered tables.
3) Set matching options	You <u>configure matching behavior</u> [p 17] for a registered table by adjusting options that set *how* and *when* matching executes. Matching operations locate potentially duplicate information.
4) Configure a merge policy	You <u>create a merge policy</u> [p 27] to tell the add-on how it should handle matching results. For example, when duplicate information is found, these settings are used to specify which values should be merged, or survived, into the golden record.
5) Activating matching	When you <u>activate matching</u> [p 35] configurations the add-on automatically validates configuration settings and informs you of any errors or warnings.

CHAPTER **3**

Task 1: Updating the model

Before creating a match configuration, you must update and publish the EBX® data model using the Data Model Assistant (DMA). The steps below describe how to include add-on functionality in your data model:

1. From the main menu bar, select the **Data models** panel and select the data model to update.
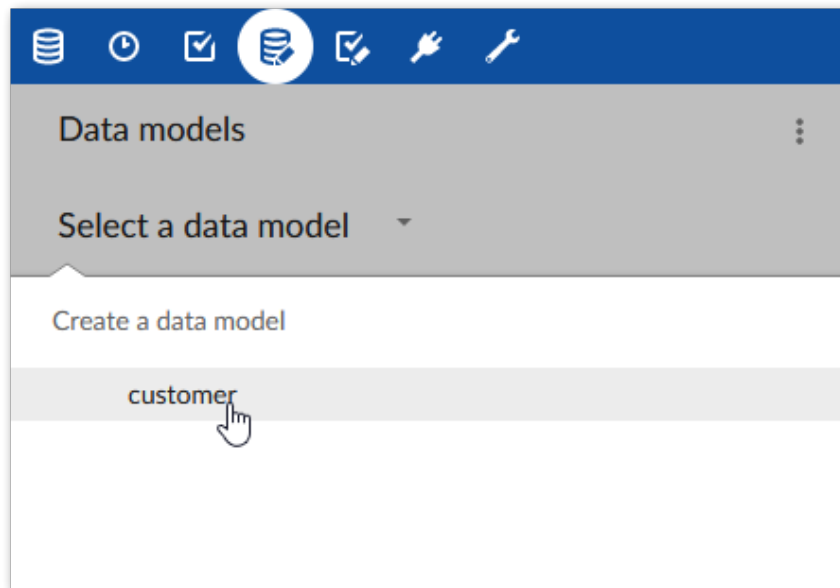
2. Expand **Configuration** and select **Add-ons**.
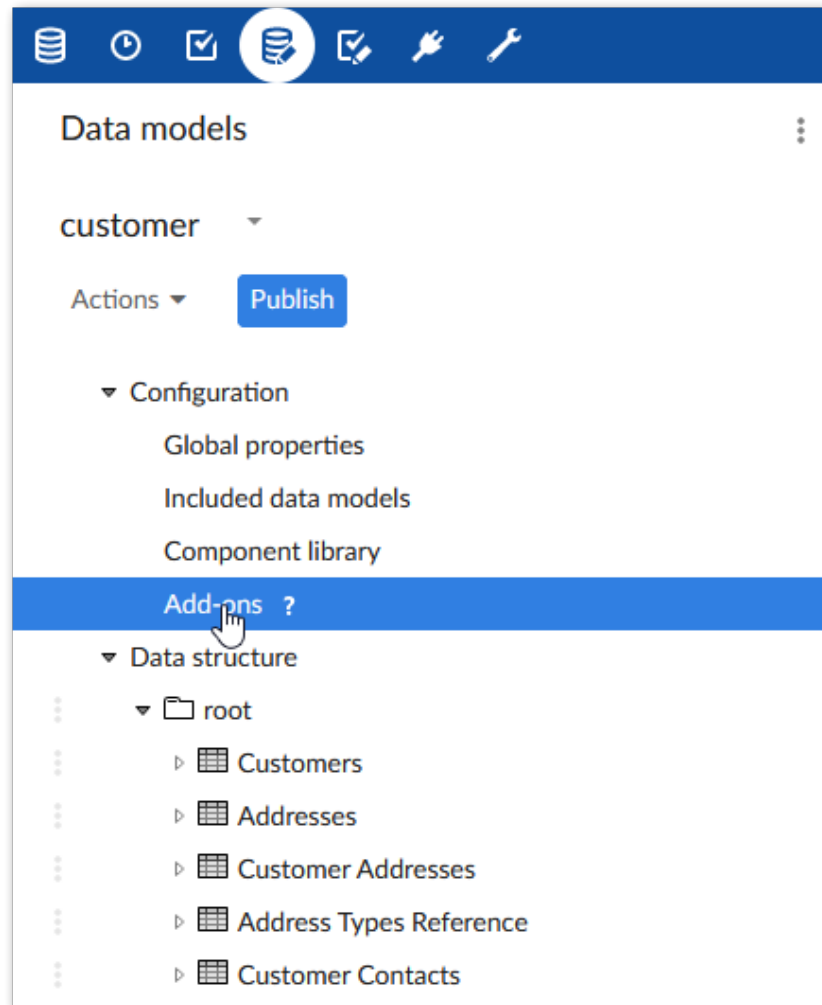
3. Select '+' to add a new entry and choose **Match and Merge**.
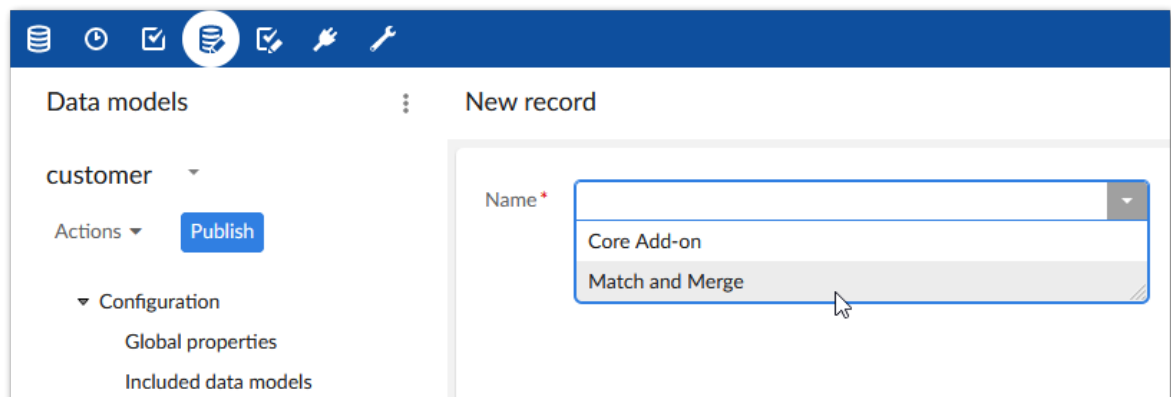
4. After saving, publish the data model.

What's next? Follow the instructions in Task 2: Registering tables [p 15] to enable add-on functionality in a table.

CHAPTER **4**

Task 2: Registering tables

By registering and activating a table, you make its data available for match and merge functionality. To register a table and activate matching:

1. From the main menu bar, select the **Administration** panel and open *Data Quality & Analytics > TIBCO EBX® Match and Merge Add-on > Table activation and settings*.
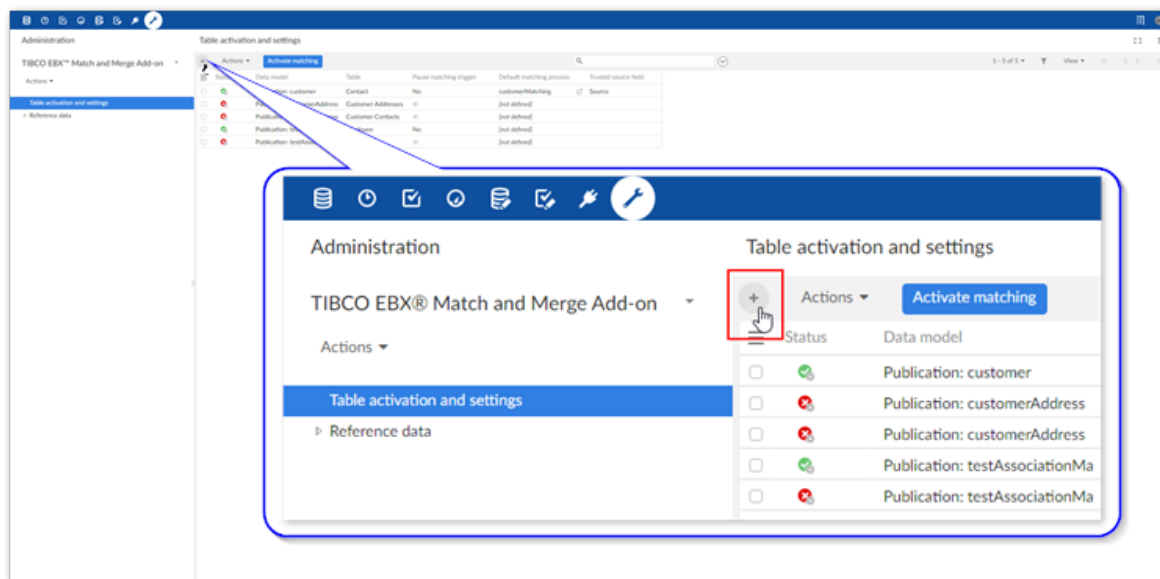
2. As shown above, select the '+' icon to begin a new configuration.

3. Specify the table to register using the **Data model** and **Table** properties.
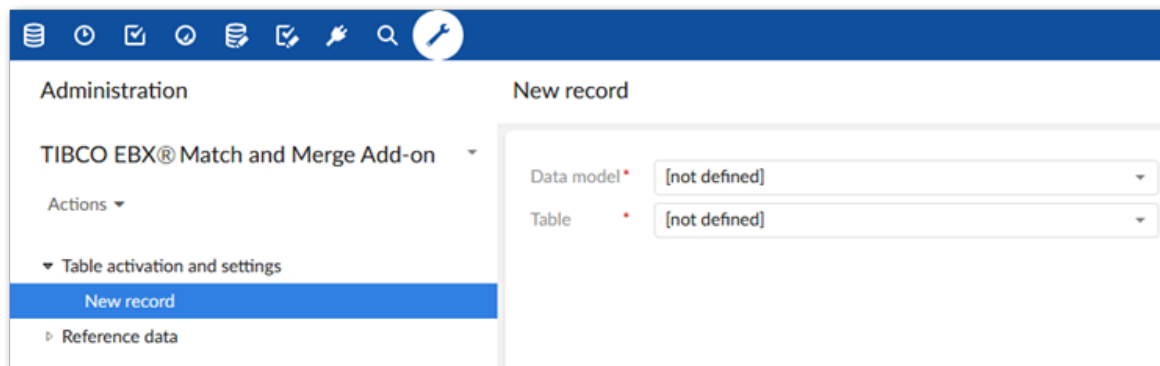
4. Select **Save** and do not close at this time.

Several new tabs display, including: 1) **Matching policies** and 2) **Merge policies**. See the steps in the following sections to configure these options: Task 3: Setting up matching [p 17] and Task 4: Configuring a merge policy [p 27]

If you expand the **Advanced settings** group, you can specify whether users can modify records that were included in a merge operation. See Modifying merged records [p 41] for more information on this option.
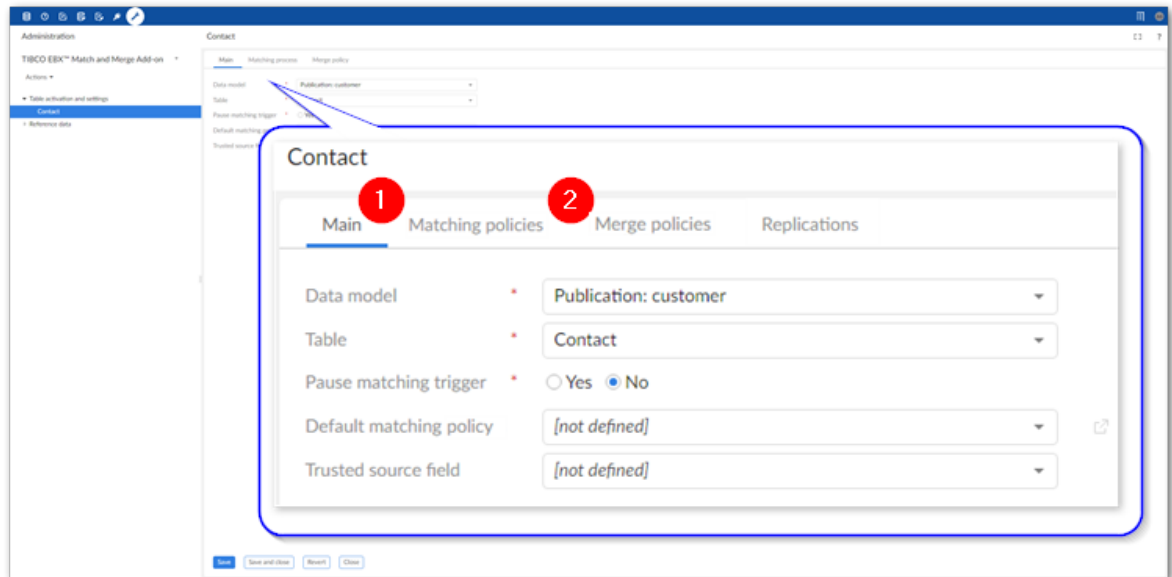
What's next? Follow the instructions in Setting up matching [p 17] to configure a matching policy.

CHAPTER **5**

Task 3: Setting up matching

This chapter contains the following topics:

5.1 Task 3: Overview

A matching operation compares data from two or more records to find potential duplicates. You create a *Matching policy* to tell the add-on:

- *when* to execute a matching operation. You can require users to manually execute a matching operation. Alternatively, a matching operation can run automatically when users create or update records.

- *how* to execute a matching operation. You use a *decision tree* to provide the add-on with criteria that determines whether records are duplicates. You can also apply a merge policy that takes action based on the results of the record evaluation. Please note that creation of a merge policy is covered in Task 4: Configuring a merge policy [p 27].

What's next? Follow the instructions in Task 3a [p 17] to configure a matching policy.

5.2 Task 3a: Creating a matching policy

You create a matching policy under and apply it to a table registered with the add-on. When creating the matching policy you tell the add-on when to execute. Users can run matching manually from a table menu, or you can specify that matching executes inline during record creation or update. When matching executes inline, a matching policy automatically executes after creating or modifying the record. If the matching operation finds potentially duplicate (suspect) records, users are notified and can review and make changes to the record.

To create a matching policy (skip to step 2 if picking up from **Task 1** [p 15]):

1. From the main menu bar, select the **Administration** panel and open *Data Quality & Analytics > TIBCO EBX® Match and Merge Add-on* and open the table settings you want to edit.

2. Select the **Matching policies** tab.
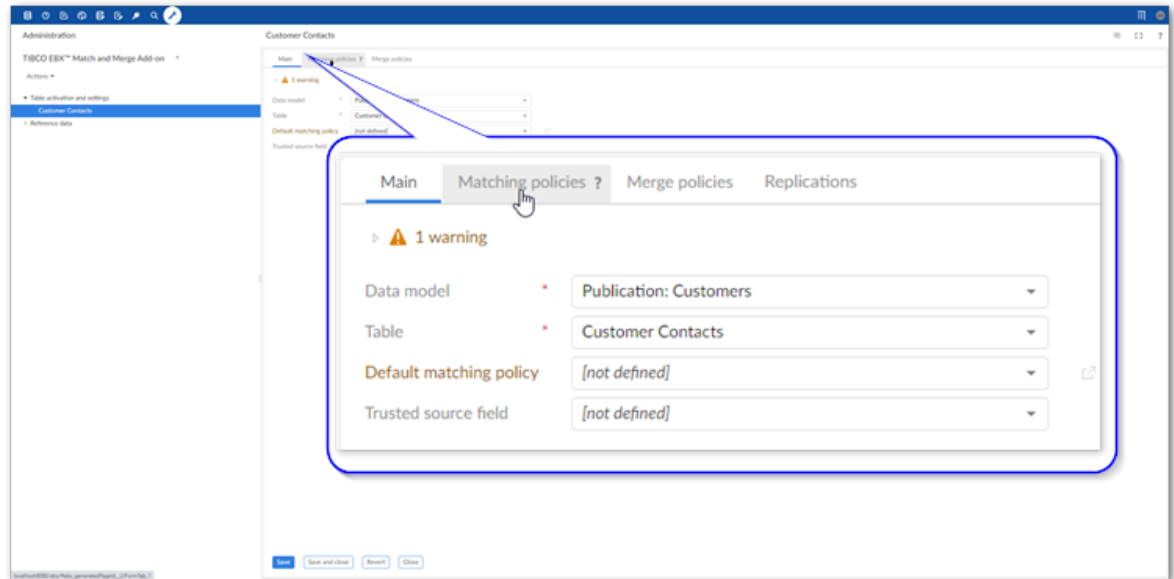
3. Select the '+' icon to create a new matching policy and fill in the required fields on the page that displays. The image below highlights required options:
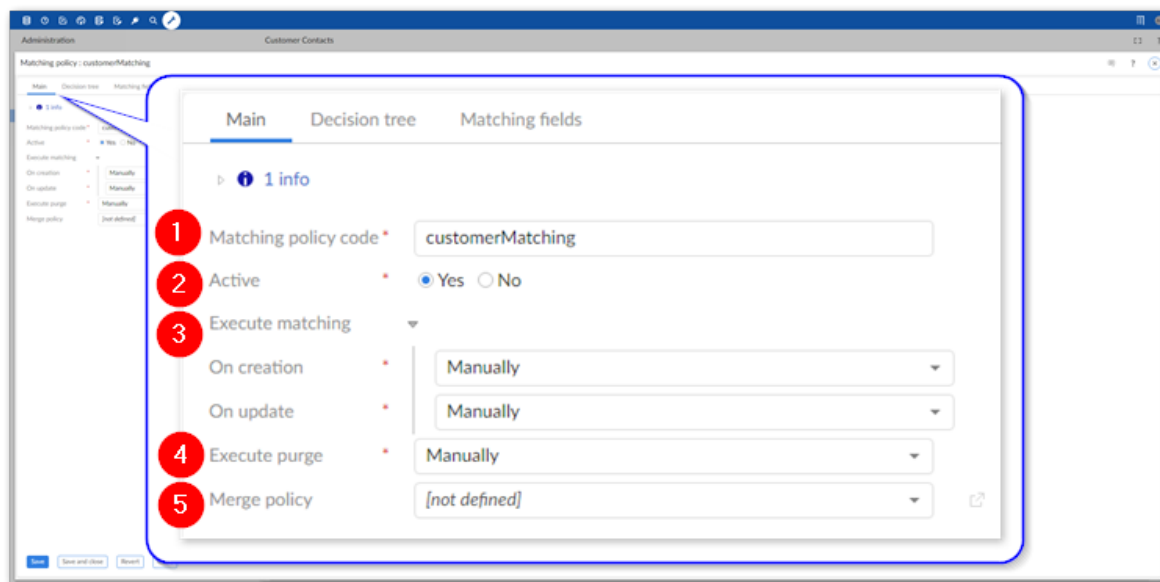
1) Matching policy code	The unique code used to identify this matching policy.
2) Active	Sets whether this matching policy is active.
3) Execute matching	The **Execute matching** group determines how matching executes during creation or update of records. If set to:

- **Manually** the add-on does nothing and users must initiate matching operations. They can do this from a table's **Actions** menu.

- **After submission** the add-on automatically executes matching after creating or updating the record. You can apply this setting when you want to perform bulk creation of records using the EBX® API. There are limitations to this functionality. Please check the *Known issues* section in the release notes for additional details.

- **Before creation** users can search for potential duplicates before creating the record.

- **Inline** the add-on automatically executes matching during the creation/update process. The user must review any potential duplicate records before continuing. When specifying that matching execute inline, the add-on also displays the following menus to select a workflow for record creation or update: **Workflow on creation** and **Workflow**

on update. The selected workflows are launched when a user takes the corresponding action. Note that depending on the assignments in the workflow, the user who created the record that trigged inline matching might not see the screen requiring that the records be reviewed before continuing.

> **Note**
>
> If you set matching execution to **After submission**, **Before creation** or **Inline**, you must set this matching policy as the table's **Default matching policy**. This option is available on the configuration's **Main** tab. Additionally, when a user creates or updates a record, the add-on will only execute automatically if a field configured for matching is modified.

4) Execute purge	Determines whether the add-on automatically removes old matching results. If set to **Automatically** the add-on executes this policy each time matching executes. As this process can be time consuming, you can set it to **Manually** and allow users to execute as needed. They can access this functionality from a table's **Actions** menu by selecting: *Match and Merge > Purge old results*.
5) Merge policy	Tells the add-on which merge policy to link to this matching policy. After creating a merge policy, you must specify it here in order to activate the automatic merge feature. See Task 4: Configuring a merge policy [p 27] for instructions on creating a merge policy.

4. After saving, the following tabs display:

    1. **Decision tree**: Contains settings that determine how the add-on compares data values.

    2. **Matching fields**: Contains configuration settings for each field used in a matching policy.

What's Next? Use the instructions in Task 3b: Creating a decision tree [p 20] to create a matching decision tree to specify *how* this matching policy compares data values.

5.3 **Task 3b: Creating a decision tree**

You create a matching decision tree to tell the add-on which data values to compare and how to compare them. Based on the comparison result you can specify that the add-on set a record's relationship status, or perform additional tests to ensure accurate results. A relationship status identifies records as positive or negative matches, or as suspected of being a match. The add-on creates groups of records identified as positive matches.

To create a decision tree:

1. Select the **Decision tree** tab.
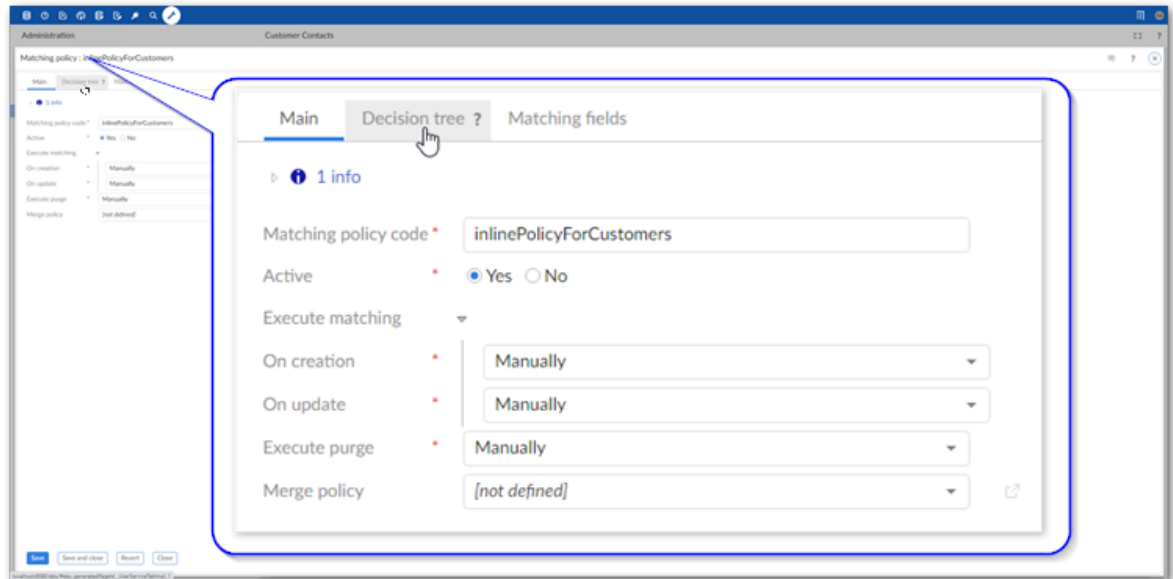
2. As shown below, 1) drag a **Data comparison** test from the **Toolbox** and 2) drag the bottom connection point from the **Start** node and link it with the test.
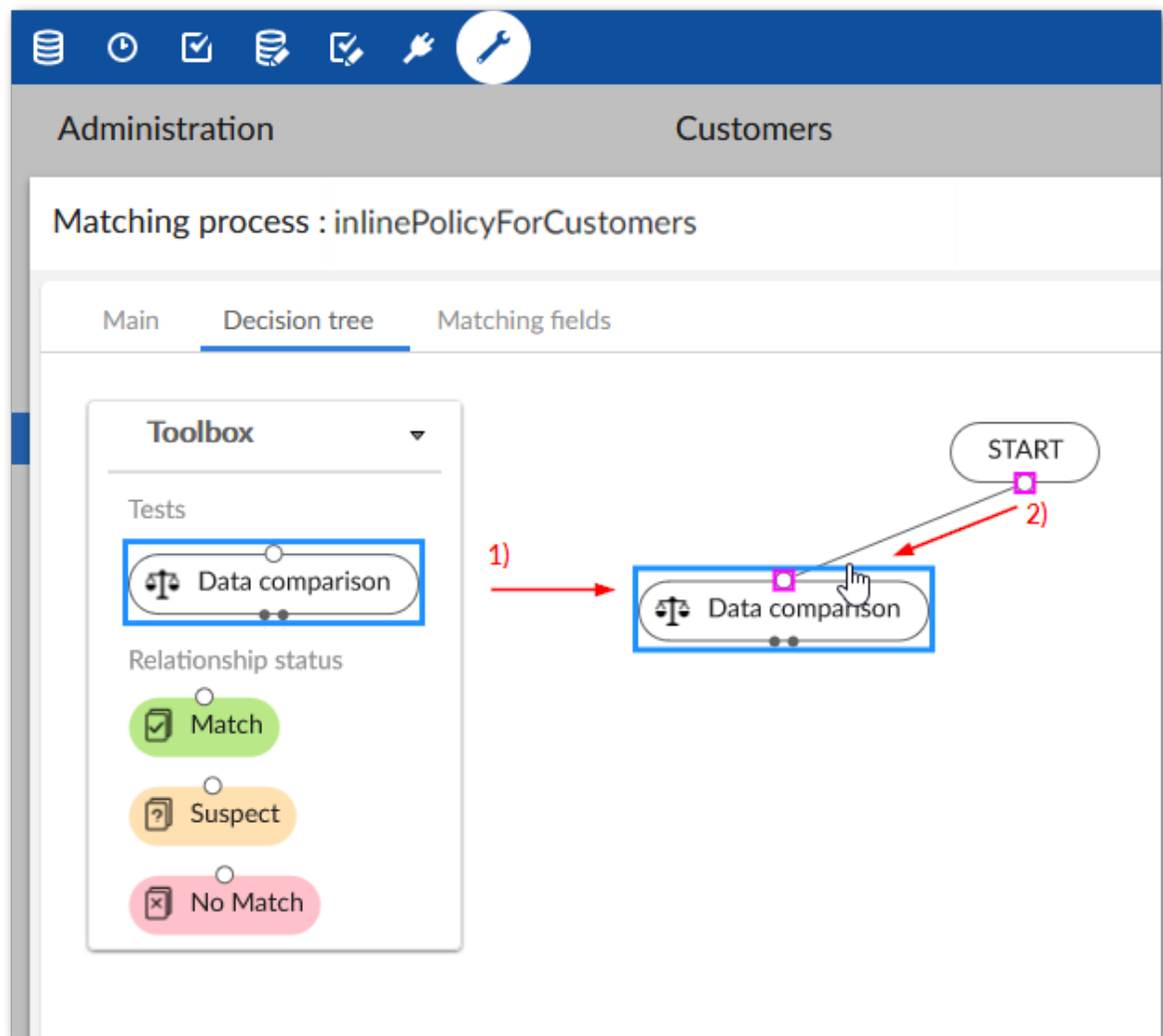
3. Double-click the test to edit its functions:

   1. The **Name** property sets a name for this test that displays on the tree.

   2. **Comparison function**: the add-on uses the selected function to compare the fields you add. When you choose a function from the menu, you can select the preview icon to view a more detailed description of the function. The following functions are available:

      • **All fields match**: For this data comparison to evaluate as true, each included field must meet or exceed its minimum score setting. See the definition of **Minimum score** in step 3 for more information.

