Configuring Data Reuse and Updating

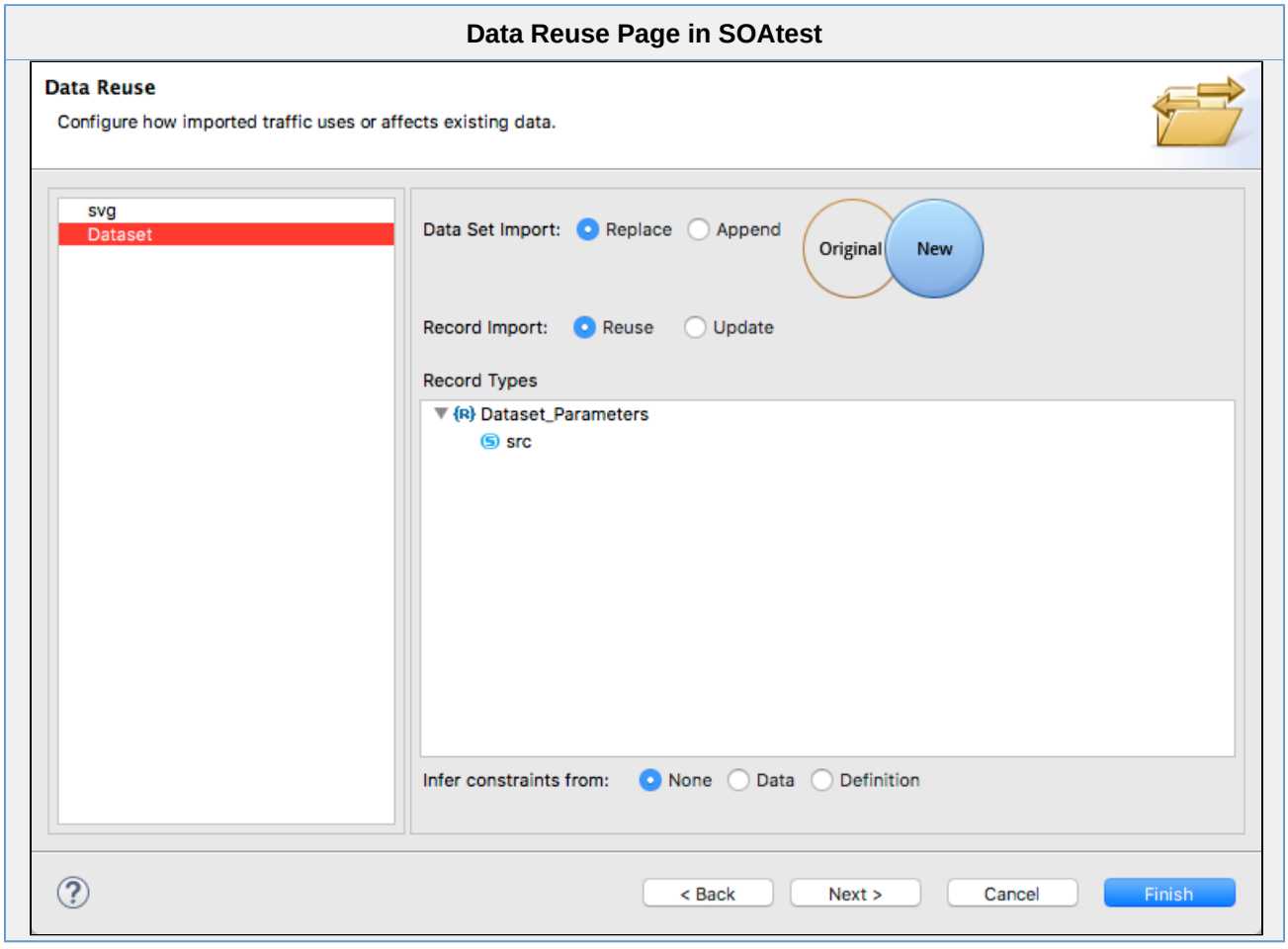
This topic provides details on completing the Data Reuse page in the "Generate Parameterized Messages from Traffic" wizard. These settings determine how new data from the traffic file will extend and/or update existing repository data sets and data records.