      • **At least N fields match**: This option allows you to specify that a certain number of included fields must meet or exceed their minimum score setting for this data comparison to evaluate as true. When you select this option, the **Minimum number of matches** property displays and allows you specify how many fields must match.

      • **Average**: An average score is computed from the score of each included field. If this score meets or exceeds the value set in **Minimum score**, this data comparison evaluates to true. This function also allows you to assign a weight to each field. When using

multiple fields as part of one data comparison, the **Weight** property is used to compute a weighted average and assign a weight to each field. Acceptable values are between 0 and 100. The higher the value, the more weight the field carries. If you set the weight to 0, this field is excluded from the matching pre-processing phase.

> **Note**
>
> To improve matching performance, set the weight to 0 for fields that have a limited number of options. For example, fields that are boolean or enumerations.

Attention

At least one or more fields should have a weight above 0 defined. If all fields are set to 0, matching results are not reliable.

- **No fields match**: For this data comparison to evaluate as true, each included field must fall below its minimum score setting. See the definition of **Minimum score** in step 3 for more information.
- **Predicate**: For this data comparison to evaluate as true, each field must meet the requirements of the defined predicate. The predicate can be any string value. If the field doesn't contain that exact value, it isn't considered a match. In other words, fuzzy matching for the predicate is not supported.

3. **Fields**: One or more fields used in the comparison.

   The **Use relationships** menu allows you to specify whether this field is: a regular field, a relationship through a link table, or a foreign key relationship. Before a related field is available to select, you must configure it in the **Matching fields** tab. See Matching with relationships [p 51] for more information.

   The **Algorithm** property determines which algorithm is used to compute a score for this field. Selecting the **Exact** algorithm also enables the option to specify whether to use case sensitivity on fields with String and Text data types. Note that the list of available algorithms is filtered depending on the search strategy specified for this field. In some cases an algorithm might be available but not recommended for use with a particular search strategy. These algorithms will display in a lighter color font at the bottom of the list. When configuring the algorithm for a field added using the **Matching fields** tab (described below), a link displays to allow you to change the search strategy defined for this field. After changing the strategy, the list of available algorithms automatically updates.

   > **Note**
   >
   > If you select a foreign key field, you should only select the **Exact** algorithm.

   You can also use the **Matching fields** tab to access additional functionality related to fields, including the exclusion of records from a matching operation. See Excluding records [p 39] for more information. Please note that the **Search strategy** option in the **Matching fields** tab must use the same type of strategy (phonetic or distance) defined in the comparison node's **Algorithm** property.

Minimum score: Sets the minimum percentage score required to consider data values from this field as matching. Acceptable values are between 1 and 100. The higher you set this value, the more similar the data values must be to match.

> **Note**
>
> You cannot select multi-value or linked fields in the **Fields** menu.

4. Specify the relationship status if the test evaluates to *true* by dragging one of the following from the **Toolbox** and connecting it with the test's right output point:

   - **Match**: the records will be considered a match.

   - **Suspect**: the records are suspected of matching, but require human review.

   - **No Match**: the records do not match.
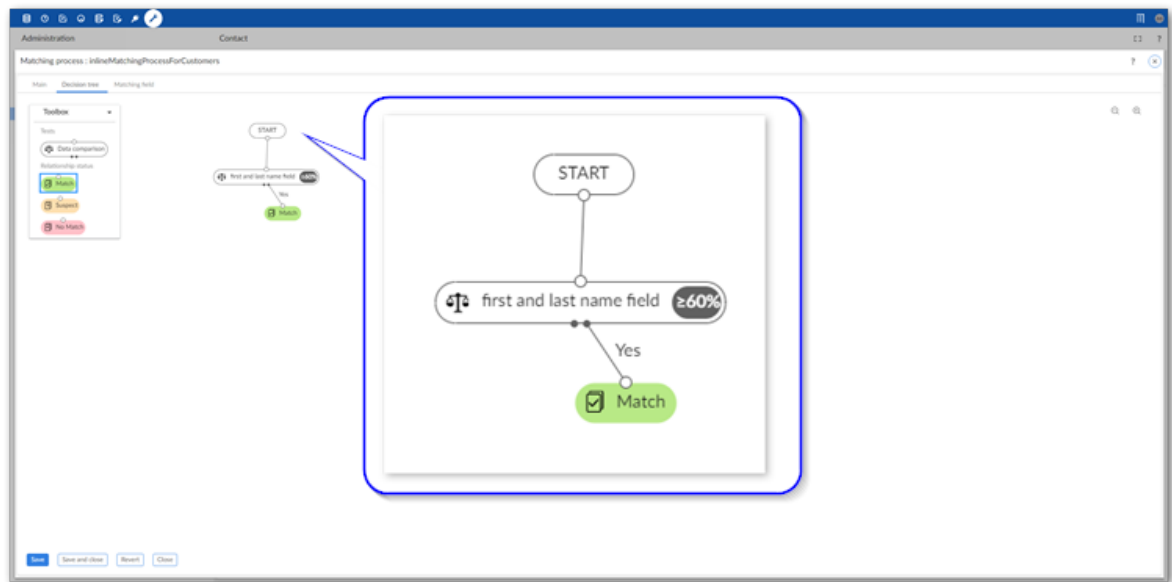
5. Specify the relationship status if the test evaluates to *false* by connecting the desired status to the test's left output point. Note that if you want the result to trigger a new test, add another **Data comparison** node and connect it to desired output.
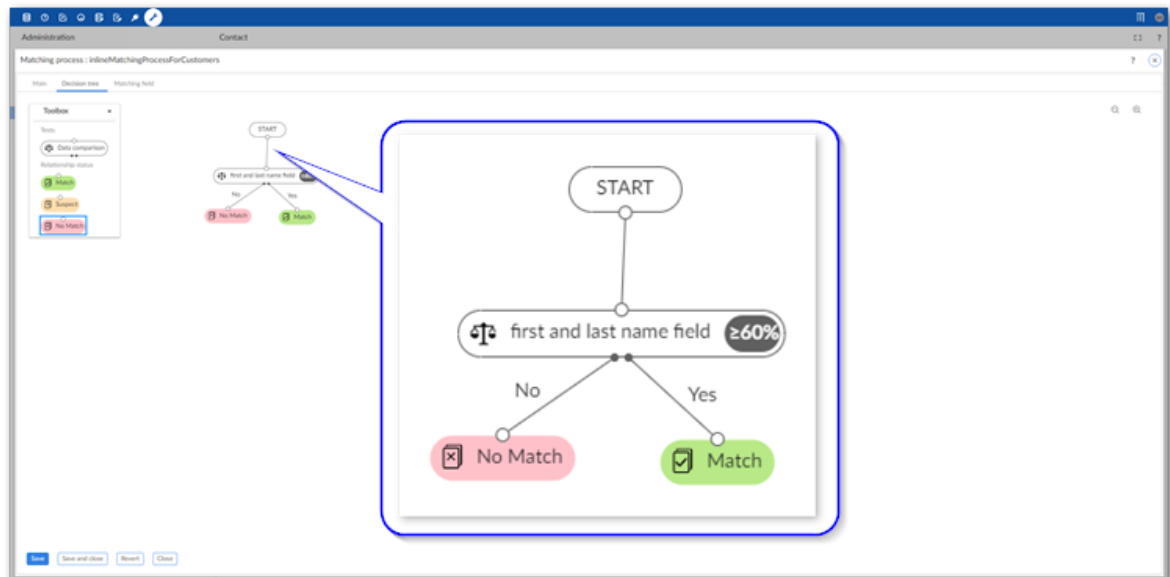
6. Use the process shown in the previous steps to build out your matching tree. Once finished, select **Save and close**.

   The following example shows using three comparisons. If the first comparison evaluates to true, the records are considered suspects from that point on. If the second and third comparisons also evaluate to true, the records are marked as matches.
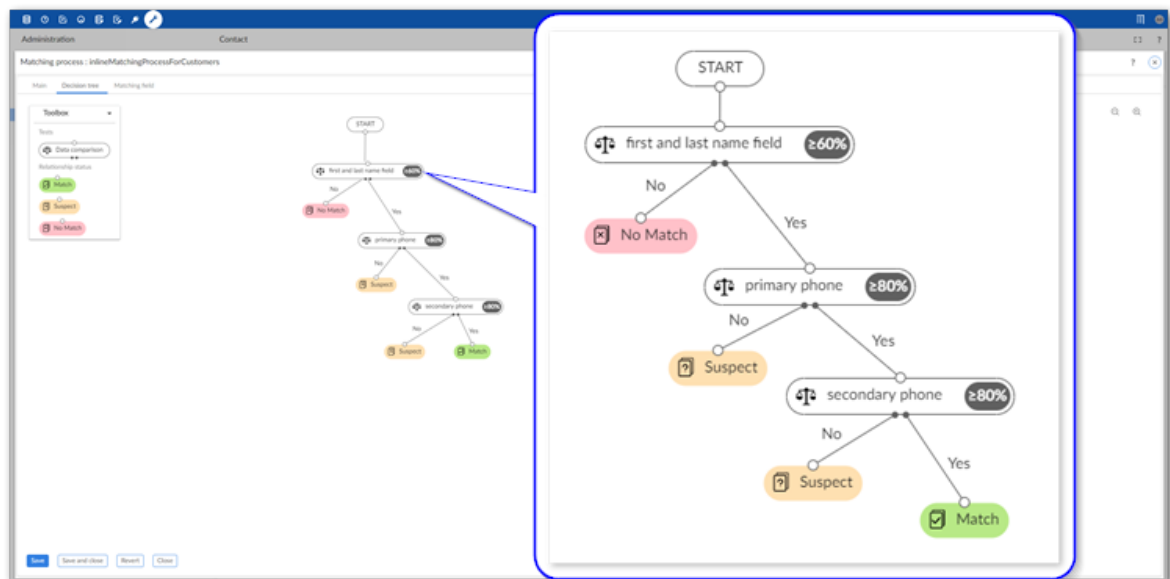
7. If required, set the configuration's default matching policy:

   • From the **Matching policies** tab:

      1. Select the policy you want to set as the default.

      2. From the **Actions** menu, select **Set as default**.

3. Save your changes.

- From the **Main** tab:

    1. In the **Default matching policy menu** select the policy you just created.

    2. Save your changes.

What's next? See <u>Task 4: Configuring a merge policy</u> [p 27] for instructions on setting up the add-on to merge data.

CHAPTER **6**

Task 4: Configuring a merge policy

This chapter contains the following topics:

6.1 Overview

When a matching operation positively identifies duplicate records, or a user manually initiates a merge, the merge policy determines:

- Which record should *survive* the merge process. In other words, the record into which values will be merged to create a golden record.

- The function used to decide which data values will be included in the golden record. One function can be applied to the entire table, or you can set a separate function for each field.

- Whether a new golden record is automatically created. If not, values are merged into an existing record.

- How FK relationships are handled as part of a merge. For example aligning foreign keys to the new golden record by updating the foreign keys in related tables to target the new golden record.

6.2 Creating Merge policy settings

To create a Merge policy (Begin with Step 2 if you completed the previous quick start task):

1. From the main menu bar, select the **Administration** panel and open *Data Quality & Analytics > TIBCO EBX® Match and Merge Add-on* and open the table settings to which you want to add a Merge policy.

2. Select the **Merge policies** tab.
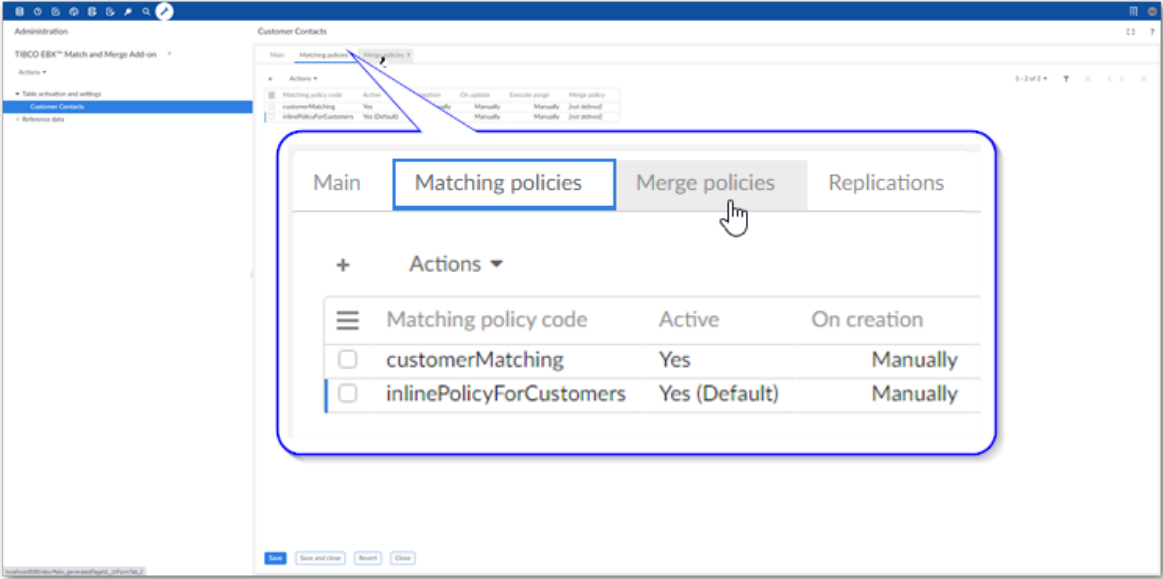
3. Select '+' icon to create a new merge policy and fill in the required fields on the page that displays. The image below highlights these options:

1) Merge policy code	A name used to describe this merge policy.
2) Survivorship strategy	Determines how values are survived into new records.
	Record selection function: Specifies the record selection function to use. A record selection function determines which record survives the merge process. For example, with the **Most trusted source** option, the record stemming from source with the highest level of trust will survive the merge process. Data values will be merged into the surviving record to create a golden record.
	Field merge function: You can use this property to specify a single function that determines which data values are merged into the golden record. If you would like to create settings for different fields, use the options in the **Survivor field** tab to do so.
3) Auto create new golden	Allows you to specify that a new record will be automatically created as part of the merge process. The **Mode** drop-down menu allows you to activate and specify when a new golden is created.
	When this option is enabled, you can use the **Customize source value for new golden** option (not shown) to specify the source to use. The field populates with

trusted sources configured in the following location: *TIBCO EBX® Match and Merge Add-on > Reference data > Trusted sources*.

> **Note**
>
> If the selected source field is a foreign key to another table, its data type must match that of the primary key in the lookup table.

4) Used for manual merge	Sets whether this merge policy's settings are applied when a user manually initiates a merge operation. If multiple merge policies are defined for the same table, you can only enable this option on one policy. When enabled: • Values for the manual merge are pre-selected based on this merge policy's settings. Users can still update selections from the merge screen. • The **Apply permission on merge view** field displays. This option allows you to set whether TIBCO EBX® permissions apply in the **Merge view**. This allows you to address the use case where a data steward needs to access all fields in order to perform a successful merge.

6.3 **Configuring survivorship fields**

Merge policies allow you to define criteria for entire tables, but also allow you to set parameters for individual fields. To set merge behavior for a specific field:

1. Select a merge policy's **Survivor field** tab and use the '+' icon to set up a new field.

2. The following list highlights some of the options:

    1. **Field**: Choose the field on which these settings apply.

    2. **Field merge function**: Select the function used to choose which value should survive the merge process. If you choose **Constant**, you can enter a value that the add-on will populate this field with (in all golden records). Note that this option is only available for fields of String and Text data types and *not* for multi-valued fields.

    3. **Condition for field value survivorship**: Optionally, enter a predicate expression to evaluate the value to be survived. Both the **Field merge function** and this expression must evaluate to true to be survived. The XPath predicate expression used here must comply with EBX® requirements. For more information, see the *Supported XPath syntax* section *EBX® Reference Guide*.

    4. **Execute only if empty**: Set whether the merge only happens when the value in the survived record is empty.

6.4 **Managing relationships**

The add-on allows you to set behavior for cases when a merge operation involves relationships. For example, you might want to update a foreign key that points to the value in a merged record to point to the value in the golden record. To manage how the add-on handles relationships in these types of scenarios:

> **Attention**
>
> The following steps apply to related tables contained in the same dataset. If the related table is in a different dataset or dataspace, the add-on will automatically align the foreign keys.

1. Select a merge policy's **Related tables** tab.

   The add-on automatically adds options here for related tables that come from the same data model and hold a foreign key to the table where matching executes. Note that if the data model was updated, you can refresh the list of related tables. To do this, open the **Actions** menu and select **Refresh**.

2. Double-click to open the relationship you want to update.

3. Use the **Align foreign keys** drop-down menu to set how FKs should be managed.

   You can choose from the following options:

   - **None**: This relationship is ignored.

   - **Manually**: The merge process will allow users to manually select whether the related value is updated.

   - **Automatically**: The FK value is updated automatically based on the survived record.

6.5 **Enabling automatic merge**

If you want the add-on to automatically merge records that it considers positive matches, you must associate a merge policy with a matching policy. Follow the instructions below to edit a matching policy and associate a merge policy:

1. Select a registered table's **Matching policies** tab and open the desired matching policy.
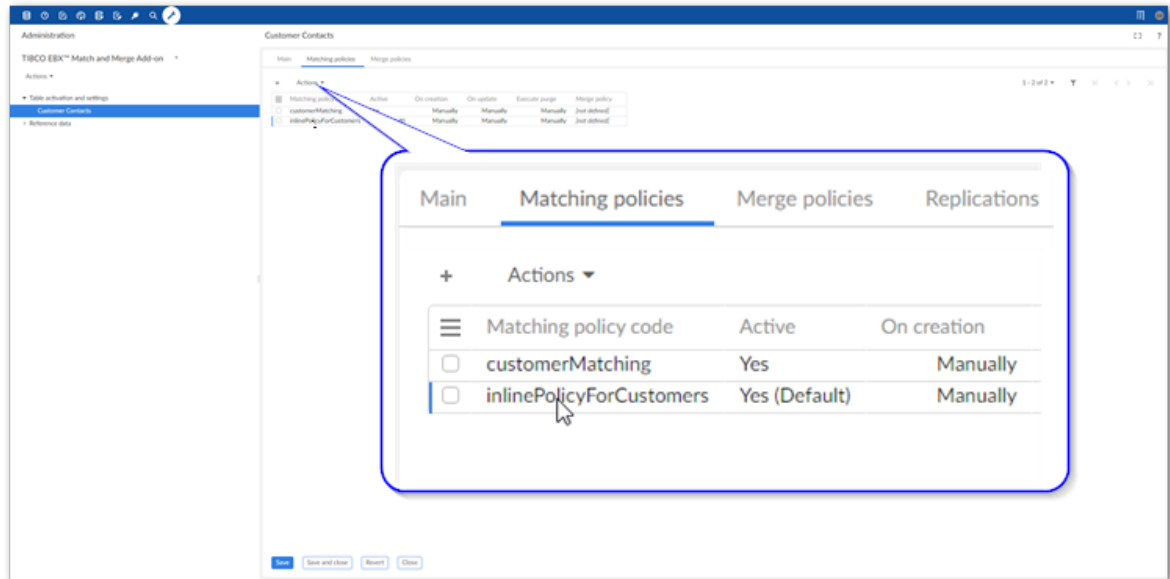
2. On the **Main** tab, use the **Merge policy** menu to select the desired merge policy.

> **Note**
>
> When you select a merge policy here, merging happens automatically and happens according to the settings in the selected policy. When a merge is triggered, it bypasses the option to merge manually.
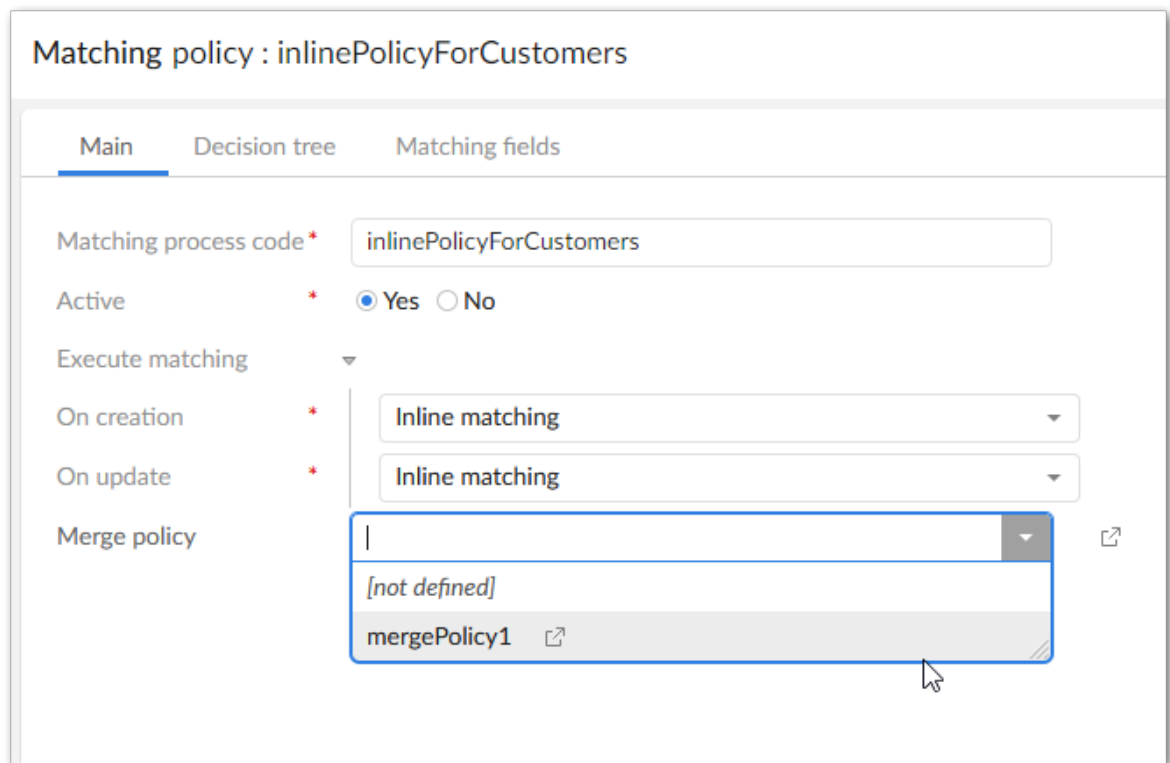
3. Save and close.

What's next? To enable automatic matching features for your data model, see the instructions in Task 5: Activating matching [p 35].

CHAPTER **7**

Task 5: Activating matching

The final step to setting up a configuration is to active matching on the table configuration. You only need to perform this step once per table configuration. As a best practice you can wait until creating several table configurations and enable them all at once.

> **Note**
>
> If matching is enabled for multiple tables from the same data model and you need to re-activate one table, you must select and activate all of the configurations from the same model.

To enable matching:

1. Navigate to **Table activation and settings** and select each configuration you want to activate. As a shortcut to activating all configurations, do not select any records.

2. Select **Activate matching** at the top of the screen.

   Part of this operation includes validating the matching configuration. You will see a validation page with any errors or warnings that you might want to address prior to using the matching configuration.
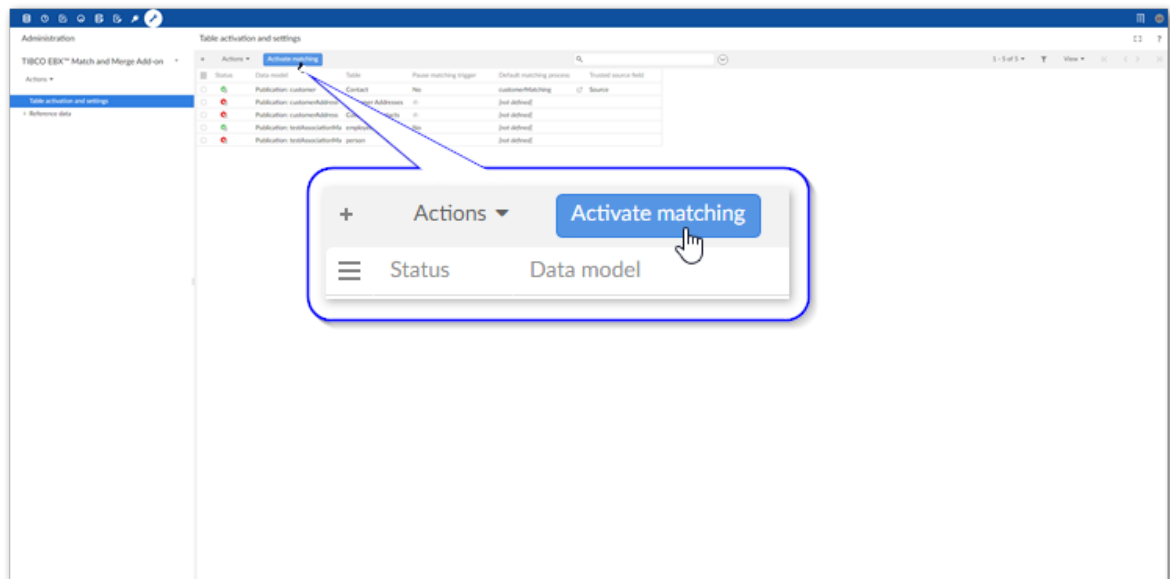
After activating matching you can use the add-on's **Evaluate matching** service. This service allows you to run a match on selected record and review the results to ensure they are consistent with your expectations. See for more information.

CHAPTER **8**

Creating matching views

This chapter contains the following topics:

1. Overview
2. Creating a tab that shows merged records
3. Creating a hierarchical view of golden records

8.1 Overview

You can create matching focused views that include:

- Adding a tab to a golden record's details view that shows all source (merged) records of the golden.

- Creating a hierarchical view where parent and child nodes are golden and merged records, respectively.

8.2 Creating a tab that shows merged records

Use the following steps to add an association to your data model using an XPath predicate:

1. Select **Data models** from the main EBX® toolbar.

2. Select the desired data model.

3. Using the ellipsis menu next to the desired table, select **Create child**.

4. Fill in the **Name** field and for the **Kind of element** option, select **Association**.

5. In the **Definition** box, select the following:

   - **By XPath predicate**

   - **Target table**: Select the current table

   - **XPath expression**: Enter the following `./Linked_Field_GoldenId = ${../Linked_Field_FunctionalId} and osd:is-not-null(./Linked_Field_GoldenId)`

6. Select **Create**.

7. Publish your data model.

   After successful publication, the details view for golden records is updated with a **Main** tab and an association tab that displays merged records.

8.3 **Creating a hierarchical view of golden records**

You can create a hierarchical view with golden records as parents and corresponding merged records as children.

To create the hierarchy view:

1. Open the table you want to create the view of.

2. From the **View** menu, select **Create a new view**.

3. Define desired options, but for the **View mode**, select **Hierarchical view**.

4. Select **Next**.

5. In the **Dimension** box, select the **Target record** option.

6. Select **Create** to create the view, or **Create and apply** to create and display the view.

   This view is now accessible from the **View** menu.

CHAPTER **9**

Excluding records

One way that you can boost matching performance is by excluding records from matching operations. When the total number of records the add-on must process is reduced, it can cut down on the total processing time and help improve the overall quality of results by preventing false positive matches. When you configure options to exclude records:

- You tell the add-on to look at the values in a specific field.

- You specify criterion that determines whether (based on the field value) a record is excluded. For example, you could specify that null values in an Expiration Date field exclude records from matching. The reason being, only products that still have an expiration date are not yet expired and should be matched.

- The configuration applies to all available types of matching.

- Any record that is excluded from matching is set to `<Unset>`.

To exclude records from matching:

1. Navigate to *Administration > TIBCO EBX® Match and Merge Add-on > Table activation and settings* and open the configuration where you want to exclude records.

2. Select the **Matching policies** tab and open the desired matching policy.

3. Select the **Matching fields** tab and create a new, or open an existing field.

4. Use the **Exclude records where** property to add at least one criteria used to exclude records.

   You can select:

   - **equals** and **is different** to enter a specific value that must be the same or different to exclude a record, respectively.

   - **is null** or **is not null** to specify that if the field is missing a value or has a value the record will be excluded, respectively.

5. Save your changes before exiting.

CHAPTER **10**

Modifying merged records

Some use cases require a record's life-cycle to continue after a merge operation where it did not become the golden record. The add-on allows you to enable merged record modification for these types of cases.

To enable merged record modification:

1. Navigate to *Administration > TIBCO EBX® Match and Merge Add-on > Table activation and settings* and open the configuration where you want to enable merged record modification.

2. Expand the **Advanced settings** group on the **Main** tab.

3. Set **Modify merged records** to **Yes**.

   The following behavior is applied when a record is in the merged state and the modified field:

   - Is *not* configured to use in matching operations: The add-on executes an automatic merge using the settings defined in the default matching policy's merge policy. This ensures the golden record contains the most up-to-date data.

   - Is configured to use in matching operations: The record is changed to the <unset> state and should be matched again against other records in the table.
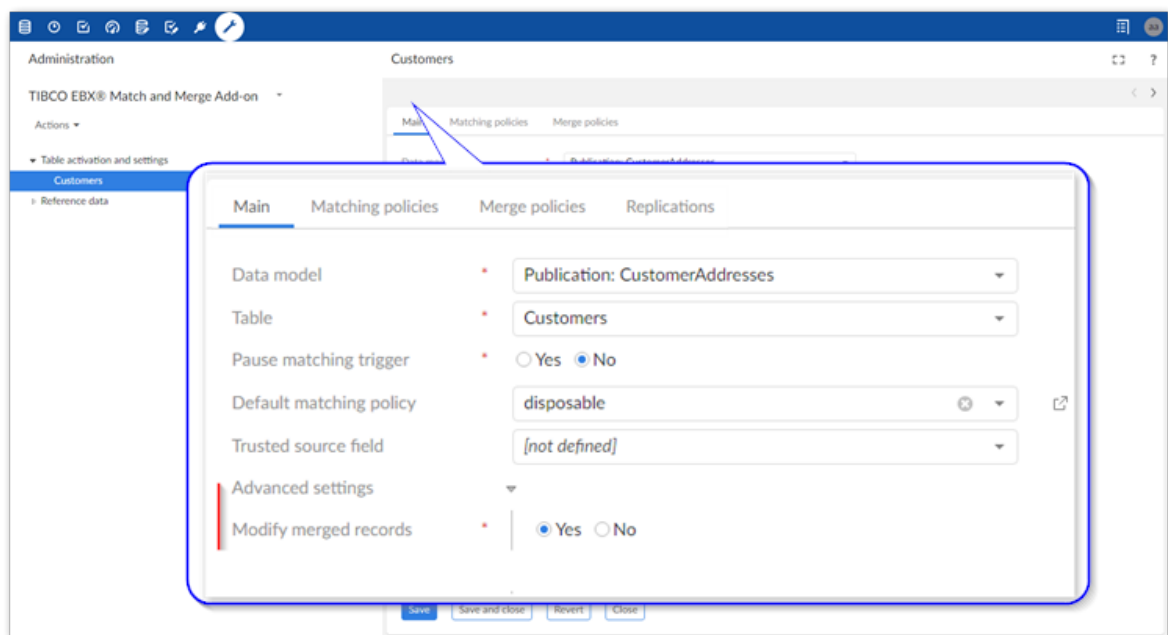
4. Save before exiting.

   Users are now able to save changes to merged records.

CHAPTER **11**

Validating matching configurations

Incomplete or invalid matching configurations can lead to tainted results and poor performance. To help prevent these issues, the add-on can check configurations and provide a report that highlights any errors or warnings.

> **Note**
>
> Just because a validation report turns up issues, it will not prevent the add-on from executing an enabled and active matching policy. If the report turns up serious issues, you might want to pause the matching configuration until you resolve them.

To validate a matching configuration:

1. Navigate to **Table activation and settings**.

2. Select one or more configuration entries.

3. Run the validation:

   - From the **Actions** menu, select **Validate configuration**.

   - Select **Activate matching** at the top of the screen. Note that this option will enable any configuration instances that are not already enabled. However, running this on a configuration that is enabled has no effect other than validating the configuration.

CHAPTER **12**

Setting trusted sources

This chapter contains the following topics:

1. Overview
2. Adding and prioritizing sources
3. Data steward decisions as sources

12.1 Overview

When you have records that stem from multiple systems, you might trust values from some sources over others. For example, take three source systems: HR, Accounting, and IT. If you had a question about an employee's job title or role, which system would provide the most accurate information? You might even determine that a different source is more reliable for each field. In these situations you can use the add-on's trusted sources feature to specify which source provides the most reliable information for a given field.

The add-on ships with a built-in source called **Data Steward decision**. This source enables users to override the order of trusted sources on a per field basis. See Data steward decisions as sources [p 49].

12.2 Adding and prioritizing sources

You can create multiple sources within the add-on and define a prioritized list for tables and fields that are included in a merge policy. For this process to work, you need a field in your record that specifies its source. When you use the steps below to add sources, ensure that the sources you add correspond to the allowable values for the field. For example, if a System field had the allowable values of HR, ACCT, and IT, you would create the following sources: HR, ACCT, IT.

To add sources and set their priority:

1. Create the sources:
    1. Open the **Administration** panel and navigate to *TIBCO EBX® Match and Merge Add-on > Reference data > Trusted sources > Sources*.

2. Create a new record.
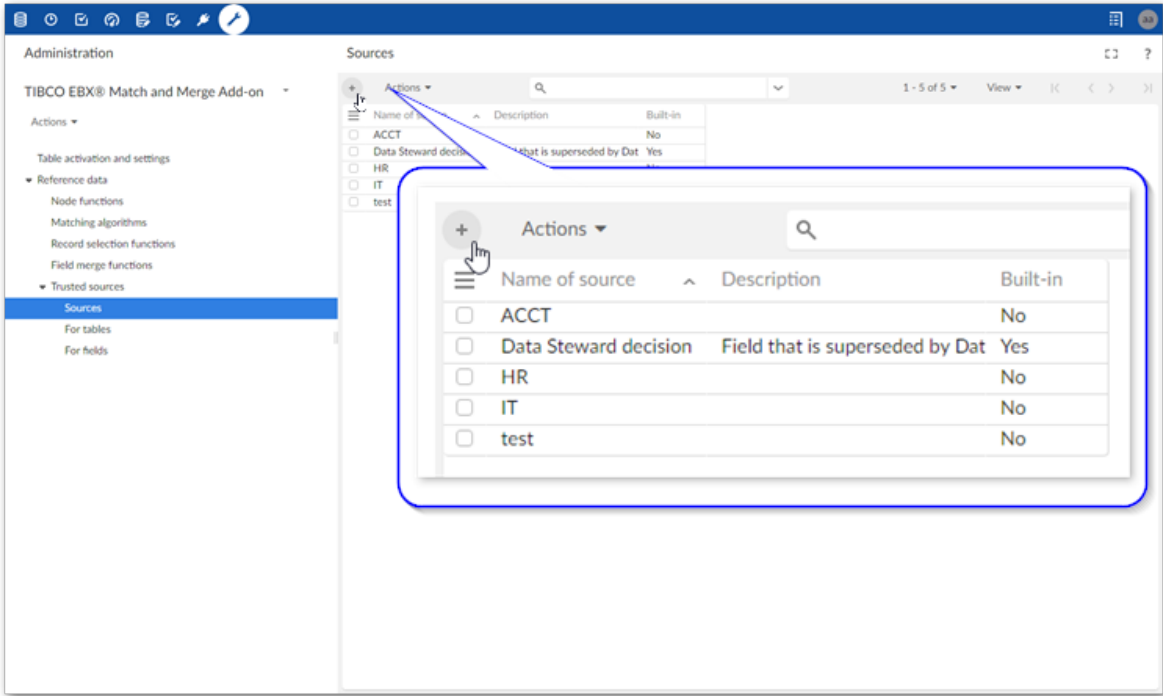
3. Fill in the required name and optional description.

> **Attention**
>
> Source names must be identical to the corresponding values that identify sources in records. For example, if a System field identified its originating source using HR, ACCT, or IT, you must provide these same values for each source name.
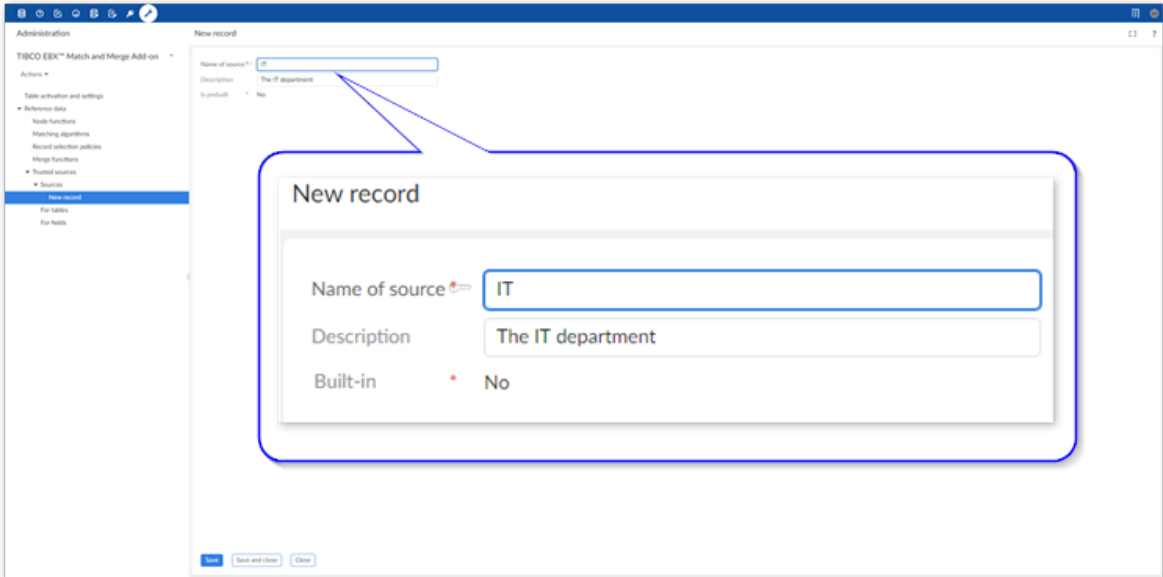
4. Save and close the record.

2. Set and prioritize sources for fields or tables:

    1. Select either **For tables** or **For fields** depending on whether you want to set sources at the table or field level.

    2. Create a new record. This example uses fields.

    3. Select the desired field or table and create the list of its trusted sources as described below:
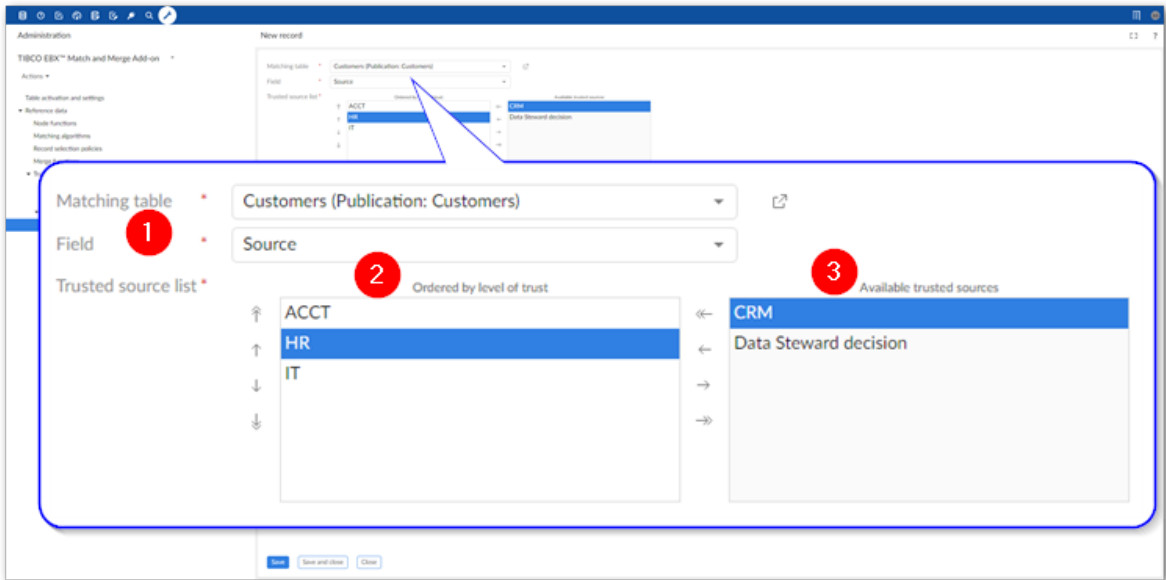
1) Matching table and Field menus	Specify the table or field that this list of trusted sources applies to. Note that only fields of the String and Text data types are available to select from the menu.
2) Trusted source list: Ordered by level of trust	This list contains all of the sources applied to the selected table or field. Prioritize the sources using the up and down arrows. The source at the top has the highest priority.
3) Trusted source list: Available trusted sources	This list contains all sources available to apply to the selected table or field. Use the right and left arrows to move sources between the lists.

3. Update the matching table configuration to use the correct source field:

1. Open the desired table configuration by selecting the **Preview** button as shown below from the trusted sources screen, or do so from **Table activation and settings**.

> **Note**
>
> Before navigating away from this screen save your changes.
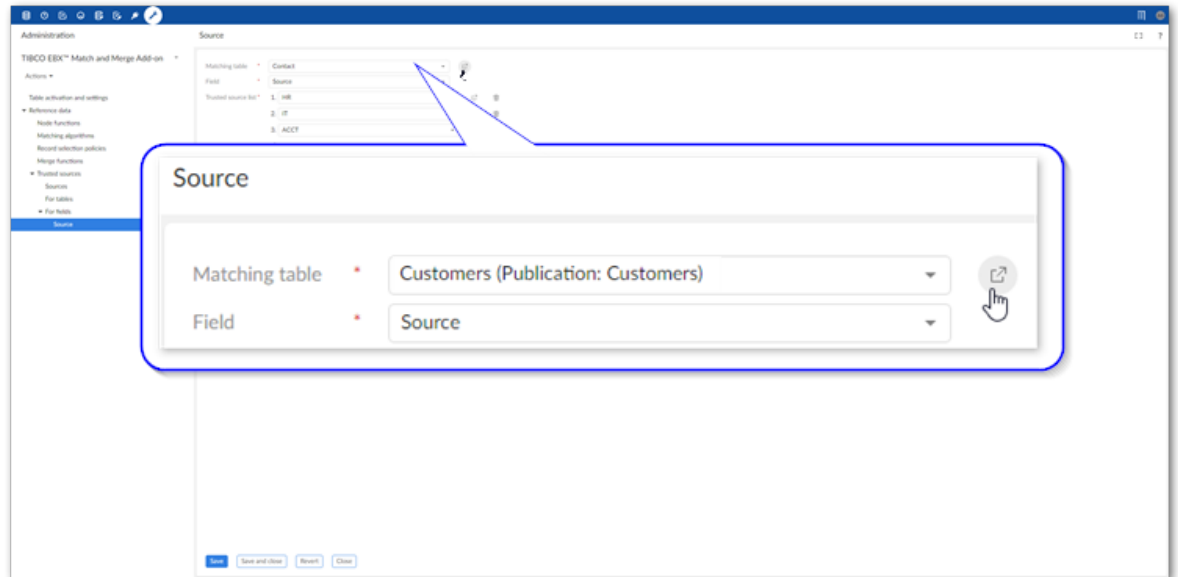
2. On the **Main** tab, select the field indicating the source from the **Trusted source field** menu:
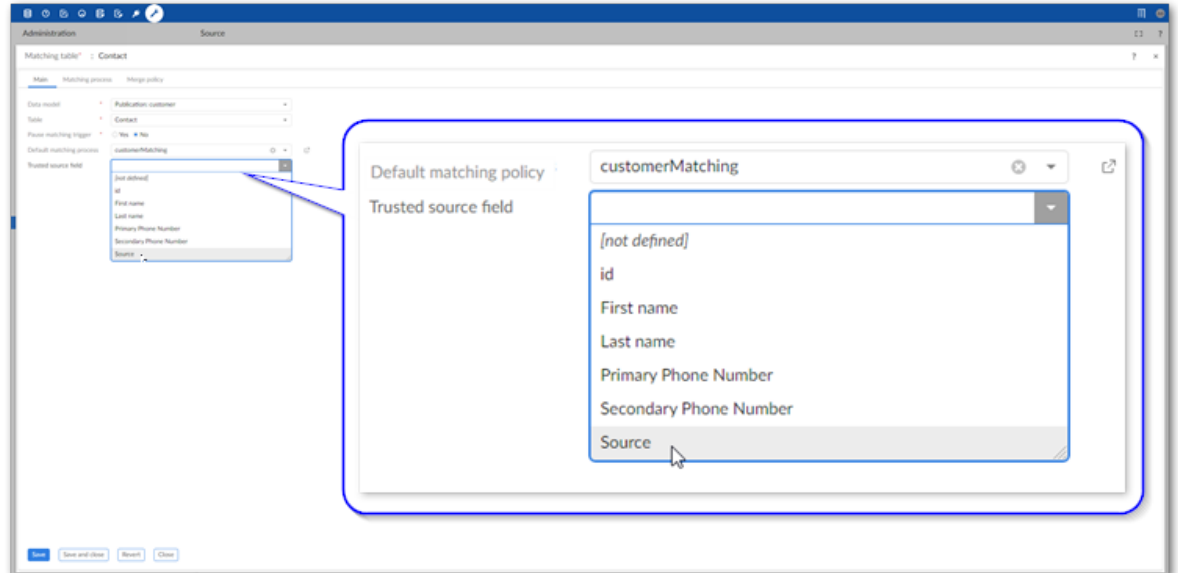
3. Save and close the record.

Now when you configure merge policies for this table and choose the **Most trusted source** option, the add-on will prioritize survived values according to the trusted source list you just created. Note that when selecting the values to survive, the add-on prioritizes those configured at the field level first and the table level second.

12.3 **Data steward decisions as sources**

You can enable user decisions to override the order of trusted sources for future merge operations by adding a **Data Steward decision** to a trusted source list. When a user runs the **Supersede survivorship** **[p 121]** service they can override a field merged as the result of adhering to a trusted source list. How the add-on handles the same field in future merges depends on the position of the **Data Steward decision** source in the trust list.

> **Attention**
>
> If a merge policy uses the **Customize source value for new golden** option to use a specific value, users cannot change source field values. This would create a direct conflict with the merge policy. If you want to allow users to override a merge decision, set the **Customize source value for new golden** to **[not defined]**.

When a user overrides a merge decision and the **Data steward decision**:

- is at the top of the list, the add-on "learns" and uses the value obtained from the same source the next time that record is involved in a merge operation.

- is *not* at the top of the list, the add-on will use values from sources with a higher trust level over the one chosen by the user.

- has trusted sources below it in the list, the add-on uses the value from the user decision in the future over those below it.

The following example includes two fields with values from different sources. Notice the result when the add-on applies different trusted source lists to an auto merge performed *after* a user executes the **Supersede survivorship** service:

		Color:	Number:	Source:
	Record 1	Red	One	CRM
	Record 2	Green	Two	ERP
		Merge results		
If the trusted source order is set to: 1) CRM 2) Data Steward decision 3) ERP	Initial auto merge	Data steward decision	Next auto merge
		Red	Green	Red
		One	One	One
If the trusted source order is set to: 1) Data Steward decision 2) CRM 3) ERP	Initial auto merge	Data steward decision	Next auto merge
		Red	Green	Green
		One	One	One

CHAPTER **13**

Matching with relationships

The add-on allows you to perform matching operations on fields that derive their value from related tables through direct foreign key relationships and join tables. You can configure these fields in the decision tree the same as other fields, with one prerequisite: you must create a configuration for the field on a matching policy's **Matching field** tab. The field is not available to select in a decision tree node until you complete this step.

The following steps outline how to configure this option:

1. Create a matching field:

   1. Select the **Administration** tab and navigate to *TIBCO EBX® Match and Merge Add-on > Table activation and settings*.

   2. Open the desired table and select its **Matching policies** tab.
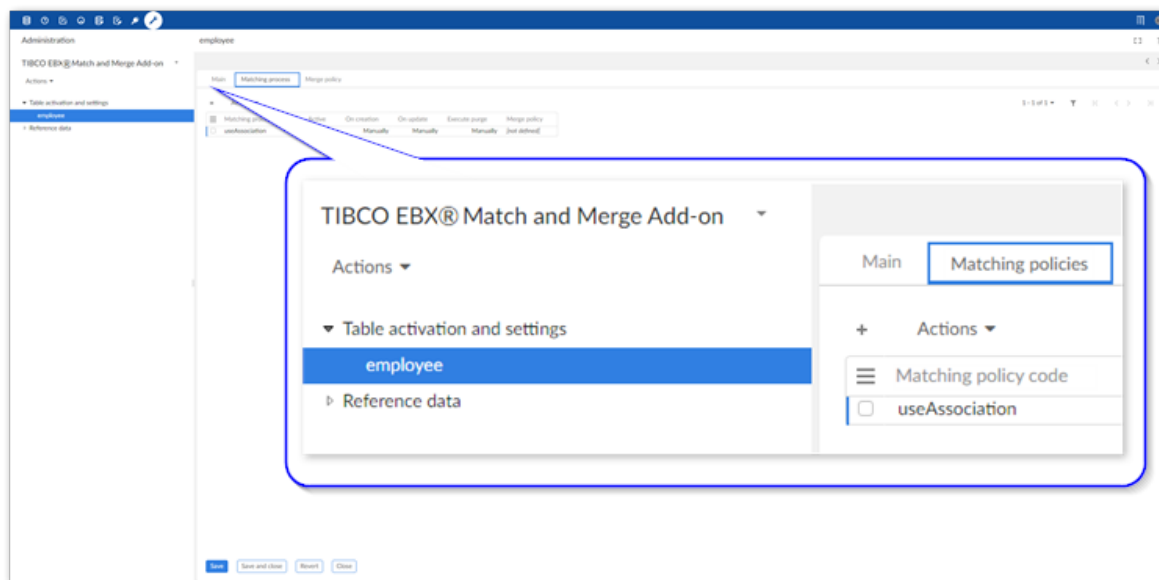
3. Open the desired matching policy and select the **Matching fields** tab.
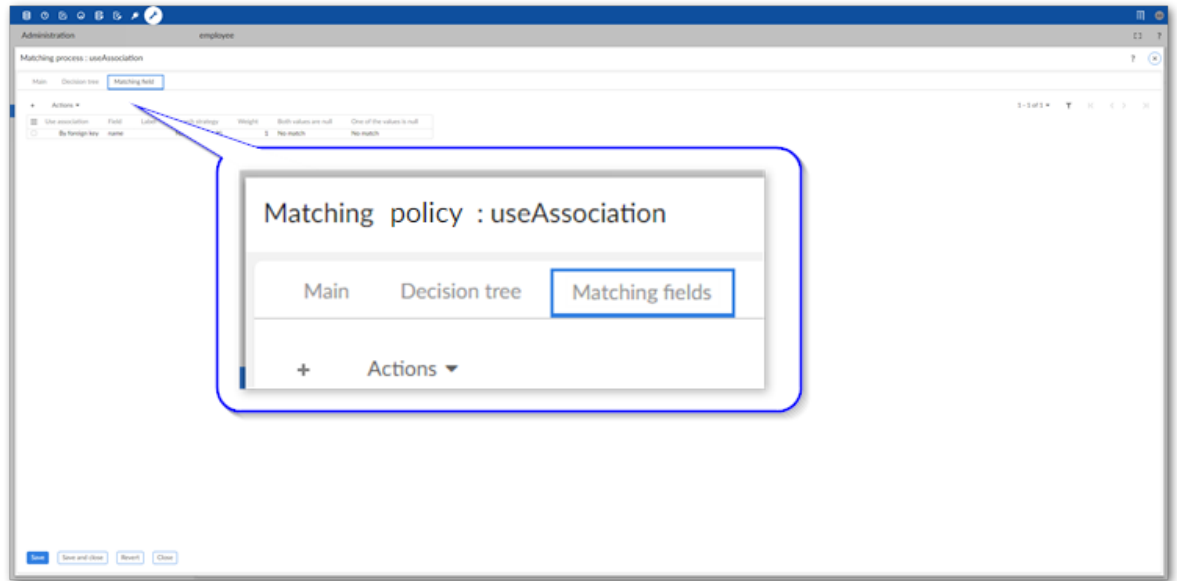
4. Create a new record:

The image below and following table highlight some of the options:

| Use relationships | Choose **By foreign key** or **Over a link table**.

Note

When selecting the **By foreign key** option, the you can only specify a single hop foreign key. In other words, it must point to a regular field and not another foreign key. |
| --- | --- |
| Location | Select the option that corresponds to the related table's location. Note that the current release only supports the **Same dataset** option. |
| Matching table foreign key and Related table foreign key | **Matching table foreign key:** Select the wand icon and choose the foreign key you want to match through. When matching using a link table, this is the relationship between the table where matching is configured and the link table.

Related table foreign key: This option is only available when the **Use relationships** property is set to **Over a link table**. Select the wand icon and choose the relationship between the link table and table containing the source value. |

Field	Select the field in the related table where you want to apply matching.
Exclude records where	Specifies values to exclude records from matching operations. See <u>Excluding records</u> [p 39] for more information.
Pre-processing	Some parts of the matching process are handled by the core EBX® engine and others by the add-on. Pre-processing refers to the processing done by EBX® to find similar records before the add-on matches them. See <u>Matching processing overview</u> [p 57] for more information. **Search strategy** and **Weight**: Select the search strategy to use and provide a weight. Available search strategies are defined at the data model level. The weight sets the level of importance placed on this field as compared with others. The higher the value, the more weight the field has.
Null value management	Use the options in this group to determine how null values are handled during a matching operation. **Attention** When enabling options to handle null values, you should set the field's **Weight** to a lower value relative to other fields. Not doing so can result in unexpected matching results.

2. Open the desired decision tree and double-click to edit a data comparison node.

3. Configure the node to use the foreign key field by adding it to the node's **Fields**:

- Set **Use relationships** to **By foreign key** or **Over a link table**.

- Use the **Field** menu to select the foreign key field configured in the **Matching fields** tab.

> **Attention**
> At least one of the fields used in the decision tree must be located in the table where matching is configured and active.

- Define the **Algorithm** and **Weight** options as needed.

4. Save and close the **Data comparison node** configuration.
5. Save and close on the **Decision tree** tab.

CHAPTER **14**

Matching processing overview

The add-on and the EBX® core share responsibility for matching process execution. The core product pre-processes the list of records to match to find similar records and passes them to the add-on. EBX® finds similar records by taking the base record, or record to match against other records, and applying certain parameter settings.

When input consists of a selection of records, the output is a *group* of similar records. If the input is an entire table, additional processing is performed and the output is *pairs* of similar records. The add-on processes the output from EBX® using the logic defined in a matching policy's decision tree to determine whether values match. The diagram below visualizes this process:
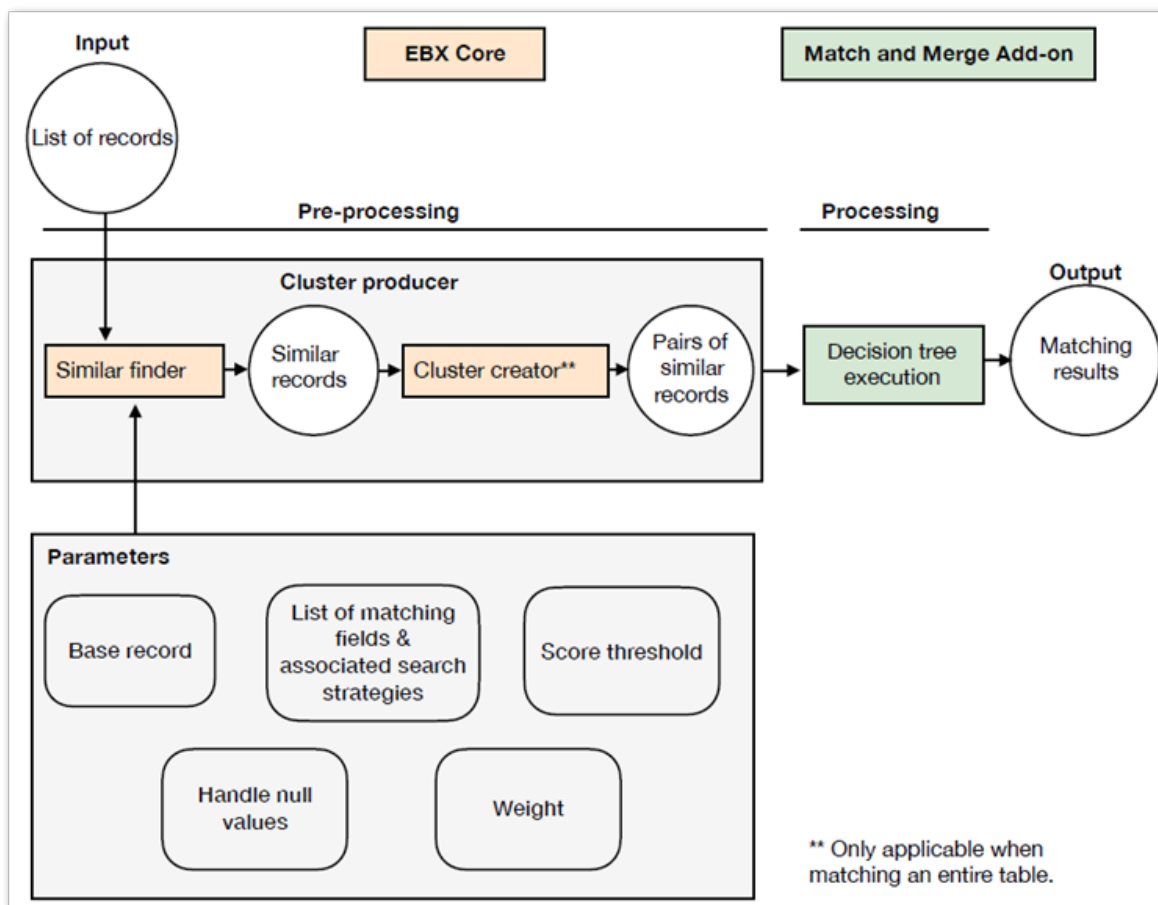
Adjusting parameter settings changes the criteria used by EBX® to find similar records. As a result, the output records passed to the add-on for processing can change. Fine-tune the settings to obtain desired results. However, keep in mind that it is wise to find a balance between the trade-off of restrictive settings vs nonrestrictive. Criteria that is too strict can result in not passing records to the add-on that a decision tree would consider a match.

To edit parameter settings:

1. Access the configuration settings for the matching field you want to update:

    1. Navigate to *Administration > TIBCO EBX® Match and Merge Add-on > Table activation and settings*.

    2. Open the table configuration and matching policy containing the field.

    3. Select the **Matching fields** tab and open the field.

2. Use the following options under **Pre-processing** to change parameter settings:

    1. **Search strategy**: Sets the search strategy used to group similar records during the pre-clustering process. The list only includes search strategies defined in the data model containing the selected field. See the *EBX® Reference Manual* for more information about search strategies.

    2. **Weight**: Specifies the weight assigned to this field during pre-processing by EBX®. When a matching operation includes multiple fields, the weight sets the relative importance of each field's score in determining whether records are similar. Note that this weight does not affect the weighted average in the decision tree comparison nodes.

    3. **Null value management**: Determines the behavior when comparing two records and one or both of them has a Null value in this field. When enabling matching of null values, adjust the weight for the field so that it is lower as compared to other fields. Otherwise, it might not get included during pre-processing with other similar records.

3. Save and close to keep your changes.

CHAPTER **15**

Perspectives and workflows

EBX® Match and Merge Add-on services can be applied to EBX® perspectives and workflows. For information on creating perspectives and workflows, see the EBX® product documentation. The add-on also allows you to specify that a workflow launches automatically when inline matching is triggered. See Creating a matching policy [p 17] for instructions on enabling the option.

> **Attention**
>
> When configuring a workflow some of the parameters require specific values. Be sure to check each parameter's tooltip to ensure expected behavior.

The following table provides descriptions of the add-on services you can use in perspecitves and workflows:

Service	Description
[ebx-addon-mame] Merge (workflow only)	Opens the add-on's **Merge** view to enable uses to manually merge records.
[ebx-addon-mame] Run match (workflow only)	Executes the **Run match** service on a set of records or entire table. Note This service is available as a workflow **User task** and **Script task**.
[ebx-addon-mame] Inline matching (workflow only)	Executes matching during the creation or update process. Users must review any potential duplicate records before continuing.
[ebx-addon-mame] Wait for processes to finish (workflow only)	Waits to execute the next workflow step until currect processes are finished.
[ebx-addon-mame] Manage group (workflow only)	Opens the **Manage group** screen where users can view and manage results related to individual records.
[ebx-addon-mame] Consolidated view (for perspectives and workflows)	Opens the **Consolidated view** where users can view matching results and access other services to manage results.
[ebx-addon-mame] Search before create (for perspectives and workflows)	Executes the **Search before create** service to allow users to search for potential duplicate records prior to creating them.

CHAPTER **16**

Displaying matching metadata

Matching metadata is automatically included in tables. However, you can create a view that displays add-on metadata in a custom location. This might be useful for larger tables to prevent users from having to scroll through many columns to locate the metadata. They could then quickly sort records by state to see if any action is required and access record groups. Metadata applied by the add-on includes:

- State
- Group
- Auto-created

To add matching metadata to a table view:

1. Open the table that you want to display matching metadata.
2. Open the **View** menu and select **Create a new view**.
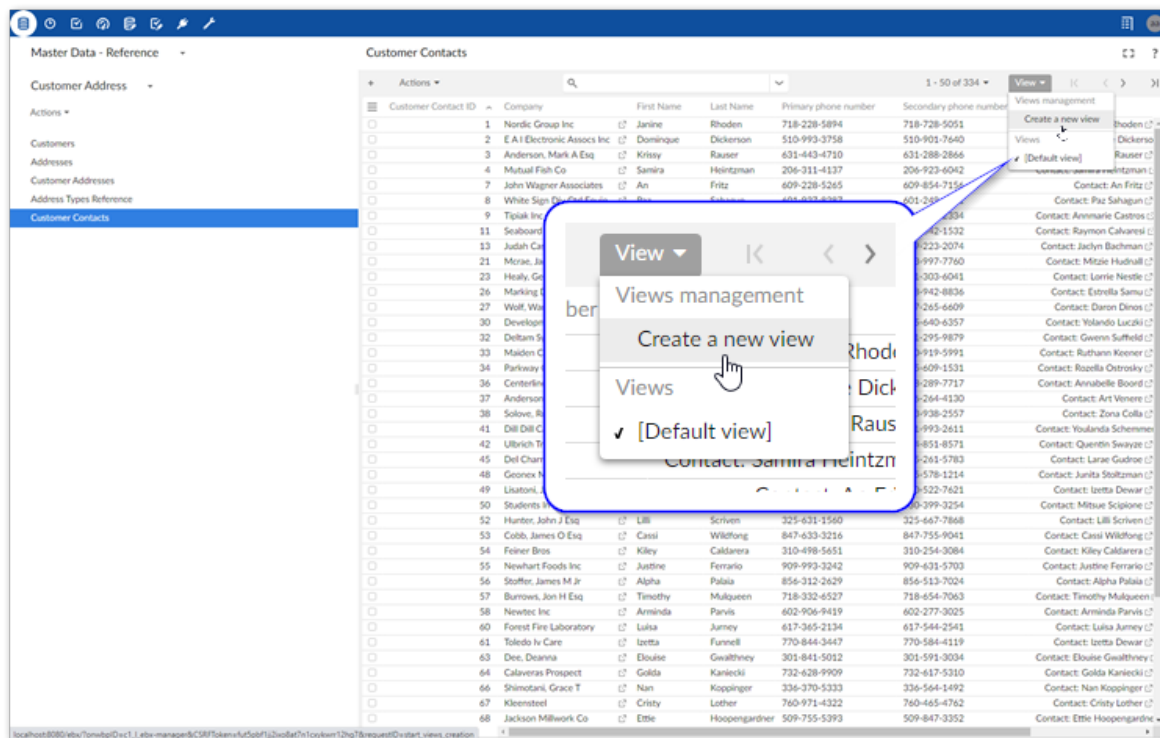
3. Optionally, provide a name and select **Next**.

4. The three matching metadata categories are automatically included in the **Displayed columns** list.
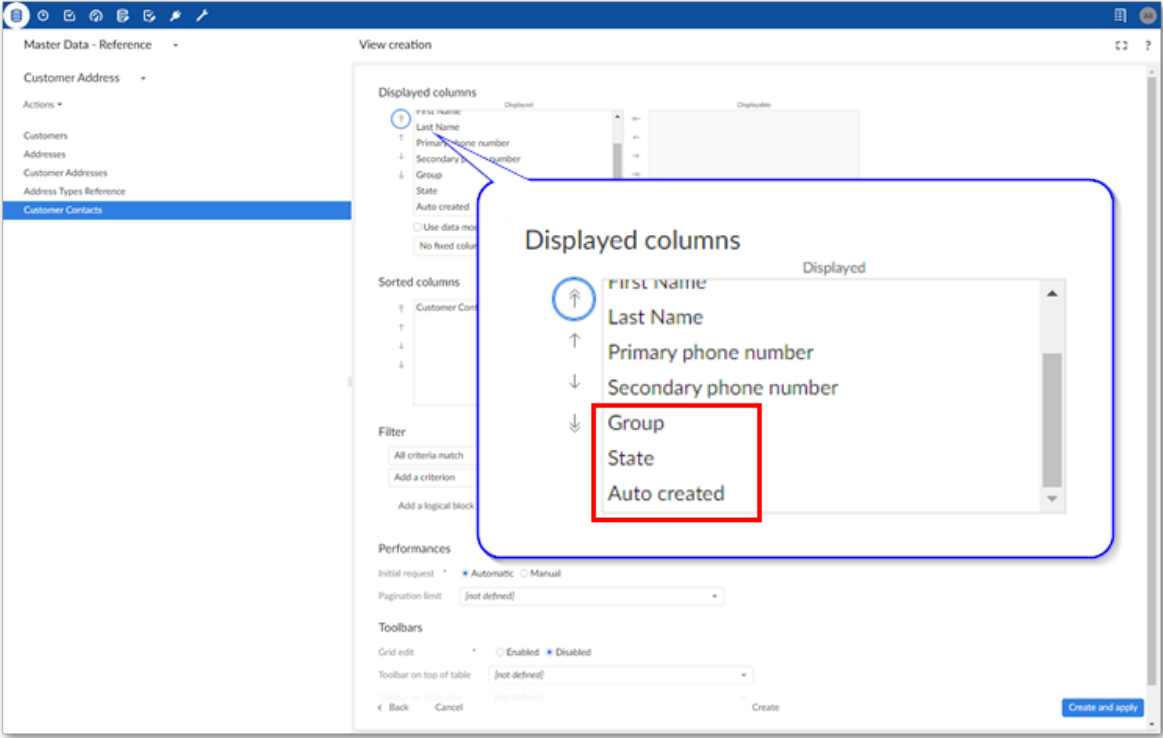
5. Make any additional desired changes to the columns displayed and their order. Select **Create and apply**.

The new columns displays in the specified location as shown below:
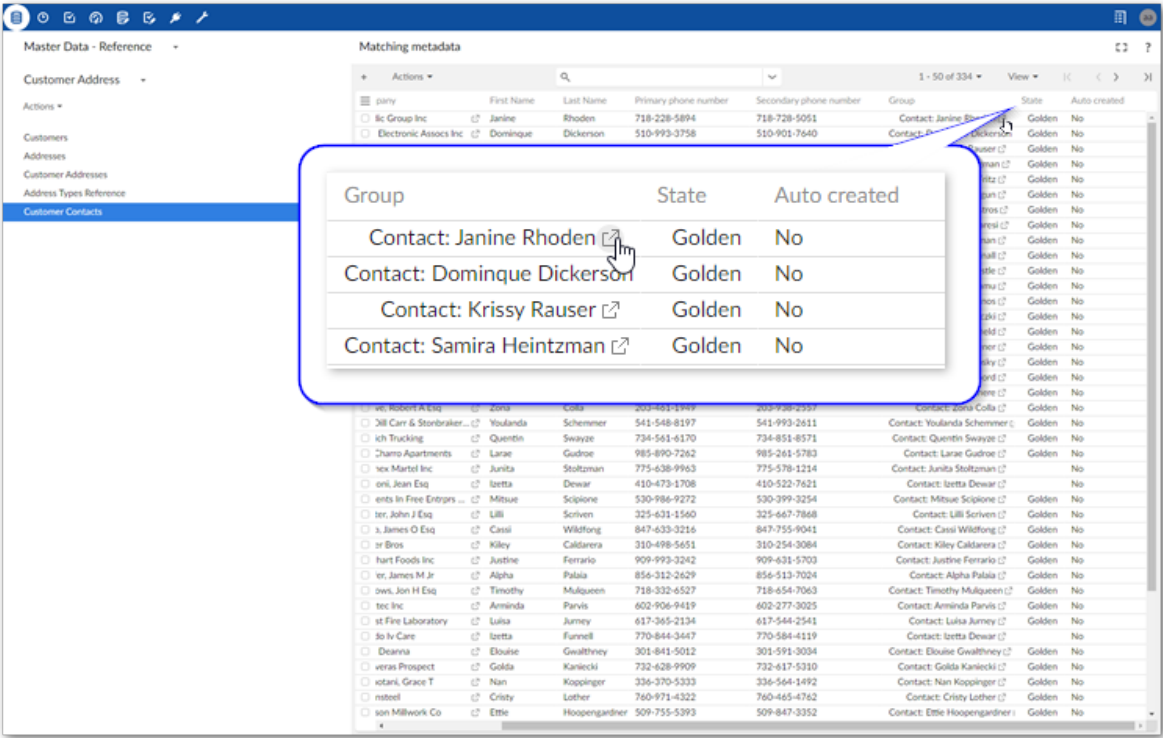
CHAPTER **17**

Matching algorithms

This chapter contains the following topics:

17.1 Included algorithms

The add-on includes several predefined algorithms to use in matching operations. Each algorithm has advantages and disadvantages depending on the type of data being searched. The following table

provides descriptions for the algorithms included with the add-on and any input parameters they might have:

Algorithm	Description
Hybrid fuzzy	An algorithm best used for string and text data types. It can capture complex relationships between two field values by automatically customizing transformations for a given criterion. Points of note about this algorithm: • Does not return a 100% matching score on synonyms. • Strings that are similar, but contain characters like spaces or dashes do not return a 0% matching score. • When strings fall within the acceptable distance score they are not considered as the exact same. For example, a distance of 1 returns .95, a distance of 2 returns .9, etc.
Phonetic full text	An algorithm best used for strings. It can recognize the phonetic equivalent of two words spelled differently using the Beider-morse phonetic tokens. The comparator takes into account the synonyms and stop words defined in the data model.
Beider-morse	A phonetic algorithm for short strings (e.g. proper names) that is able to recognize a phonetically equivalent of two words written in a different way. The "sounds-alike" test is based not only on the spelling but on linguistic properties of various languages.
Full text	An algorithm best used for strings. It can find the case-insensitive exact matches of the words in the compared values. The comparator takes into account the synonyms and stop words defined in the data model. > **Note** > Values that contain only stop words are not supported. Additionally, this algorithm is only compatible with String and text data types.
Fuzzy full text	An algorithm best used for strings. It can find the similar and fuzzy matches of the words in the compared values and is based on the Levenshtein distance algorithm. The comparator takes into account the synonyms and stop words defined in the data model. > **Note** > Values that contain only stop words are not supported. Additionally, this algorithm is only compatible with String and text data types.
Exact	An algorithm to match data that should be exact (e.g. code). This algorithm returns a matching score of 100% for values that are exactly the same. Matching using the Exact algorithm is case-sensitive by default for fields with String and Text data types. You can set it to case-insensitive when configuring data comparison nodes in the decision tree.
JaroWinkler	A distance algorithm for short strings (e.g. proper names) that tallies the number of characters in common and places a higher emphasis on differences at the start of the string. **Input parameters:** The 'Threshold' parameter values should be from 0.0 to 1.0. The parameter determines when a Winkler bonus should be added. Decreasing this parameter might result in an increased score. For example: With the keyword 'Fra' and data 'France', if the 'Threshold' = 0.7 score = 88.33. If the 'Threshold' = 0.9 score = 83.33. Default value: 0.7
Levenshtein	A distance algorithm for short strings that works well when only few differences between the values are expected. For example, this works well for dialects spoken in a particular part of the country, or by a specific group of people.

Algorithm	Description
Soundex	A phonetic algorithm for short strings (e.g. proper names) that indexes the strings based on the way they sound in English rather than the way they are spelled. The homophones are encoded to the same representation so that they can be matched despite minor differences in spellings.
Range	An algorithm to match values within a predefined range. The two values are considered a match if the distance between two values is within the range. **Note** This algorithm is only available for numeric and date/time fields.

17.2 Search strategies and recommended algorithms

Some matching algorithms are not compatible with certain search strategies and could return inaccurate matching results. Thus, when configuring a comparison node in a matching policy's decision tree, the list of available algorithms is filtered depending on a field's configured search strategy. This helps prevent an incompatible configuration. Additionally, the combination of a field's configured search strategy in the Data Model Assistant and its matching algorithm can impact performance. See Performance recommendations [p 67] for more information.

The following table shows the compatibility between search strategies and algorithms:

Search strategy	Algorithms
Default search template (set in DMA)	
Default full text search strategy or any other configured for the default template.	• Exact • Levenshtein • JaroWinkler • Full text • Fuzzy full text Not recommended but still available: • Beider-morse • Soundex • Phonetic full text
Non-default search templates	
Tese Levenshtein	• Exact • Levenshtein • JaroWinkler • Full text • Fuzzy full text
Tese NGram	• Exact • Levenshtein • JaroWinkler • Full text • Fuzzy full text
Tese Soundex	• Exact • Soundex • Beider-morse • Phonetic full text
Tese Double Metaphone	• Exact • Soundex • Beider-morse • Phonetic full text
Jaro-winkler	• Exact • Levenshtein • Jaro-winkler • Full text • Fuzzy full text

CHAPTER **18**

Performance recommendations

This chapter contains the following topics:

1. Overview
2. EBX® optimization
3. EBX® Match and Merge Add-on optimization suggestions
4. Troubleshooting

18.1 Overview

The EBX® Match and Merge Add-on and EBX® core product work together to handle matching tasks. Due to this shared responsibility, the biggest performance gain comes from making optimizations in both the add-on *and* the core product. For more details on the roles of each during the matching process, see Matching processing overview [p 57].

The following topics describe:

- EBX® optimization [p 67]
- EBX® Match and Merge Add-on optimization suggestions [p 69]
- Troubleshooting [p 69]

18.2 EBX® optimization

To ensure the best performance when matching, it is important to follow EBX® performance guidelines, especially for high volumes of data. It is of particular importance to allocate only half of the application server's available memory to the JVM. This leaves the other half free for file system caching. Additionally, the pre-processing phase is optimized to benefit from multiple CPUs. This means that CPU oriented hardware can improve matching performance.

Identifying indexing issues

The search strategies applied to fields involved in matching can have a large impact on performance. The following topics present some best practices that can help prevent performance issues:

- Stop words [p 68]
- Prefer code search strategy [p 68]
- Avoid empty or null value placeholders [p 68]

- [Use fuzzy and phonetic search strategies sparingly](#) [p 68]

Stop words

String values that contain common tokens greatly increase the number of irrelevant match candidates found when pre-processing and slow the matching process. To alleviate this, you can define stop words, which EBX® ignores during pre-processing (i.e. they are not indexed). Stop words are registered in a module, then defined in the search strategy of an `xs:string` field in the **Search** extension of the Data Model Assistant.

The best candidate terms for stop words are frequently found values. For example, 'street' and 'blvd' make good stop words because they exist in almost all records related to addresses.

> **Note**
>
> The result of a matching process is not affected by a set of well-chosen stop words. They are ignored by the core product during pre-processing, but are still taken into account when executing the add-on's decision tree logic.

Prefer code search strategy

The **Name** and **Text** search strategies tokenize values to support full-text search. Nevertheless, some data are ill-suited for tokenization. This can lead to an abundance of common tokens, which slows the matching process. This especially holds true for `xs:string` fields holding identifiers, dates, phone numbers or emails. For these cases, the preferred method is to use the **Code** search strategy, which ensures that the whole value is indexed as a single token.

Avoid empty or null value placeholders

When a value is null or empty, do not use a placeholder like 'Not available', 'N/A', or default values like '1', 'xxx', etc. Those placeholder are common tokens, and slow down matching. It is recommended to leave the value empty instead.

Use fuzzy and phonetic search strategies sparingly

The fuzzy and phonetic search strategies are slower than the default, built-in strategies. In the context of matching, this impacts:

- Core product performance when gathering similar records during pre-processing. The cost is mainly due to the increased number of results for a fuzzy strategy even if only applied to one decision tree node.
- Add-on performance is impacted as fuzziness can add a lot of false positives in the results. The increase in the number of records passed to the add-on can slow the process. Since these additional records might not meet conditions of other comparison nodes, the add-on is required to process additional records with no guarantee of finding matches.

If the conditions on other comparison nodes are met, then a record is considered 'similar' during pre-processing, even without applying a fuzzy strategy to a field. Hence, we recommend keeping the default search strategy when using fuzzy or phonetic algorithms in the decision tree, and when the decision tree contains at least 4 fields.

However, if your matching policy still requires a fuzzy or phonetic search strategy, the following information can help to mitigate the performance impact:

- Jaro-Winkler is the most expensive.

- The Levenshtein search strategy accepts parameters to tune its behavior, which impact performance. First, Levenshtein is an 'edit-distance based' algorithm. If distance 2 is too expensive, try decreasing the distance to 1. Second, there is a parameter to specify whether the values are tokenized or not. Without tokenization, the algorithm behaves like a fuzzy 'Code' strategy, and is less expensive. With tokenization, it behaves like a fuzzy 'Text' strategy. The fuzziness is applied to all tokens, but it greatly increases the number of results returned and is more expensive. The parameters are configured in the 'Search' extension of the DMA

- The phonetic parameter of 'Name' and 'Text' search strategies uses the Beider-Morse strategy. This is the least expensive.

18.3 EBX® Match and Merge Add-on optimization suggestions

The following matching policy recommendations can help improve performance:

- **Decision tree:** Before performing matching on an entire table with a large volume of records, test the matching policy on a subset of records (less than 50k records). And:

  - Check that the number of matches and suspects is consistent with your dataset. If it is not consistent, start the troubleshooting process with the decision tree. Begin by reviewing the nodes leading to the **Match** or **Suspect** output.

  - When a grouping or merging phase is slower than usual, it is likely the symptom of other configuration issues.

- **Weight:** When a decision tree has a **Match** output that is reached after 1 or 2 comparison nodes, it means that the fields involved in these nodes are more important than the others. To improve performance and result quality, set the weight parameter higher for these fields to help identify them during pre-processing.

  > **Note**
  >
  > To improve matching performance, set the weight to 0 for fields that have a limited number of options. For example, fields that are boolean or enumerations.

- **Exclude records:** By defining a set of criteria that excludes certain records from the matching operation, you can eliminate a subset of records that the add-on must process. This can lead to performance improvements. See Excluding records [p 39] for more details.

18.4 Troubleshooting

Activating the logs can help you investigate matching process performance. To activate the logs of the pre-processing phase, open `ebx.properties` file and set the property `ebx.log4j.category.log.kernel` to `DEBUG`. With this setting, it displays precise information about matching progress: number of records, average throughput, percentage of completion. This is useful to estimate the duration of the matching phase. It can also help you determine whether pre-processing is slower than it should be, considering the volume of data.

CHAPTER **19**

Replicating matching metadata

This chapter contains the following topics:

1. Overview
2. Configuring metadata replication

19.1 **Overview**

Replicating your matching metadata allows you to use third-party tools to query the metadata. For example, you might want to expose matching metadata to a BI dashboard for analysis.

During configuration you specify which table's metadata you want to replicate. The configuration is done at the matching table level. You specify which table publication to get metadata from, the method used to trigger table refresh, and the table name in the database for the replicated metadata table.

Some of the information included in the replicated metadata is each record's:

- Group id
- State
- Golden id (For merged records)
- Merging process id

Note that the State only includes the identifiers used for matching states. The ids are mapped to their state in the following table:

Id	State
0	UNSET
1	GOLDEN
2	MERGED
3	PENDING APPROVAL
4	DELETED

19.2 **Configuring metadata replication**

To enable matching metadata replication:

1. Add replication to a matching table configuration:

    1. Open the desired matching table configuration.

    2. Select the **Replications** tab.

    3. Fill in the required information. Note that the **Table name in database** property must specify a unique name in the database. Additionally, this name must conform to any database requirements.

        - If set to **On commit**, the replication metadata is pushed to the database after matching executes and commits happen in the EBX® repository.

        - If set to **On demand**, a user must manually push the replication metadata to the database from the dataset's **Actions** menu by selecting: *Replication > Refresh replicas*.

2. Publish your data model. If you come back to this replication configuration and make changes, you must re-publish your data model to apply them.

> **Attention**
>
> When setting up replications, the maximum length for table's primary key must be specified (not left undefined) in the Data Model Assistant (DMA).

Developer Guide

CHAPTER **20**

REST overview

The TIBCO EBX® Match and Merge Add-on allows access to its services using a RESTful API. The add-on follows the OpenAPI Specification (OAS) to document its REST API. This allows you to use tools such as Swagger to generate and interact with our REST documentation. It is not our intent to document how to use these tools here. However, the following example shows one of the more basic ways to view the REST API documentation.

To generate the add-on's OAS:

1. Use a tool such as Postman to send a GET request to the following endpoint on your EBX® server: `http://<host>:<port>/ebx-addon-mame/rest/v1`.

2. Ensure the response is formatted in JSON.

3. Save the response to a file.

4. Open a web browser and navigate to Swagger Editor: `https://editor.swagger.io/`.

5. Select *File > Import file* to import the add-on's specification.

   The editor displays the formatted text on the left and HTML on the right.
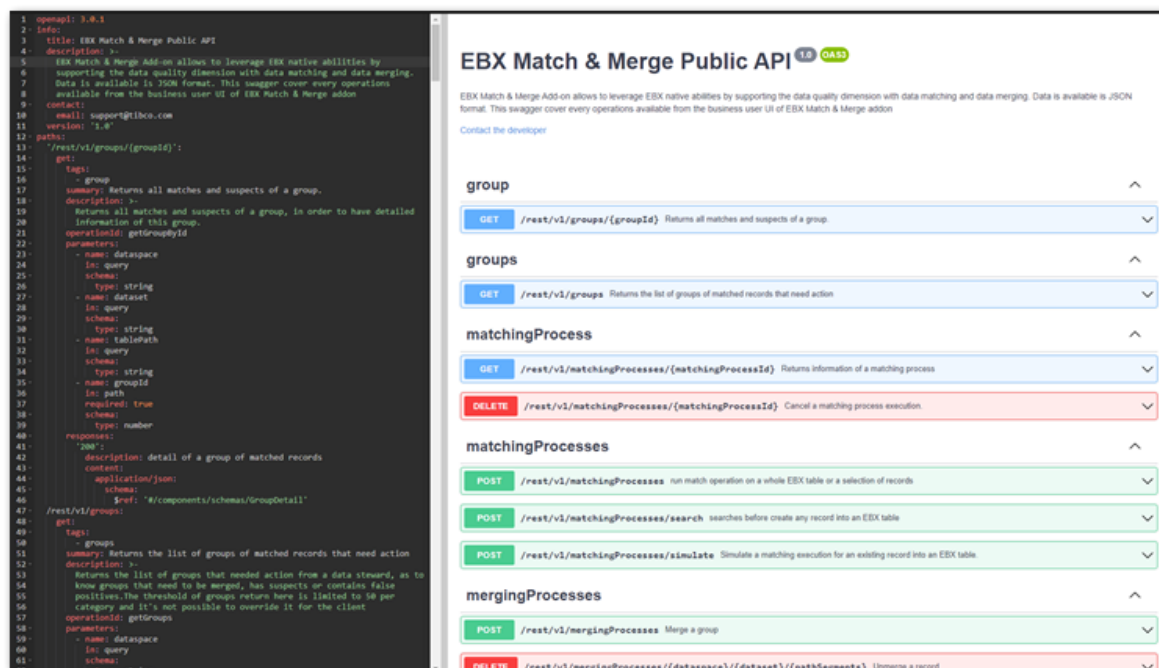
User Guide

CHAPTER **21**

Matching and merging overview

This chapter contains the following topics:

1. Overview
2. Matching
3. Merging

21.1 Overview

The TIBCO EBX® Match and Merge Add-on's purpose is to locate and merge duplicate data into a *golden record*. A golden record represents the most truthful, accurate, and up-to-date data. As a business user, the way you interact with the add-on to match and merge data depends on configuration settings.

21.2 Matching

The add-on's *matching operations* entail looking at data values to determine whether duplicate records exist. Administrators will configure how the add-on accomplishes this. For configuration instructions, administrators can use the *Quick Start Guide* [p 10].

At any time, you can manually initiate a matching operation. When administrators enable inline matching, the add-on automatically matches newly created or updated records against existing table records. When the add-on identifies positive matches during creation or update, you will be prompted to make changes to the record or access merge functionality. See Automatic Matching [p 95] for instructions on initiating a matching operation.

After a matching operation, you can view matching related information in the **Consolidated view**. Also, you can access the **Manage group** screen to analyze and take actions on records that are confirmed matches or possible matches (suspects). See Consolidated view [p 113] and Managing groups [p 103].

21.3 Merging

Merging records is the process of combining data values from multiple records into a single, golden record. Records involved in a merge operation that do not become the golden record are called *merged records*. Administrators create *merge policies* that determine how the add-on automatically merges data values. Merge policies can also pre-select data values when you initiate a manual merge.

If a merge policy is configured to happen automatically, the add-on merges records after a matching operation. This happens whether matching was manually or automatically initiated. If matching and merging are set up for automatic execution, these processes will be transparent to you. However, you can view the results in the **Consolidated view**.

When you manually initiate a merge, you use the **Merge view** to select the data values to survive into the golden record. See [Merging Records](#) [p 107] for details on merging. You can manually initiate a merge:

- From a table's **Actions** menu.

- From the **Manage group** screen.

- When an inline matching operation finds a positive match while creating or updating a record. (as long as automatic merge isn't enabled)

Matching and Merging Walkthrough

CHAPTER **22**

Overview

The following image and accompanying table provide a high-level overview of the matching and merging workflow:

Matching policy initiates	How the add-on is configured determines when the matching policy is initiated. The most common methods are:
	• A user manually initiates the matching policy.
	• The add-on automatically initiates the matching policy.
	See <u>Part 1 of this walkthrough</u> [p 83] for more information.
Data is processed	While the add-on is processing data you can choose to run the operation in the background and navigate in EBX®, or cancel the operation if needed.
Consolidated view	The **Consolidated view** displays matching results. You can initiate data stewardship tasks from here. These tasks include:
	• Validating the results.
	• Managing groups of candidate records by determining whether to accept suspect records and begin the merge process.
	See <u>Part 2 of this walkthrough</u> [p 85] for more details.
Merge process	The merge process allows you to create a golden record by merging the most accurate data from candidate records.

See for additional information.

CHAPTER **23**

Part 1: Initiate matching

The following steps show how to initiate the matching process manually:

1. Open a table where the add-on is enabled.

2. Determine the matching process scope:

   - To include only specific records in matching, select those records now.

   - Select all records to run matching on the entire table.

3. From the table's **Actions** menu select *Match and Merge > Run match*.
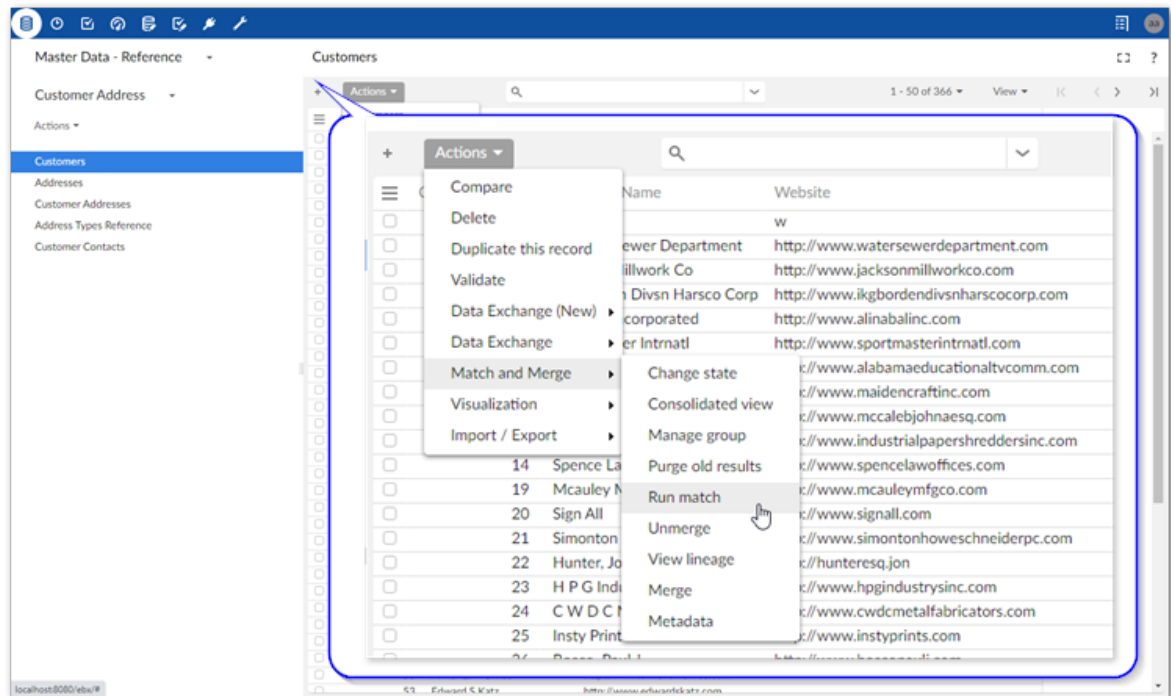
If the matching process takes a while, you can choose to run the process in the background and perform other limited tasks, such as navigate and view your data.
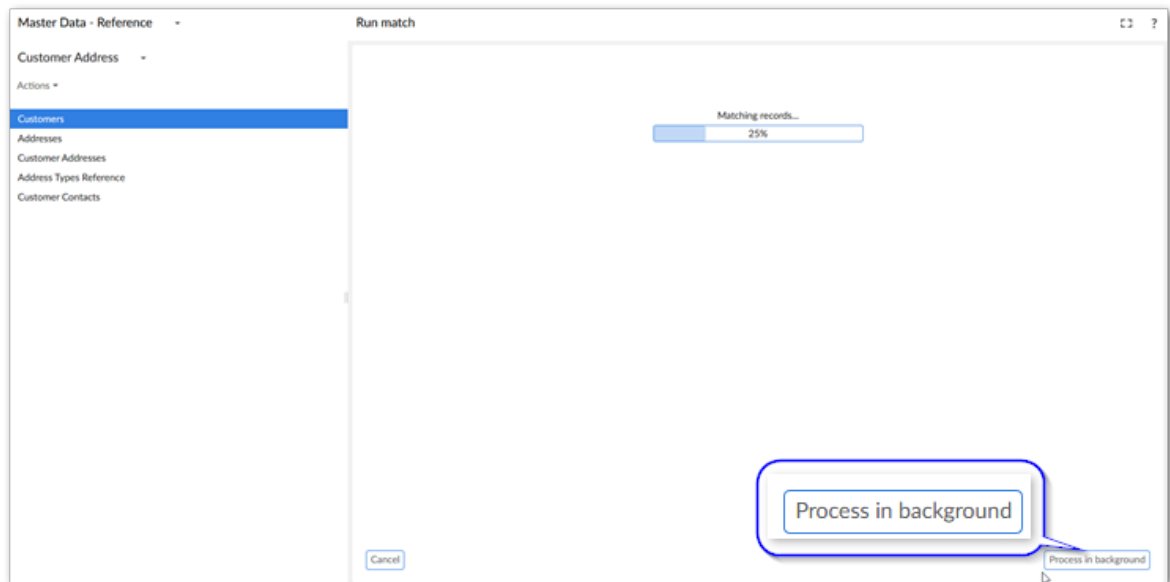
When the matching process completes successfully, the add-on:

- Depending on configuration settings, prompts you on whether to purge matching results.
- Displays the **Consolidated view** or **Manage group** view.

What's next? See <u>Part 2: Using the Consolidated view</u> [p 85] for additional details and instructions.

CHAPTER **24**

Part 2: Using the Consolidated view

This chapter contains the following topics:

24.1 Overview

The **Consolidated view** is a stewardship view that displays matching-specific metadata. The metadata is linked to each record that was part of a matching process. From this view, you can:

- Get a global view of matching results. Part of the results include: warnings/errors, grouped records, record states, and whether the record was part of the auto-creation process.

- Run a validation report to retrieve information on any groups that contain errors or warnings. The validation report will include any groups flagged for containing suspect records. Suspect records share enough similarity with another record that they are candidates to represent the same entity. You can access and manage the groups of suspects from the validation report.

- Manage matching results by entering the **Manage group** view. Here you can accept or reject candidate records. You can also enter the **Merge view** and reconcile differences between records.

24.2 Opening the view

The **Consolidated view** automatically opens after a manually initiated matching completes. However, you can access it at anytime after matching runs on a table.

As shown below, you can open the **Consolidated view** from a table's **Actions** menu by selecting *Match and Merge > Consolidated view*:
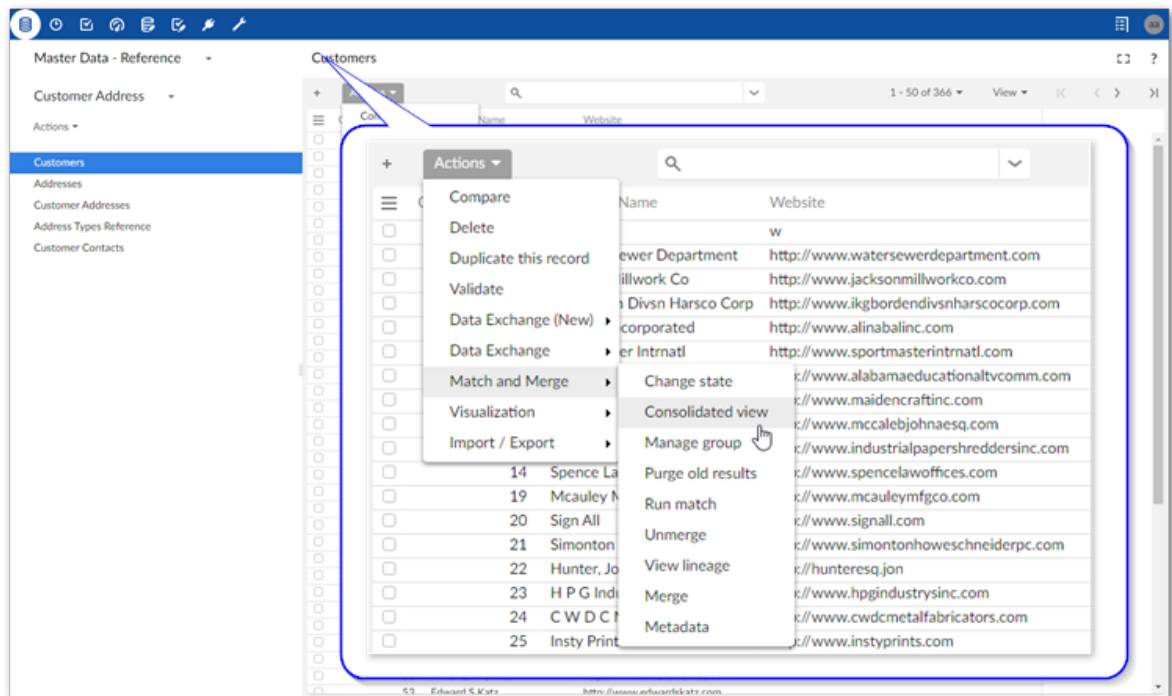
24.3 **Matching result stewardship**

Stewardship is required when matching finds suspect records and:

- your merging policy is set to manual.
- results are not definitive enough to trigger an automatic merge.

As a steward in these cases, you will most likely want to verify results, and when required, merge records. The steps below show how to do this using the options in the **Consolidated view**:

1. Find groups of records that require your attention by selecting **Validate** at the top of the view.
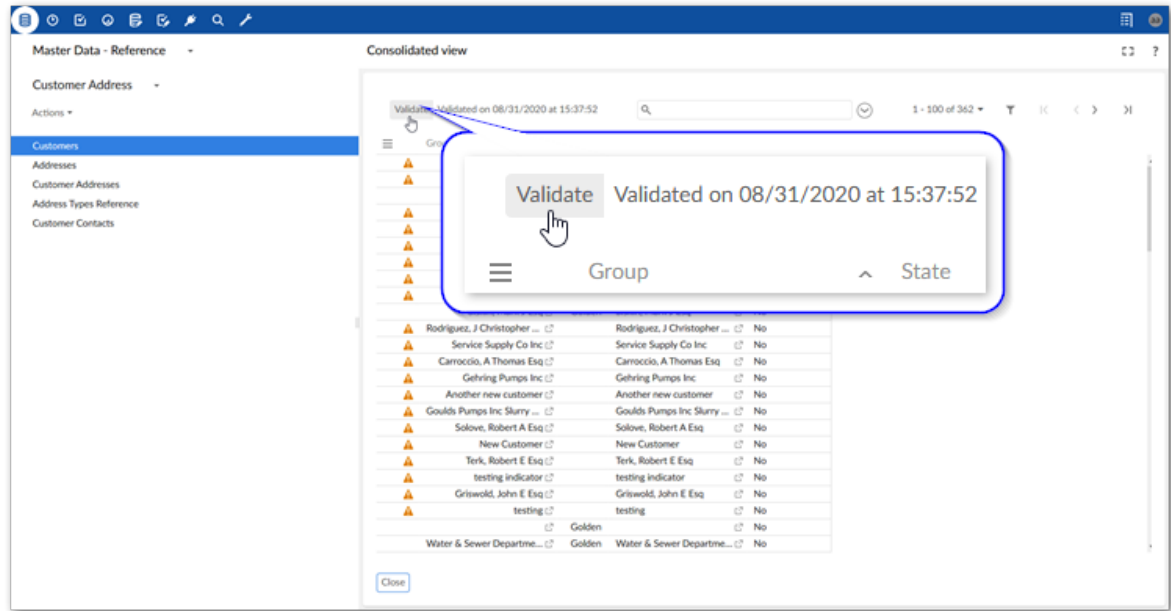
2. Analyze matching results:

    1. Locate the **Warnings** description that indicates there are suspect records.

       Each entry is a group.

    2. Select a group from the list to enter the **Manage group** view.
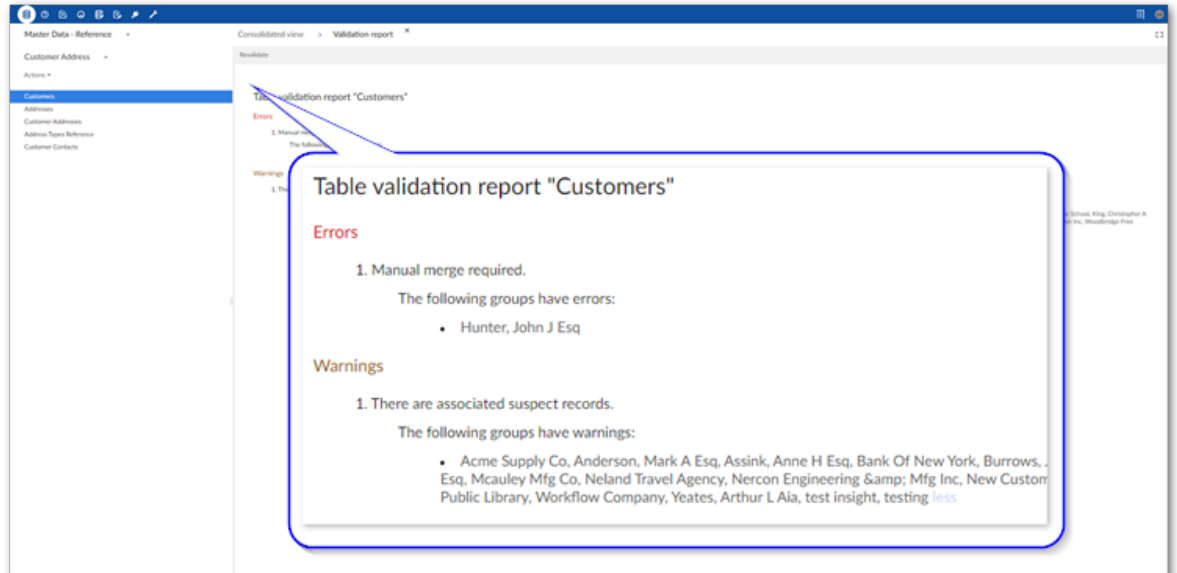
3. Manage groups of suspect records:

   The idea is to determine whether records in the **Suspects** and **Matches** lists refer to the same entity. To assist in this process the add-on allows you to:

    1. Compare records: Select two or more records in one of the lists and open *Actions > Compare*.

    2. View record metadata to gain insight related to matching such as current state, group, and execution log: Open *Actions > Metadata*.

3. Once you have made a determination, use the **Actions** menu or select the '-' icon to remove any false positives. Use the **Actions** menu or the check mark icon to confirm suspects and add them to the **Matches** list. When the required suspects are added to the **Matches** list, you can merge the data from these records.

What's next? Use the instructions in <u>Part 3: Merging suspects</u> [p 89] to merge records.

CHAPTER **25**

Part 3: Merging suspects

This chapter contains the following topics:

1. Overview
2. Merging suspect records

25.1 Overview

The **Merge view** allows you to merge data from multiple records into a golden record. This record is the most accurate, up-to-date, and comprehensive representation of an entity. The following section demonstrates how to use the view.

25.2 Merging suspect records

To merge data from suspect records and create a golden record:

1. Enter the **Merge view**:

1. From the **Manage group** screen (first image) or after inline matching executes (second image), select **Merge**.
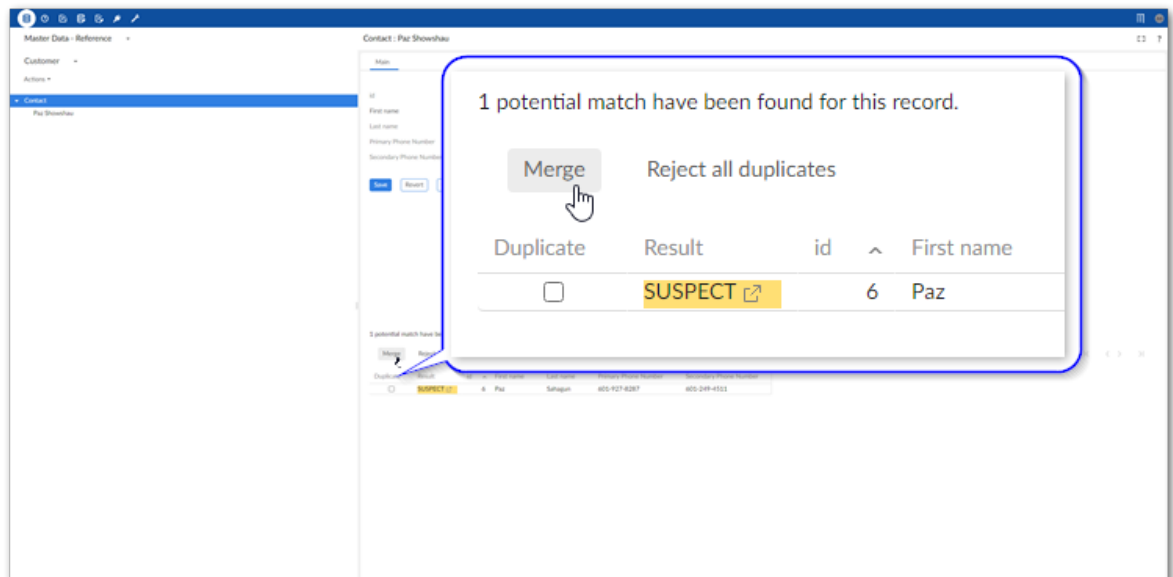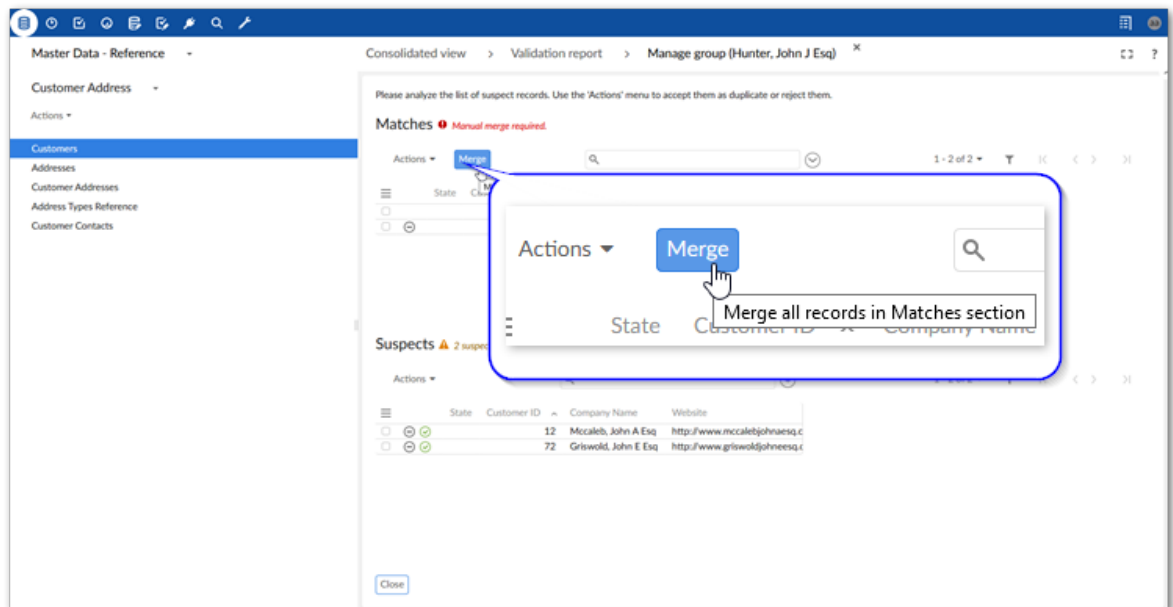
2. You can manually select records and from a table's **Actions** menu, select *Match and Merge > Merge*.

2. Select the values to persist into the golden record:
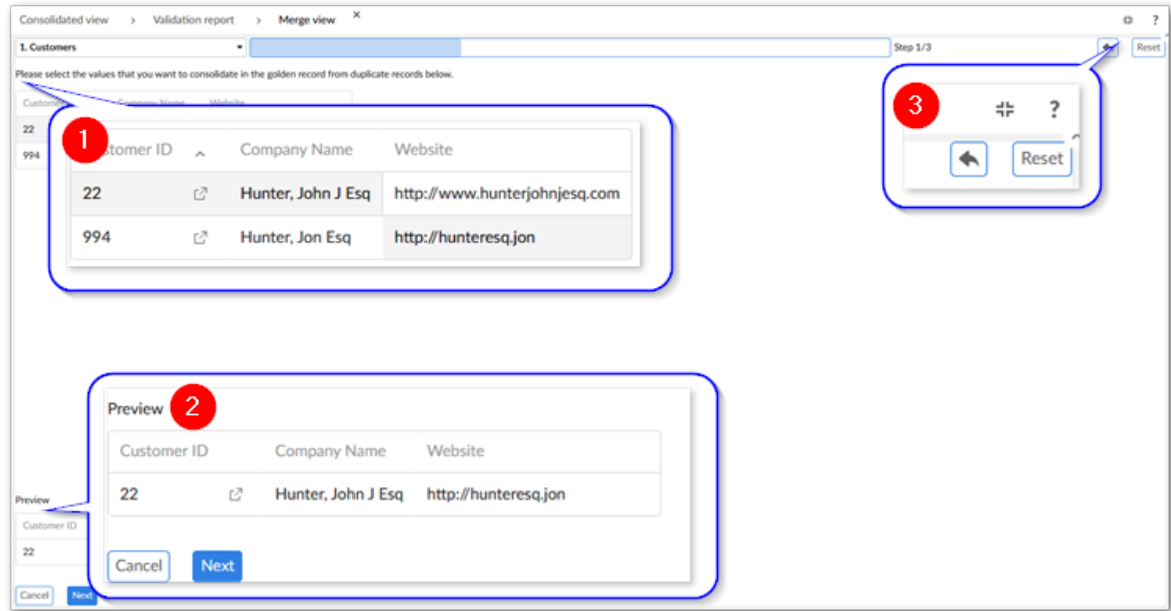
1)	Compose the candidate record by selecting the values to persist. Depending on how your administrator has configured the merge feature, the system might pre-select certain data values. Selected values are shaded in gray.
2)	The **Preview** shows what data is included in the golden record. It updates in real-time based on your selections.
3)	Use the **Revert** icon to undo up to 5 of your most recent modifications. Depending on your configuration settings you can use the **Reset** or **Apply merge policy** buttons to reset all of your modifications.

3. Select **Next** to preview results.

4. Select **Merge** to complete the process.

   You can select the link to access the record, or **Back to main view** to return to the previous view.

General Tasks and Views

CHAPTER **26**

Automatic Matching

If automatic matching is enabled and set to inline, the add-on finds positive results when you create or update a record, the add-on displays the following screen. The list below the screen describes actions you can take here.
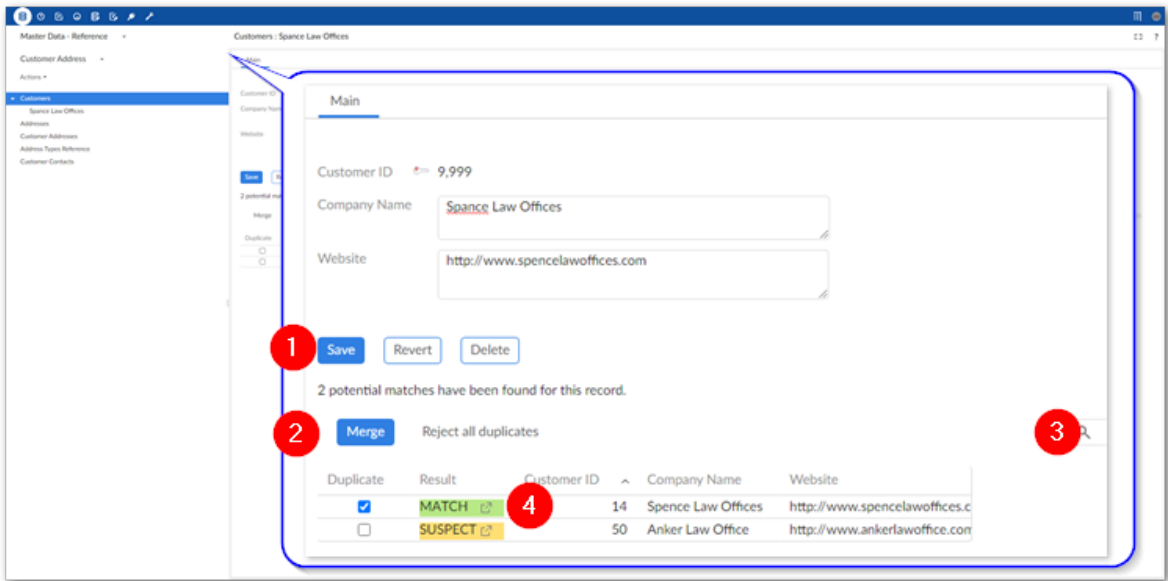
1) Save, Revert, and Delete	You can make changes to record fields and select **Save** or **Save and close**. The add-on runs another match. If it finds no duplicates, the record saves normally. The **Revert** and **Delete** buttons work as expected and revert any changes or delete the record, respectively. Note that the option to delete is only available when matching is triggered during the process of creating a new record.
2) Merge and Reject all duplicates	You can select the **Duplicate** checkbox for any positively matched record and select **Merge** to open the **Merge view** and merge values. By selecting **Reject all duplicates**, the new or updated record will be saved in the table as-is. See Merging Records [p 107] for available actions in the **Merge view**.

3) Search bar	Use the search bar to locate records in the list of potential matches.
4) Results	Select the details icon in the **Result** column to display the decision tree and view execution of the logic used to identify the matches. When viewing the decision tree, you can double click a node to view the matching score.

CHAPTER **27**

Evaluating matching

This chapter contains the following topics:

1. Overview
2. Running the evaluation
3. Evaluation results

27.1 **Overview**

The add-on's **Evaluate matching** service allows you to check the outcome of a matching operation against expected results. This can help you fine tune configuration settings to meet your specific needs.

This service provides the following benefits:

- Quick results
- Ability to test different policies without exiting the service
- Uncover points of matching execution failure
- Upon completion, view details that show how the algorithms score and a match/no match was reached
- With administrative credentials, directly access configuration settings without leaving the screen

27.2 **Running the evaluation**

To run the service:

1. Select two records from a table with a configured matching policy.
2. From the table's **Actions** menu, select **Evaluate matching**.
3. Use the **Matching policy** menu to select the policy you want to test.

   With sufficient permission, you can access the matching configuration and make changes.
4. Select **Launch** to run the evaluation.

   After running, the results display at the bottom of the screen.

27.3 **Evaluation results**

The following image and list describe results of the matching process:

> **Note**
>
> For more details on the matching process, see Matching processing overview [p 57].

1. As highlighted in the above image, some matching policy configurations might cause inconsistent results in the pre-processing phase of matching. If the process stops at step 2, the records do not pass the pre-processing step and are prevented from evaluation using the matching logic in the decision tree. If you find that records are not making it past pre-processing, you can:

   - Analyze the fields configured to use matching. Depending on your data, too many irrelevant, or too few important configured fields can cause the system to view records as not similar.

   - Ensure that the correct weight is used for fields used in matching. Assign higher weights to the more important fields.

   - Verify that you are using the recommended search strategies for each field's data type.

2. Change policies or configuration settings on the fly and re-test. Select the ⎘ icon to access policy settings.

3. Step 4 shows the final decision tree result. (Match/No Match/Suspect) Select details to view a map of the decision tree execution. Double-click a data comparison node to view details about how the records were processed.

CHAPTER **28**

Searching before taking action

This chapter contains the following topics:

28.1 Overview

The **Search before create** service allows you to use the default matching policy configured for a table to search through existing records to prevent duplicating data. If an administrator has enabled the **Search before create** service you can manually initiate it prior adding records.

You can decide which fields, configured to use matching, are included when searching for duplicates. If the search turns up any possible matches, the add-on presents you with a list. You can then choose whether to update an existing record, or create a new one.

> **Note**
>
> You cannot use fields from associated tables with this service when matching is configured to use relationships.

If you are an administrator, Task 3a: Creating a matching policy [p 17] provides information on enabling this option.

28.2 Using search before create

Once enabled by an administrator, you can search for potential duplicates prior to creation using the following steps:

1. From a table's **Actions** menu, select *Match and Merge > Search before create*.

2. Fill the available fields with information from the record you want to create and select **Search**.

3. After the search completes you have the options to select:

   - **Reset** to clear the search form's fields.

   - **Update record** icon in the **Potential matches** panel to change values in an existing record.

   - **Create a new record** to add a new record to the table. In the creation form that displays, any fields that included searched values will be populated with these values.

- The details icon (in the **Results** column) to view the decision tree and see an overview of the logic used to determine positive matches. When viewing the decision tree, you can double click a node to view the matching score.

CHAPTER **29**

Manual Matching

You can manually initiate a matching operation on:

* A selection of records to match them against the other records in the table.

* An entire table. All records are matched against each other.

To run a manual match on a table that has an active matching policy:

1. Set the matching scope by selecting one or more records.

2. Open the **Actions** menu and select *Match and Merge > Run match*.
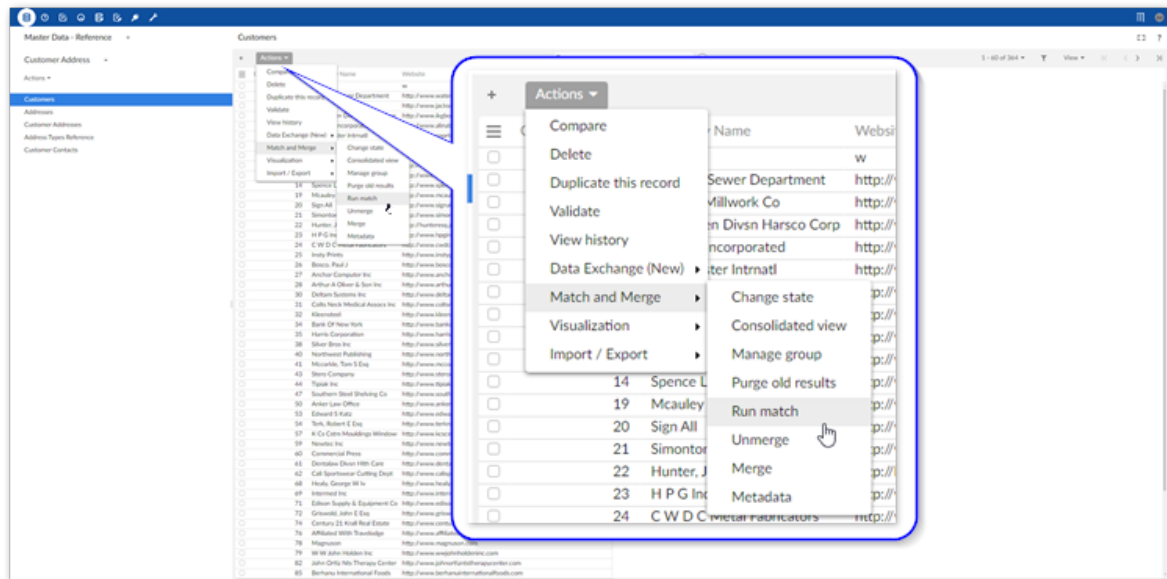
3. In the **Run match** screen, select the matching policy to use from the list of active policies and run the match.

   If you executed the match:

   * On an entire table or a selection of multiple records, the **Consolidated view** displays. See for more information about actions available.

- With only one record selected, the **Manage group** screen displays. See <u>Managing groups</u> [p 103] for more information about actions in this screen.

> **Note**
>
> While the matching process is running, the add-on displays its progress. At any time you can cancel the process, or navigate in EBX® to view other data. When the matching process completes, you can view the results in the **Consolidated view**.

CHAPTER **30**

Managing groups

During matching, the add-on groups records that meet the criteria to be considered positive matches. The add-on also identifies records that are suspected of matching records in a group. When you manage a group of records, you can compare records, view record metadata, accept suspect records as duplicates, and reject false positive results. You can also enter the **Merge view** to merge data into a golden record.

After a matching operation, you can open the **Manage group** view:

- By selecting a record and then accessing the **Manage group** service from a table's **Actions** menu.

- From the **Consolidated view** or **Metadata view** by selecting the **Manage group** icon next to a group.

The following image and table describe available actions:
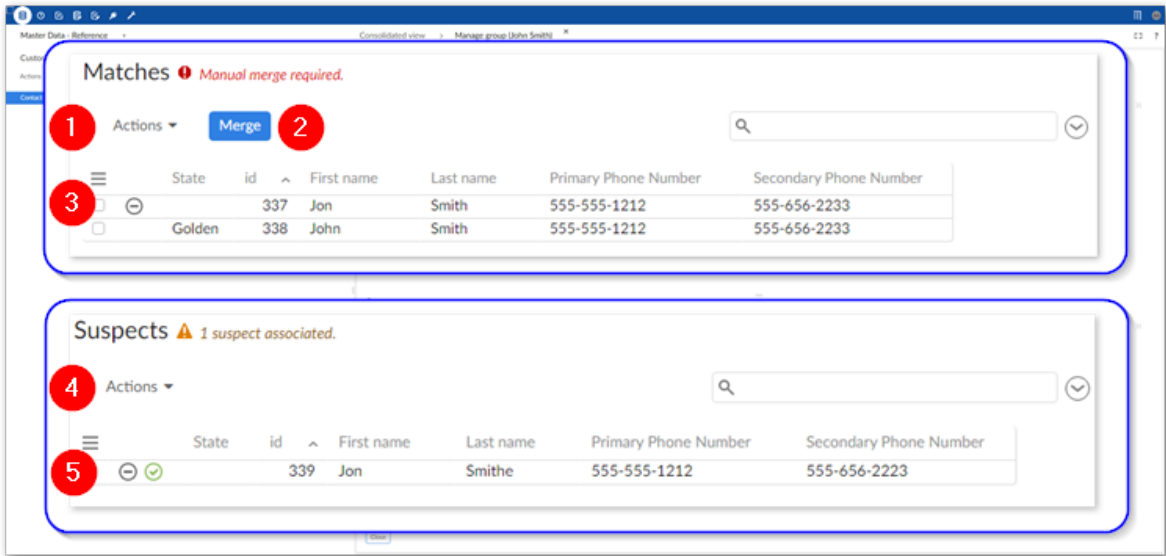
1) Matches pane Actions menu	This menu allows you to:
	• Compare two selected records using the **Compare** service.
	• Use the **Metadata** service to view a selected record's matching-specific metadata. When viewing metadata, you can use the **Identified as false positives** tab to see records that were manually identified as not matching with the selected record. False positive records are no longer grouped with a records suspects or matches. You can use the **Actions** menu when viewing metadata to remove the false positive categorization to reverse this behavior. Note that in some instances, merged records marked as false positive can be re-included in groups. In these cases the add-on uses the ⊠ icon as a designation. You can hover your mouse over the icon to view list of related false positives.
	• Use the **Unmerge** option to revert a merge operation.
2) Merge or Set as golden (not shown)	The **Merge** button opens the **Merge view** and makes all of the records in the **Matches** pane available for a manual merge.
	The **Set as golden** button displays when the **Matches** group contains only one record and its state is not yet golden. Selecting this button sets the record to golden. If this button is enabled and there are suspects, setting the record to golden will automatically reject the others as suspects.

3) List of matches	The list of records identified as matching. Records are considered matches if they meet the criteria set out in the matching configuration, or if you confirm a record in the **Suspects** list. Use the **Reject** icon to remove a record from the list.
4) Suspects pane Actions menu	In addition to compare and metadata access, you can select records and: • Use the **Accept** service to add it to the list of **Matches**. • Use the **Remove** service to remove it from the list. When you reject a record, the add-on marks it as a false positive and will no longer include it in groups with the current records.
5) Reject and Accept icons	Use these icons to reject or accept a suspect record as duplicate.

CHAPTER **31**

Merging Records

The process of merging data allows you to select the field records that contain the most accurate data and *merge* them into a golden record. A golden record represents the most accurate and up-to-date version of data.

To merge records:

1. Access the merge view in one of the following ways:

   - When in the EBX® table view, select two or more records to merge:
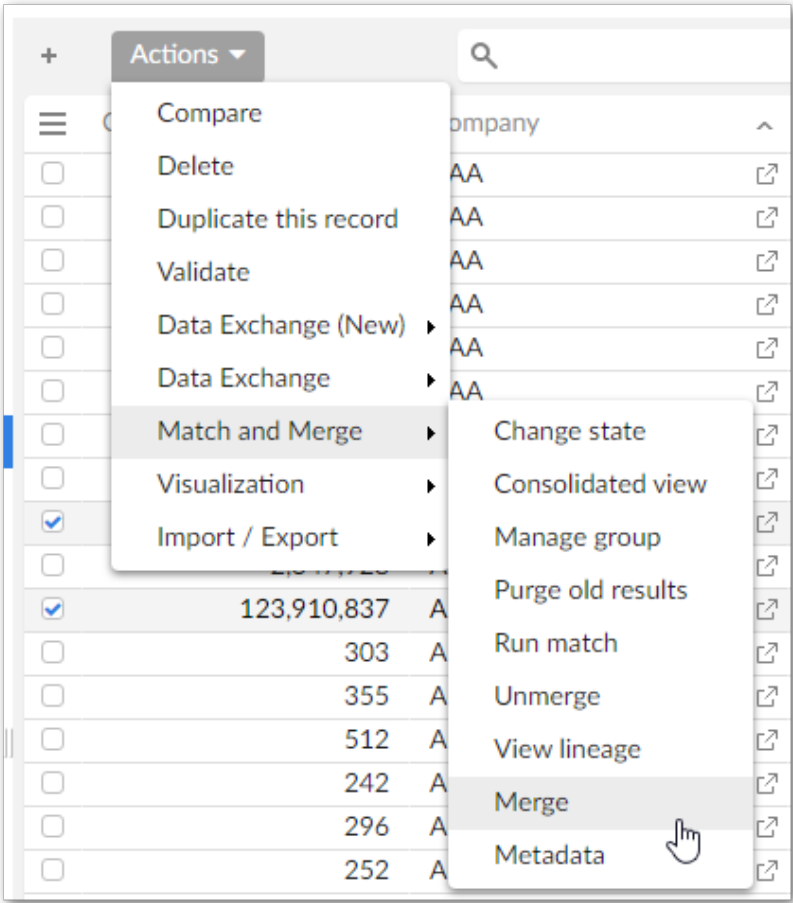
- When using the add-on's **Manage group** service:
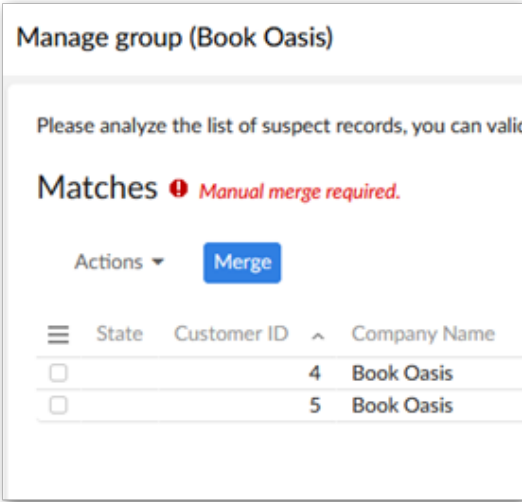
- When inline matching is active and finds potential matches for a newly created or updated record.
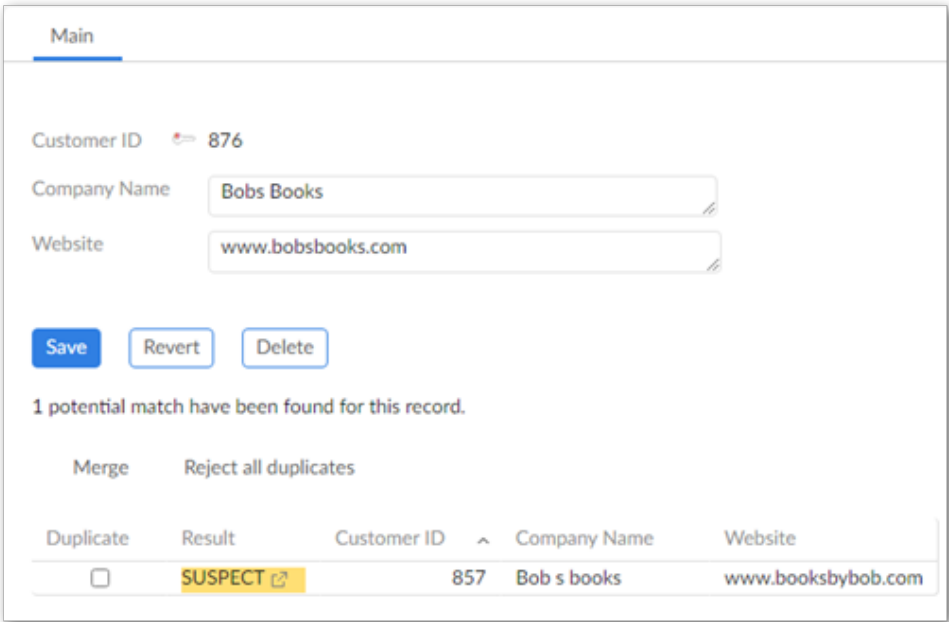
2. Click to select the fields you want to merge into the golden record. If not already selected automatically, selecting a record's primary key selects the entire record. You may want to select the most accurate record first to act as a baseline. Then you can fine tune values by selecting other fields. The choices you make are reflected by the preview at the bottom of the page. Once

satisfied, click **Next**. Note that if a merge policy is configured, some values will be pre-selected. You can still change these values.

1) Undo and Reset or Apply merge policy	You can undo your last five actions, or reset all selected values to their original selections when entering the **Merge view**.
2) Data selection area	Select the data values that should be survived into the golden record.
3) Preview record	The **Preview** shows the merge result with the currently selected values.

3. This step depends on your data structure and configuration settings:

- If this merge will not impact other tables, or your configuration is set to ignore relationship dependencies, the add-on presents you with the **Preview** page. See the next step for more information.

- If other tables can be impacted by this merge, you are presented with a page to select the relationships to update. The add-on pre-selects values based on the selected golden record when entering the **Merge view**. If you do not have sufficient permissions to display the dependency, the add-on automatically makes the selection based on the record considered the golden record.

4. When you reach the **Preview** step, the add-on provides you an overview of what the merged record will look like. Select **Merge** to complete the process, or use the navigation dropdown to access previous steps to make changes.

   > **Note**
   >
   > You cannot update records that are in the **Merged** or **Deleted** states.

CHAPTER **32**

Reversing a merge operation

You can rollback a merge operation using the **Unmerge** service. This service reverts the last merge operation for the selected golden or merged records. Records will remain duplicate.

To reverse a merge operation:

> **Note**
>
> You can perfrom the same action described below from the **Manage group** screen.

1. Select either a golden or merged record.
2. From the table's **Actions** menu, select *Match and Merge > Unmerge*.
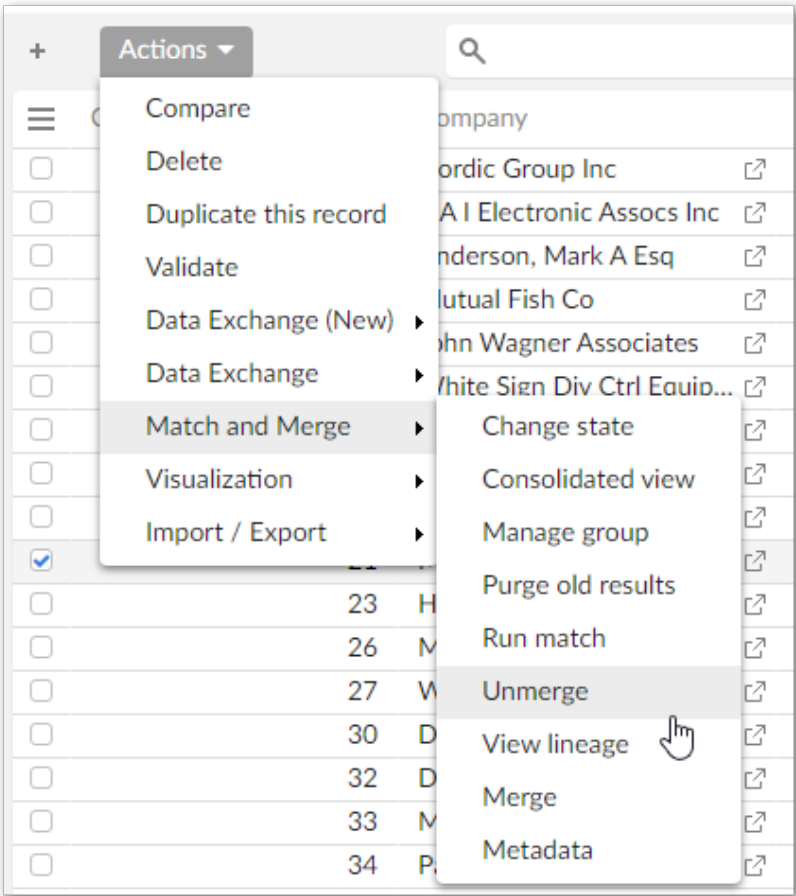
CHAPTER **33**

Consolidated view

The **Consolidated view** allows you to view add-on related metadata applied to a table. Note that records only display here if they have matching related metatdata.

The following image and list describe features and available actions:
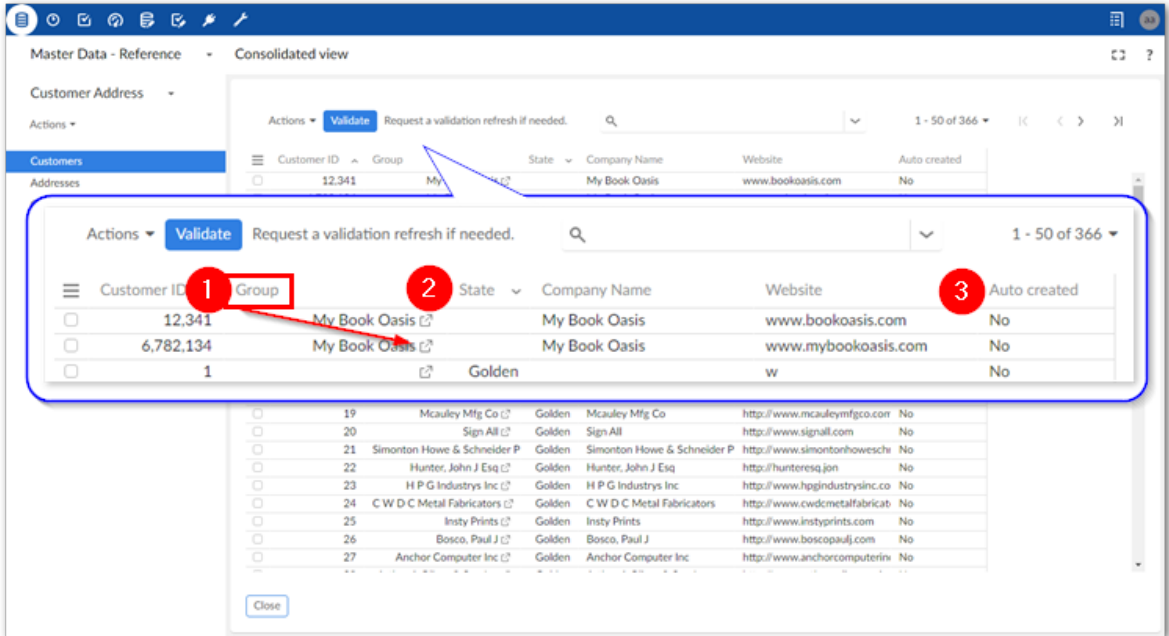
1) Group and Manage group link	The add-on groups records positively identified as matching records. This field identifies which group each record has been placed in.
	Select the icon next to a group to open the **Manage groups** screen. This screen allows you to compare records and confirm or reject records as duplicates.
2) State	When a record has taken part in a match or merge operation, the add-on assigns it state. The state identifies the result of these operations.
3) Auto created	Indicates whether the corresponding record was automatically created during match or merge operation.

CHAPTER **34**

Changing record state

When the add-on completes match and merge operations, it sets a state for each evaluated record. Possible states include:

- Golden: The singular record that most completely defines a business entity.

- Merged: A record that has been included in a merge operation, but does not meet the requirements for a golden record.

- Unset: A record that has null matching state.

- Pending approval: after an unmerge operation, all records that were part of the initial merge operation are set to pending approval.

- Deleted: Auto-created golden records that are no longer targetd by other records, or were merged with other records are deleted.

If for some reason you want to change a record's state, you can use the **Change state** service. To use this service:

1. Select one or more records to change their state.

2. From the table's **Actions** menu, select *Match and Merge > Change state*.
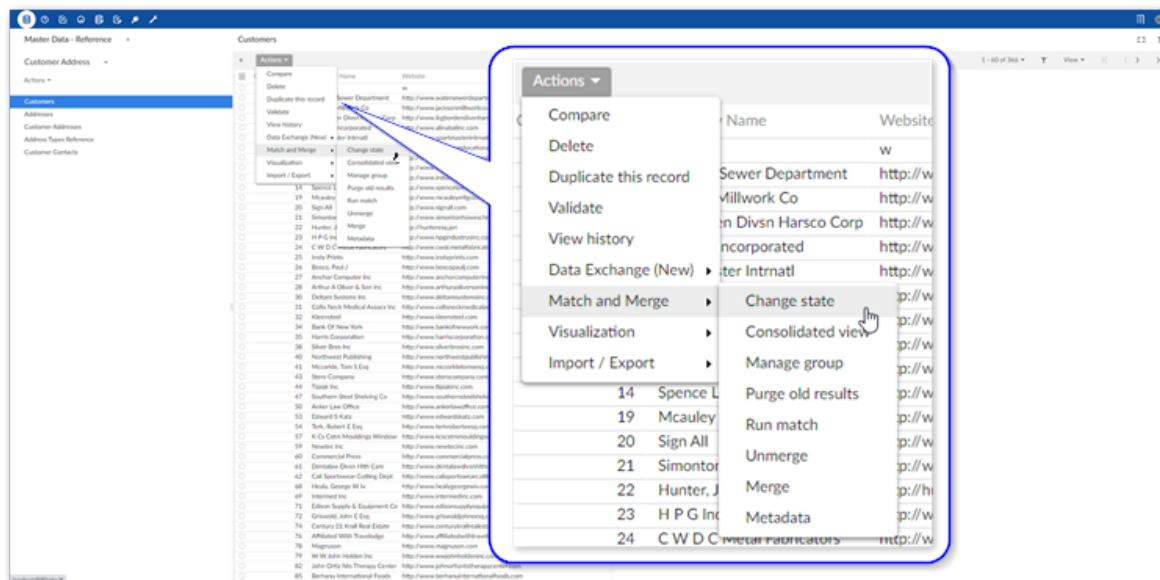
3. Choose the target state and select **Launch service** at the bottom of the screen.

> **Note**
>
> You cannot change the state for records that are in the **Deleted** state.

CHAPTER **35**

Viewing add-on metadata

This chapter contains the following topics:

1. Opening the view
2. Viewing golden record lineage

35.1 **Opening the view**

The add-on uses metadata to track records where the add-on is active. The metadata includes information about record state, groups, and target records. Additionally, the metadata tracks matching execution. To view record metadata:

1. Select a record and from the **Actions** menu, select *Match and Merge > Metadata*. As shown in the images below you can access the view from several locations:

   Table view:
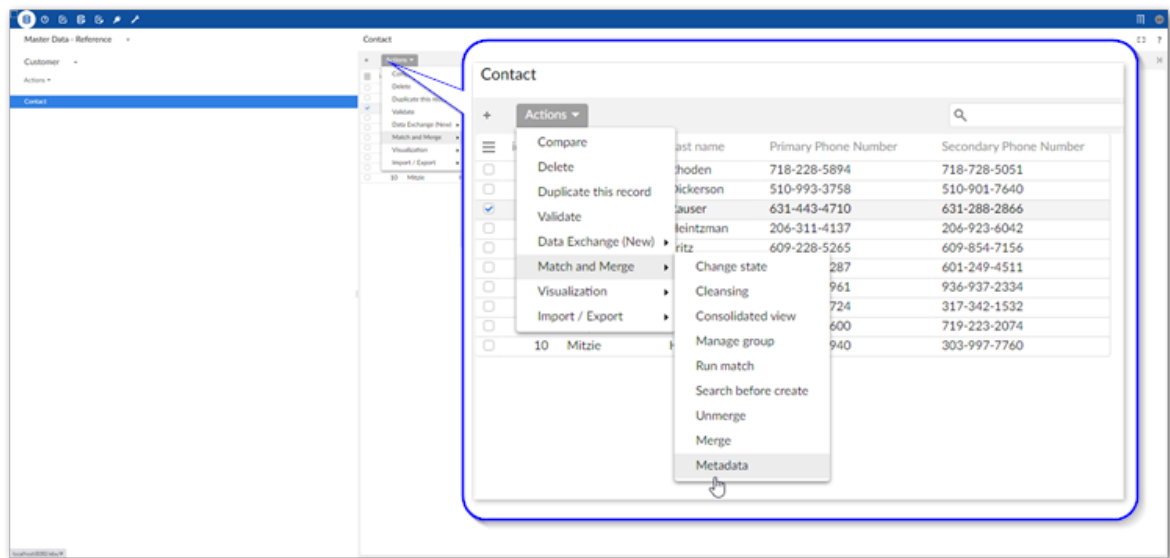
Record details view:
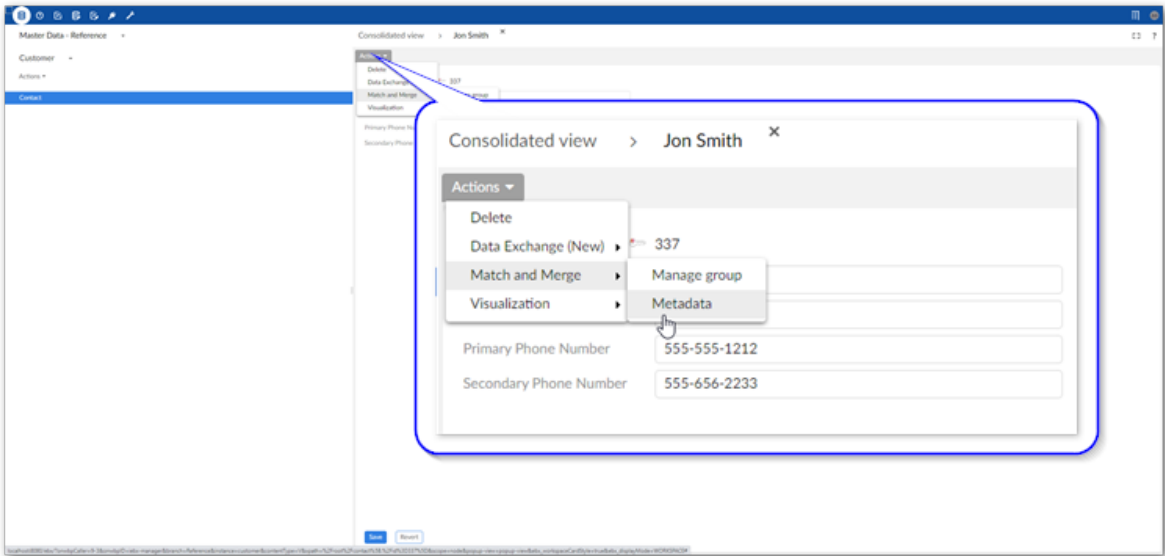
Stewardship screens such as **Manage group**:
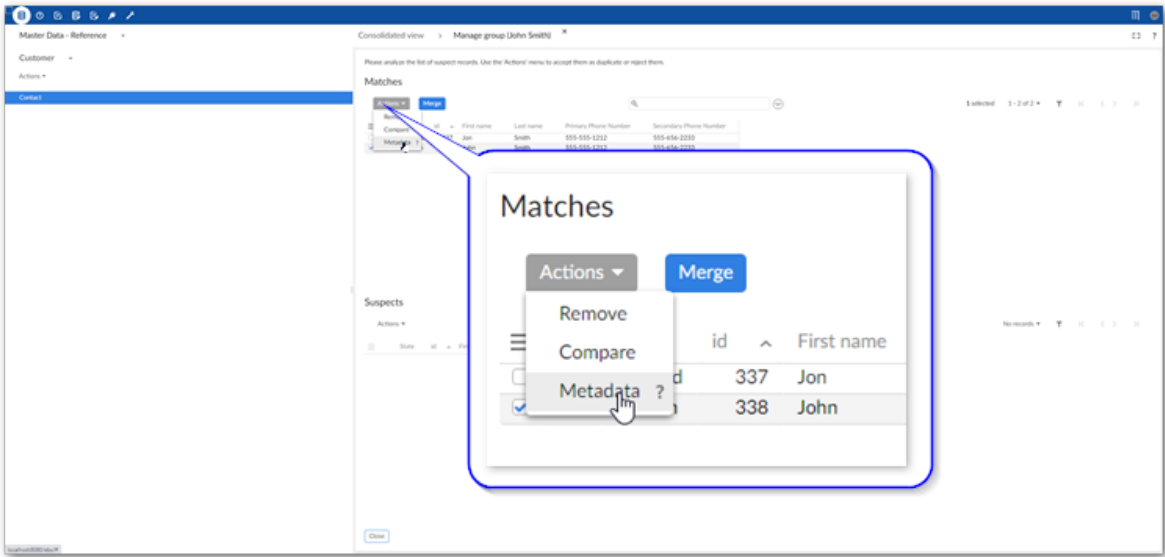
2. From the metadata screen you can perform the actions described below:

1) Execution log, Lineage, and Identified as false positives tabs	These tabs contain the following information: • **Execution log:** Displays the most recent matching execution date and time. Additionally, it shows the most recent stewardship decision, date and time, and user that performed the action. • **Lineage:** Shows any lineage related to a golden record including values from records that were merged into this record. • **Identified as false positives:** Lists any records that were rejected as suspects or matches with this record. The false positive records will no longer be grouped with this record in future matching operations.
2) Record icon	Use this icon to open the record in edit mode.
3) Group icon	Select this icon to open the **Manage group** screen. You can confirm suspect records from this screen.

35.2 **Viewing golden record lineage**

The **Metadata** screen contains the **Lineage** tab which allows you to view the values from records that were merged into a golden record. To ensure the most accurate lineage is tracked, you must enable EBX® history in the Data Model Assistant.

> **Note**
>
> You can access the **Lineage** tab directly from a table. Open the table's **Actions** menu and select **View lineage**.
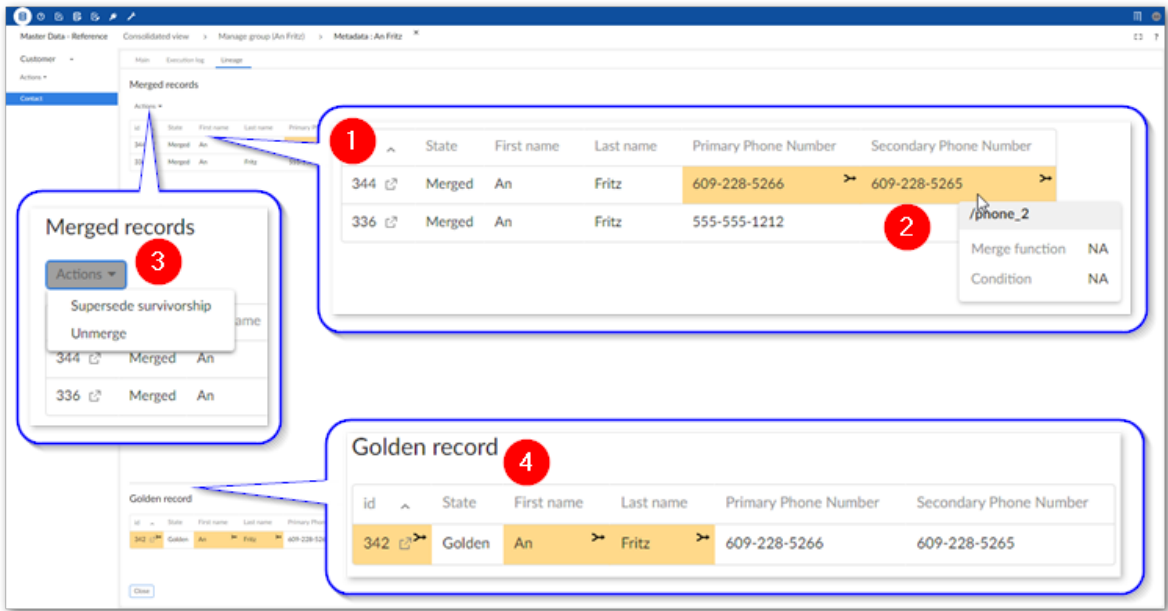
1) **Merged records**	The list of merged records includes any records that were part of a merge operation to create the golden record. This is true even if no values from a given record were merged into the golden record.
2) **Highlighted values**	These values were merged into the golden record. Hovering your mouse over a highlighted value shows the merge function and condition used to determine that this value should be merged into the golden.
3) **Actions menu options**	The **Actions** menu in the **Lineage** tab includes the following options: • **Supersede survivorship:** This allows you to override merge decisions to change, without unmerging, values in the golden record. For more information, see Superseding survivorship [p 121].

	• **Unmerge:** This service is like performing an undo action on the most recent merge operation that created the selected golden record.
4) Golden record and highlighted values	This group shows the golden record that resulted from a merge. Highlighted values are those that survived from the original record. In the example shown, the golden record kept its original values from the id, First name, and Last name fields.

Superseding survivorship

In some instances incorrect values are merged into golden records. When this happens, you can run the **Supersede survivorship** service to change the values in the golden record. This service allows you to reverse merge decisions on specific attributes without using the **Unmerge** service to completely revert the merge process.

Depending on how an administrator has configured trusted source options, the decision you make to override a previous merge can impact future automatic merges, or merge recommendations during manual merge. For example, you might override a value so that it comes from an HR system over an IT system. The next time the record is involved in a merge operation, the add-on can automatically prioritize the value coming from the HR system. If you are in an administrative role and would like to learn more about using and configuring trusted sources, see Setting trusted sources [p 45].

To supersede surviviorship:

1. Open the **Metadata** screen for a golden record using one of the following methods:

   1. From a table view, select a golden record and from the **Actions** menu, select *Match and Merge > Metadata*.

   2. From the **Manage group** view, select a golden record and from the **Actions** menu, select **Metadata**.

3. Alternatively, open a golden record from the **Manage group** view and select **Metadata** from the **Actions** menu.
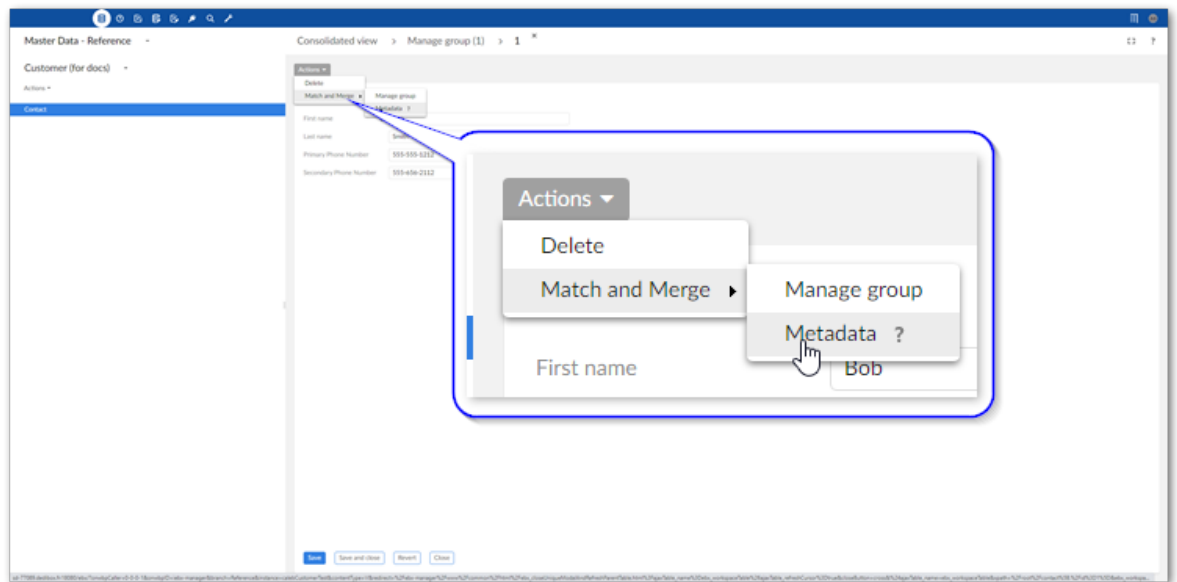
2. Run the **Supersede survivorship** service:

   1. Select the **Lineage** tab.

   2. From the **Actions** menu, select **Supersede survivorship**.
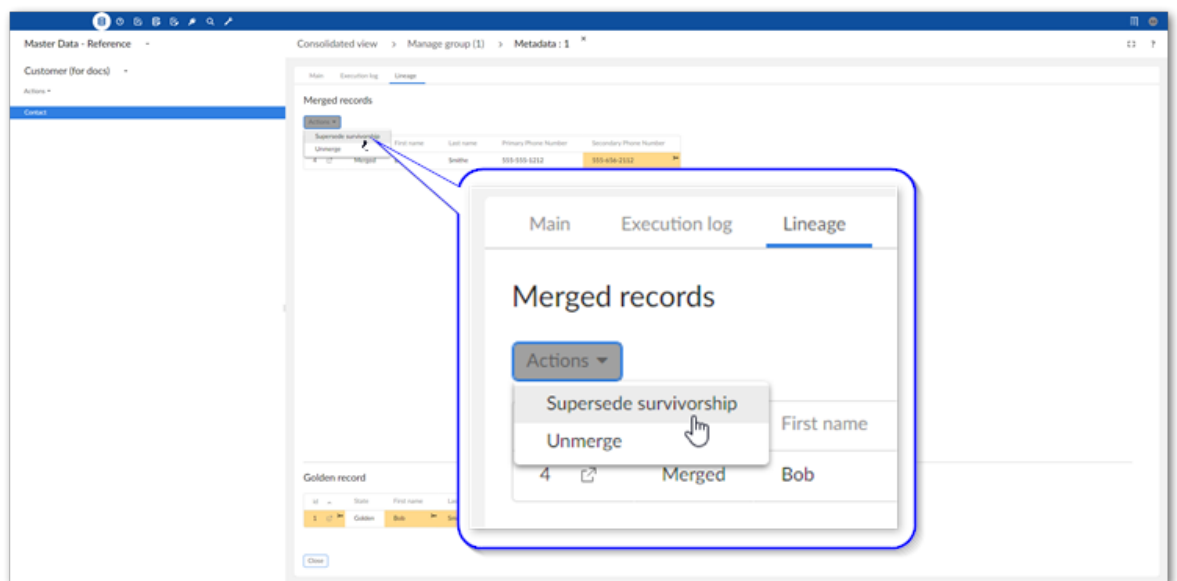
3. Select the values to survive into the golden record. Selected values are shaded in gray.

4. Select **Next**.

5. Double-check the results and select **Supersede** to perform the operation and view the results page.

Release Notes

CHAPTER **36**

Version 3.4.1

Released: July2022

This chapter contains the following topics:

1. New features
2. Changes in Functionality
3. Changes to third-party libraries
4. Closed Issues
5. Known Issues

36.1 **New features**

This release contains no new features.

Attention

When upgrading from version 3.0.0, 3.1.0, or 3.1.1 you must update existing matching configurations. After updating configuration settings, you should re-execute matching to ensure that matching related metadata is up to date.

36.2 **Changes in Functionality**

This release contains no changes in functionality.

36.3 **Changes to third-party libraries**

This release contains no changes to third-party libraries.

36.4 **Closed Issues**

[MAME-7715] The merge process is slow when changing the record state to `Unset`.

36.5 **Known Issues**

This release contains the following known issues:

- **[MAME-7434]** Linked records are duplicated in the link table when one of the foreign keys is not part of the primary key.

- When matching using the foreign key option, you can only specify a single hop foreign key. In other words, it cannot be a foreign key to a foreign key.

- The **Validation** service in the **Consolidated** view runs as expected when first executed, but slow on subsequent executions.

- In order to accurately track record lineage, table history must remain activated over time. If the table's activation status repeatedly changes, the lineage data will be inconsistent.

- Severe errors occur when multiple datasets based on the same model exist and a table from one of these datasets is activated in an add-on matching configuration. As a workaround, you can create another data model publication using the **Manage publications** service in the Data Model Assistant. After creating a dataset for the new publication, you can follow the normal procedures to configure a matching policy for tables in the new dataset.

- For foreign key alignment, the FKs that are multi-value fields cannot be aligned (either manually or automatically).

CHAPTER **37**

All Release Notes

This chapter contains the following topics:

37.1 **Version 3.4.1**

Released: July2022

New features

This release contains no new features.

> **Attention**
>
> When upgrading from version 3.0.0, 3.1.0, or 3.1.1 you must update existing matching configurations. After updating configuration settings, you should re-execute matching to ensure that matching related metadata is up to date.

Changes in Functionality

This release contains no changes in functionality.

Changes to third-party libraries

This release contains no changes to third-party libraries.

Closed Issues

[MAME-7715] The merge process is slow when changing the record state to `Unset`.

Known Issues

This release contains the following known issues:

- **[MAME-7434]** Linked records are duplicated in the link table when one of the foreign keys is not part of the primary key.

- When matching using the foreign key option, you can only specify a single hop foreign key. In other words, it cannot be a foreign key to a foreign key.

- The **Validation** service in the **Consolidated** view runs as expected when first executed, but slow on subsequent executions.

- In order to accurately track record lineage, table history must remain activated over time. If the table's activation status repeatedly changes, the lineage data will be inconsistent.

- Severe errors occur when multiple datasets based on the same model exist and a table from one of these datasets is activated in an add-on matching configuration. As a workaround, you can create another data model publication using the **Manage publications** service in the Data Model Assistant. After creating a dataset for the new publication, you can follow the normal procedures to configure a matching policy for tables in the new dataset.

- For foreign key alignment, the FKs that are multi-value fields cannot be aligned (either manually or automatically).

37.2 **Version 3.4.0**

Released: June 2022

New features

This release contains the following new features:

- The following algorithms were added:

  - Phonetic full text: An algorithm best used for strings. It can recognize the phonetic equivalent of two words spelled differently using the Beider-morse phonetic tokens. The comparator takes into account the synonyms and stop words defined in the data model.

  - Hybrid fuzzy: An algorithm best used for string and text data types. It can capture complex relationships between two field values by automatically customizing transformations for a given criterion. Points of note about this algorithm:

    - Does not return a 100% matching score on synonyms.

- Strings that are similar, but contain characters like spaces or dashes do not return a 0% matching score.

- When strings fall within the acceptable distance score they are not considered as the exact same. For example, a distance of 1 returns .95, a distance of 2 returns .9, etc.

- A script task was added to run a matching operation from a workflow.

- Fields can now have a weight of 0. Setting the weight to 0 excludes the fields from the pre-processing phase.

- An API was added to check whether the add-on has any ongoing background operations.

Attention

When upgrading from version 3.0.0, 3.1.0, or 3.1.1 you must update existing matching configurations. After updating configuration settings, you should re-execute matching to ensure that matching related metadata is up to date.

Changes in Functionality

The options to specify what happens after a merged record is modified were removed. Now, the following behavior is applied when a record is in the merged state and the modified field:

- Is *not* configured to use in matching operations: The add-on executes an automatic merge using the settings defined in the default matching policy's merge policy.

- Is configured to use in matching operations: The record is changed to the <unset> state.

Changes to third-party libraries

This release contains no changes to third-party libraries.

Closed Issues

This release contains the following closed issues:

- **[MAME-7400]** Changing record state to `Unset` fails when many transactions are committed in a short period of time.

- **[MAME-7455]** The **Evaluate matching** and **Run match** services return inconsistent results.

- **[MAME-7479]** There is conflict between the **Evaluate matching** and **See details** results.

- **[MAME-7542]** Results cannot display after running **Evaluate matching** on a display with 1920X1080 resolution at 150% zoom.

Known Issues

This release contains the following known issues:

- **[MAME-7434]** Linked records are duplicated in the link table when one of the foreign keys is not part of the primary key.

- When matching using the foreign key option, you can only specify a single hop foreign key. In other words, it cannot be a foreign key to a foreign key.

- The **Validation** service in the **Consolidated** view runs as expected when first executed, but slow on subsequent executions.

- In order to accurately track record lineage, table history must remain activated over time. If the table's activation status repeatedly changes, the lineage data will be inconsistent.

- Severe errors occur when multiple datasets based on the same model exist and a table from one of these datasets is activated in an add-on matching configuration. As a workaround, you can create another data model publication using the **Manage publications** service in the Data Model Assistant. After creating a dataset for the new publication, you can follow the normal procedures to configure a matching policy for tables in the new dataset.

- For foreign key alignment, the FKs that are multi-value fields cannot be aligned (either manually or automatically).

37.3 **Version 3.3.5**

Released: June 2022

New features

This release contains no new features.

> **Attention**
>
> When upgrading from version 3.0.0, 3.1.0, or 3.1.1 you must update existing matching configurations. After updating configuration settings, you should re-execute matching to ensure that matching related metadata is up to date.

Changes in Functionality

This release contains no changes in functionality.

Changes to third-party libraries

This release contains no changes to third-party libraries.

Closed Issues

[MAME-7532] A performance issue occurs during the grouping phase when running matching on an entire table.

Known Issues

This release contains the following known issues:

- When matching using the foreign key option, you can only specify a single hop foreign key. In other words, it cannot be a foreign key to a foreign key.

- The **Validation** service in the **Consolidated** view runs as expected when first executed, but slow on subsequent executions.

- In order to accurately track record lineage, table history must remain activated over time. If the table's activation status repeatedly changes, the lineage data will be inconsistent.

- Severe errors occur when multiple datasets based on the same model exist and a table from one of these datasets is activated in an add-on matching configuration. As a workaround, you can create another data model publication using the **Manage publications** service in the Data Model

Assistant. After creating a dataset for the new publication, you can follow the normal procedures to configure a matching policy for tables in the new dataset.

- For foreign key alignment, the FKs that are multi-value fields cannot be aligned (either manually or automatically).

37.4 **Version 3.3.4**

Released: May 2022

New features

This release contains no new features.

Attention

When upgrading from version 3.0.0, 3.1.0, or 3.1.1 you must update existing matching configurations. After updating configuration settings, you should re-execute matching to ensure that matching related metadata is up to date.

Changes in Functionality

This release contains no changes in functionality.

Changes to third-party libraries

The Spring framework was updated to version 5.2.22.

Closed Issues

This release contains no closed issues.

Known Issues

This release contains the following known issues:

- When matching using the foreign key option, you can only specify a single hop foreign key. In other words, it cannot be a foreign key to a foreign key.

- The **Validation** service in the **Consolidated** view runs as expected when first executed, but slow on subsequent executions.

- In order to accurately track record lineage, table history must remain activated over time. If the table's activation status repeatedly changes, the lineage data will be inconsistent.

- Severe errors occur when multiple datasets based on the same model exist and a table from one of these datasets is activated in an add-on matching configuration. As a workaround, you can create another data model publication using the **Manage publications** service in the Data Model Assistant. After creating a dataset for the new publication, you can follow the normal procedures to configure a matching policy for tables in the new dataset.

- For foreign key alignment, the FKs that are multi-value fields cannot be aligned (either manually or automatically).

37.5 **Version 3.3.3**

Released: April 2022

New features

This release contains no new features.

> **Attention**
>
> When upgrading from version 3.0.0, 3.1.0, or 3.1.1 you must update existing matching configurations. After updating configuration settings, you should re-execute matching to ensure that matching related metadata is up to date.

Changes in Functionality

This release contains no changes in functionality.

Changes to third-party libraries

This release contains no changes to third-party libraries.

Closed Issues

This release contains the following closed issues:

- **[MAME-7397]** The **Merge view** and the **Lineage view** cannot be accessed when all fields in a group or table are computed fields.
- **[MAME-7398]** Performance is slow when fetching values n to n relationships.
- **[MAME-7436]** When running the **Evaluate matching** service the field values do not display correctly when using the same field identities, but in different tables.
- **[MAME-7407]** Boolean fields cannot be merged when **Check null input** is enabled.

Known Issues

This release contains the following known issues:

- When matching using the foreign key option, you can only specify a single hop foreign key. In other words, it cannot be a foreign key to a foreign key.
- The **Validation** service in the **Consolidated** view runs as expected when first executed, but slow on subsequent executions.
- In order to accurately track record lineage, table history must remain activated over time. If the table's activation status repeatedly changes, the lineage data will be inconsistent.
- Severe errors occur when multiple datasets based on the same model exist and a table from one of these datasets is activated in an add-on matching configuration. As a workaround, you can create another data model publication using the **Manage publications** service in the Data Model Assistant. After creating a dataset for the new publication, you can follow the normal procedures to configure a matching policy for tables in the new dataset.

- For foreign key alignment, the FKs that are multi-value fields cannot be aligned (either manually or automatically).

37.6 **Version 3.3.2**

Released: April 2022

New features

This release contains no new features.

> **Attention**
>
> When upgrading from version 3.0.0, 3.1.0, or 3.1.1 you must update existing matching configurations. After updating configuration settings, you should re-execute matching to ensure that matching related metadata is up to date.

Changes in Functionality

This release contains no changes in functionality.

Changes to third-party libraries

The Spring framework was updated to version 5.2.20.

Closed Issues

This release contains no closed issues.

Known Issues

This release contains the following known issues:

- When matching using the foreign key option, you can only specify a single hop foreign key. In other words, it cannot be a foreign key to a foreign key.

- The **Validation** service in the **Consolidated** view runs as expected when first executed, but slow on subsequent executions.

- In order to accurately track record lineage, table history must remain activated over time. If the table's activation status repeatedly changes, the lineage data will be inconsistent.

- Severe errors occur when multiple datasets based on the same model exist and a table from one of these datasets is activated in an add-on matching configuration. As a workaround, you can create another data model publication using the **Manage publications** service in the Data Model Assistant. After creating a dataset for the new publication, you can follow the normal procedures to configure a matching policy for tables in the new dataset.

- For foreign key alignment, the FKs that are multi-value fields cannot be aligned (either manually or automatically).

37.7 **Version 3.3.1**

Released: March 2022

New features

Performance was improved to retrieve groups containing suspect records.

Attention

When upgrading from version 3.0.0, 3.1.0, or 3.1.1 you must update existing matching configurations. After updating configuration settings, you should re-execute matching to ensure that matching related metadata is up to date.

Changes in Functionality

This release contains no changes in functionality.

Changes to third-party libraries

This release contains the following changes to third-party libraries:

- The Spring framework was updated to version 5.2.19.
- The FasterXML/jackson-databind library was updated to version 2.13.2.1.

Closed Issues

This release contains no closed issues.

Known Issues

This release contains the following known issues:

- When matching using the foreign key option, you can only specify a single hop foreign key. In other words, it cannot be a foreign key to a foreign key.
- The **Validation** service in the **Consolidated** view runs as expected when first executed, but slow on subsequent executions.
- In order to accurately track record lineage, table history must remain activated over time. If the table's activation status repeatedly changes, the lineage data will be inconsistent.
- Severe errors occur when multiple datasets based on the same model exist and a table from one of these datasets is activated in an add-on matching configuration. As a workaround, you can create another data model publication using the **Manage publications** service in the Data Model Assistant. After creating a dataset for the new publication, you can follow the normal procedures to configure a matching policy for tables in the new dataset.
- For foreign key alignment, the FKs that are multi-value fields cannot be aligned (either manually or automatically).

37.8 **Version 3.3.0**

Released: February 2022

New features

This release includes the following new features:

- You can now replicate matching metadata. This allows you to use third-party tools to access and query the metadata.
- The following algorithms are now available:
  - Fuzzy full text: This algorithm works best for strings and can find the similar and fuzzy matches of the words in the compared values. It is based on the Levenshtein distance algorithm. The comparator takes into account the synonyms and stop words defined in the data model.
  - Range: This algorithm matches values within a predefined range. The two values are considered a match if the distance between two values is within the specified range.

> **Attention**
>
> When upgrading from version 3.0.0, 3.1.0, or 3.1.1 you must update existing matching configurations. After updating configuration settings, you should re-execute matching to ensure that matching related metadata is up to date.

Changes in Functionality

This release contains the following changes in functionality:

- When **Auto create new golden** is enabled, the requirement was removed for the matching table's primary key to be an auto-incremented integer.
- When executing add-on operations, only the add-on's table triggers are disabled. All user custom triggers remain enabled. Note that table triggers related to metadata linked fields are not recommended as it might affect the accuracy of match and merge operations.

Changes to third-party libraries

This release contains no changes to third party libraries.

Closed Issues

When matching is set to **After submission** and batch creation of records occurs using the EBX® API, the number of submitted or created records is limited to 2000 records in an API call for tables that have a single primary key. For a composite primary the limit is divided by the number of fields that makes up the primary key. For example, with a table that has a primary key comprised of 2 fields, the limit would be 1000 records. When your requirements exceed these limits, it is recommended that you pause the matching trigger, load the data, and resume matching. Also, note that when matching is triggered the dataspace is locked until matching completes. Any incoming requests during this time are blocked and will return errors.

Known Issues

This release contains the following known issues:

- When matching using the foreign key option, you can only specify a single hop foreign key. In other words, it cannot be a foreign key to a foreign key.
- The **Validation** service in the **Consolidated** view runs as expected when first executed, but slow on subsequent executions.

- In order to accurately track record lineage, table history must remain activated over time. If the table's activation status repeatedly changes, the lineage data will be inconsistent.

- Severe errors occur when multiple datasets based on the same model exist and a table from one of these datasets is activated in an add-on matching configuration. As a workaround, you can create another data model publication using the **Manage publications** service in the Data Model Assistant. After creating a dataset for the new publication, you can follow the normal procedures to configure a matching policy for tables in the new dataset.

- For foreign key alignment, the FKs that are multi-value fields cannot be aligned (either manually or automatically).

37.9 **Version 3.2.1**

Released: December 2021

New features

This release includes no new features.

> **Attention**
>
> When upgrading from version 3.0.0, 3.1.0, or 3.1.1 you must update existing matching configurations. After updating configuration settings, you should re-execute matching to ensure that matching related metadata is up to date.

Changes in Functionality

The Apache Log4j library was upgraded to version 2.17.1.

Changes to third-party libraries

This release contains no changes to third party libraries.

Closed Issues

This release contains no closed issues.

Known Issues

This release contains the following known issues:

- When matching using the foreign key option, you can only specify a single hop foreign key. In other words, it cannot be a foreign key to a foreign key.

- The **Validation** service in the **Consolidated** view runs as expected when first executed, but slow on subsequent executions.

- In order to accurately track record lineage, table history must remain activated over time. If the table's activation status repeatedly changes, the lineage data will be inconsistent.

- Severe errors occur when multiple datasets based on the same model exist and a table from one of these datasets is activated in an add-on matching configuration. As a workaround, you can create another data model publication using the **Manage publications** service in the Data Model

Assistant. After creating a dataset for the new publication, you can follow the normal procedures to configure a matching policy for tables in the new dataset.

- When matching is set to **After submission** and batch creation of records occurs using the EBX® API, the number of submitted or created records is limited to 2000 records in an API call for tables that have a single primary key. For a composite primary the limit is divided by the number of fields that makes up the primary key. For example, with a table that has a primary key comprised of 2 fields, the limit would be 1000 records. When your requirements exceed these limits, it is recommended that you pause the matching trigger, load the data, and resume matching. Also, note that when matching is triggered the dataspace is locked until matching completes. Any incoming requests during this time are blocked and will return errors.

37.10 **Version 3.2.0**

Released: November 2021

New features

This release includes the following new features:

- You can now exclude records from matching operations by specifying a field value. If a record contains the specified value, it is not included in matching operations. Using this functionality can help improve performance and improve the quality of matching results.

- You can now modify records that were included in a merge operation.

- The new **Evaluate matching** service allows you to test matching results. This can help you fine tune configuration settings to achieve desired results.

- Adding an association to your data model allows you to view: a hierarchy of golden and merged records, and a tab when viewing golden record details that displays all merged records that target the golden.

- Matching now supports use of non-Latin based text.

- A new merge function on field survivorship allows you to add a constant value that is merged into golden records on survivorship.

- The Full Text algorithm is now available to use in matching for fields with Text and String data types.

- You can now use a foreign key field as a field in a decision tree's comparison node.

Attention

When upgrading from version 3.0.0, you must update existing matching configurations. After updating configuration settings, you should re-execute matching to ensure that matching related metadata is up to date.

Changes in Functionality

This release contains the following functionality changes:

- The default mode for **Align foreign keys** is now set to **Manually**.

- The **Case sensitivity** option is now only available for fields with String and Text data types.

Changes to third-party libraries

This release contains no changes to third party libraries.

Closed Issues

This release contains no closed issues.

Known Issues

This release contains the following known issues:

- When matching using the foreign key option, you can only specify a single hop foreign key. In other words, it cannot be a foreign key to a foreign key.

- The **Validation** service in the **Consolidated** view runs as expected when first executed, but slow on subsequent executions.

- In order to accurately track record lineage, table history must remain activated over time. If the table's activation status repeatedly changes, the lineage data will be inconsistent.

- Severe errors occur when multiple datasets based on the same model exist and a table from one of these datasets is activated in an add-on matching configuration. As a workaround, you can create another data model publication using the **Manage publications** service in the Data Model Assistant. After creating a dataset for the new publication, you can follow the normal procedures to configure a matching policy for tables in the new dataset.

- When matching is set to **After submission** and batch creation of records occurs using the EBX® API, the number of submitted or created records is limited to 2000 records in an API call for tables that have a single primary key. For a composite primary the limit is divided by the number of fields that makes up the primary key. For example, with a table that has a primary key comprised of 2 fields, the limit would be 1000 records. When your requirements exceed these limits, it is recommended that you pause the matching trigger, load the data, and resume matching. Also, note that when matching is triggered the dataspace is locked until matching completes. Any incoming requests during this time are blocked and will return errors.

37.11 **Version 3.1.1**

Released: September 2021

New features

This release includes updates and enhancements to improve the performance of matching operations.

> **Attention**
>
> When upgrading from version 3.0.0, 3.1.0, or 3.1.1 you must update existing matching configurations. After updating configuration settings, you should re-execute matching to ensure that matching related metadata is up to date.

Changes in Functionality

This release contains no functionality changes.

Changes to third-party libraries

This release contains no changes to third party libraries.

Closed Issues

This release contains no closed issues.

Known Issues

This release contains the following known issues:

- The **Validation** service in the **Consolidated** view runs as expected when first executed, but slow on subsequent executions.

- In order to accurately track record lineage, table history must remain activated over time. If the table's activation status repeatedly changes, the lineage data will be inconsistent.

- Severe errors occur when multiple datasets based on the same model exist and a table from one of these datasets is activated in an add-on matching configuration. As a workaround, you can create another data model publication using the **Manage publications** service in the Data Model Assistant. After creating a dataset for the new publication, you can follow the normal procedures to configure a matching policy for tables in the new dataset.

- When matching is set to **After submission** and batch creation of records occurs using the EBX® API, the number of submitted or created records is limited to 2000 records in an API call for tables that have a single primary key. For a composite primary the limit is divided by the number of fields that makes up the primary key. For example, with a table that has a primary key comprised of 2 fields, the limit would be 1000 records. When your requirements exceed these limits, it is recommended that you pause the matching trigger, load the data, and resume matching. Also, note that when matching is triggered the dataspace is locked until matching completes. Any incoming requests during this time are blocked and will return errors.

37.12 **Version 3.1.0**

Released: July 2021

New features

This release includes the following new features:

- A new REST API is available to perform matching related operations. You can use Swagger to view the API documentation. See REST overview [p 74] for more information.

- Matching operations after record submission are now available for records created using the EBX® API. There are some limitations to this functionality. See the *Known issues* section below for details.

- Several matching services are now available to use in workflows and perspectives. See Perspectives and workflows [p 59] for more information.

- The add-on now supports link tables when matching with relationships. Additionally, to avoid confusion with EBX®'s concpet of associations, this feature was renamed to **Use relationships**.

- The add-on's matching algorithm library now includes the Soundex algorithm. This algorithm works well for short strings such as names. See Matching algorithms [p 63] for more information.

- The list of algorithms is now filtered based on the search strategy configured for a matching field. Additionally, the validation report will display a warning if the configured algorithm is not compatible with the field's search strategy. See Matching algorithms [p 63] for more information.

- Administrators can now set a matching policy as the default policy from the **Matching policies** tab in the **Actions** menu.

- A merge policy's record selection mode now includes the **First acquired** option to select the oldest record during survivorship.

- The **Customize source value for new golden** now supports foreign keys.

- Matching metadata can now be added to tables by creating a custom view. See Displaying matching metadata [p 61] for more details.

- The add-on's **Consolidated view** the **Actions** menu where you can execute matching related operations.

- When viewing table data, you can now access the **Metadata** view's **Lineage** tab by selecting **View lineage**.

- When a group contains only one record that does not need to be merged, a new **Set as golden** button allows you to set its state to golden.

Changes in Functionality

This release contains the following functionality changes: When using the **Run match** service to run a match on all records in a table, you must now select all records in the table instead of selecting nothing.

Changes to third-party libraries

This release contains no changes to third party libraries.

Closed Issues

This release contains no closed issues.

Known Issues

This release contains the following known issues:

- The **Validation** service in the **Consolidated** view runs as expected when first executed, but slow on subsequent executions.

- In order to accurately track record lineage, table history must remain activated over time. If the table's activation status repeatedly changes, the lineage data will be inconsistent.

- Severe errors occur when multiple datasets based on the same model exist and a table from one of these datasets is activated in an add-on matching configuration. As a workaround, you can create another data model publication using the **Manage publications** service in the Data Model Assistant. After creating a dataset for the new publication, you can follow the normal procedures to configure a matching policy for tables in the new dataset.

- When matching is set to **After submission** and batch creation of records occurs using the EBX® API, the number of submitted or created records is limited to 2000 records in an API call for tables that have a single primary key. For a composite primary the limit is divided by the number of fields that makes up the primary key. For example, with a table that has a primary key comprised of 2 fields, the limit would be 1000 records. When your requirements exceed these limits, it is

recommended that you pause the matching trigger, load the data, and resume matching. Also, note that when matching is triggered the dataspace is locked until matching completes. Any incoming requests during this time are blocked and will return errors.

37.13 **Version 3.0.1**

Released: March 2021

Bug Fixes

[MAME-5857] A documentation issue was corrected.

37.14 **Version 3.0.0**

Released: March 2021

Release overview

The TIBCO EBX® Match and Merge Add-on locates and merges duplicate data values. The goal is to obtain a singular record that most accurately defines a business entity. The add-on identifies these as *golden records*.

Features available to administrators in include the ability to:

- Make data available to the add-on by registering and activating matching on tables.

- Configure a matching policy that determines how the add-on compares data. A matching policy also specifies whether users must start the matching process manually or if it starts automatically when data is created or updated.

- Create a merge policy that establishes how duplicate data values are merged into a golden record. Merge policies can also execute automatically after a matching process runs.

- Set up trusted sources that define a hierarchy of data sources. The add-on uses the source's trust level to during matching to choose the most accurate record and during a merge to select values to merge.

Administrators can use *Administrator Guide* [p 10] to learn more about configuration options available in this release.

Features available to business users include the ability to:

- Select data and initiate matching and merge operations. Search for records that might match when creating a new record.

- Manage records that grouped together after running a matching process. The add-on groups records that are suspected of containing duplicate data or are positively identified as being duplicate. When managing groups you can also specify that a record is a false-positive match so that it no longer matches with the records in its group.

- View record metadata applied by the add-on to categorize records. Two different views allow users to view individual record, or an entire table's metadata. Some information displayed includes the record's history, lineage, state, and records identified as false positive matches.

Business users can learn more about available actions and view in the *User Guide.* [p 76]