Sections include:

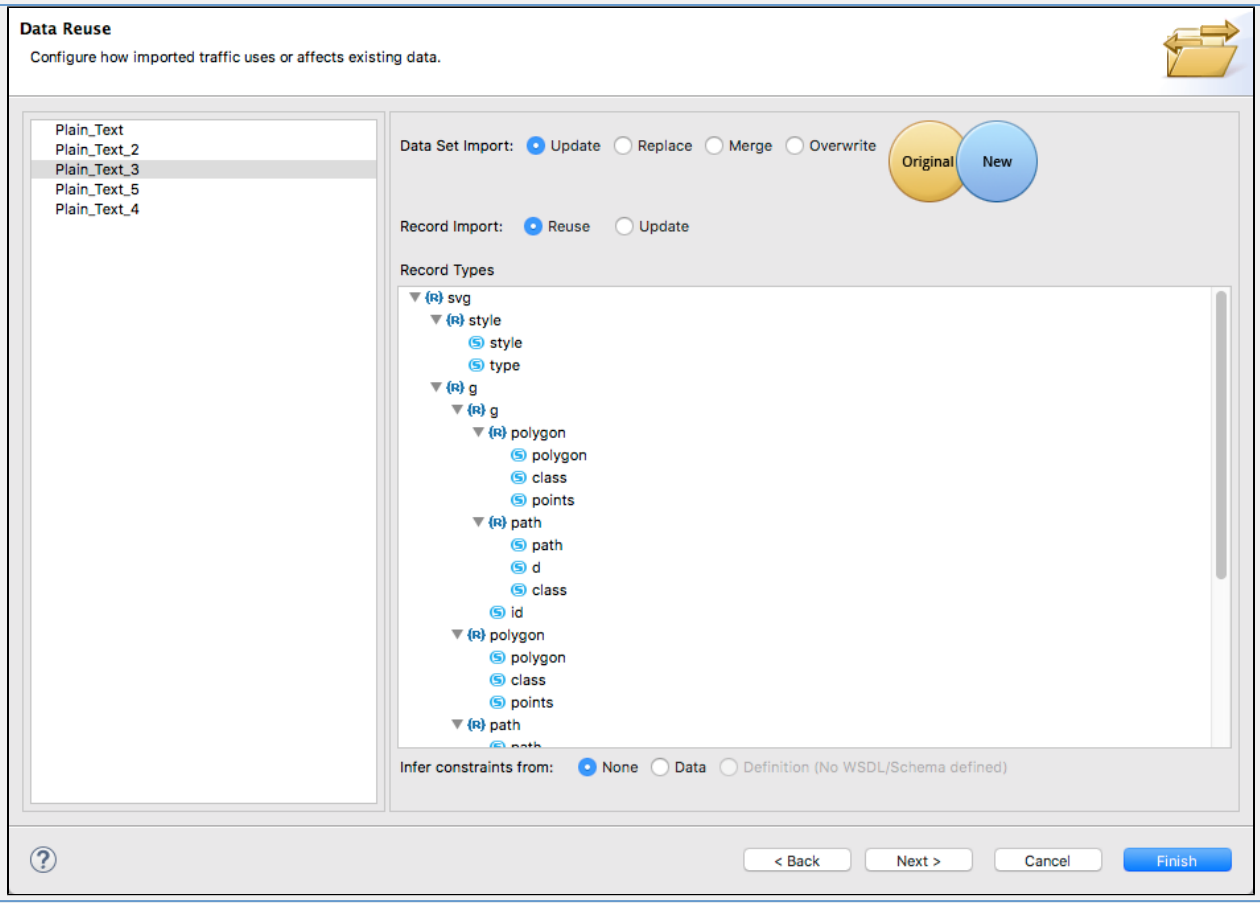
Understanding the Display of Data Sets and Record Types

The left panel displays data sets. Any data sets that are already present in the specified repository will be marked with the (**existing**) label.

Data Reuse Page in SOAtest



Data Reuse Page in Virtualize

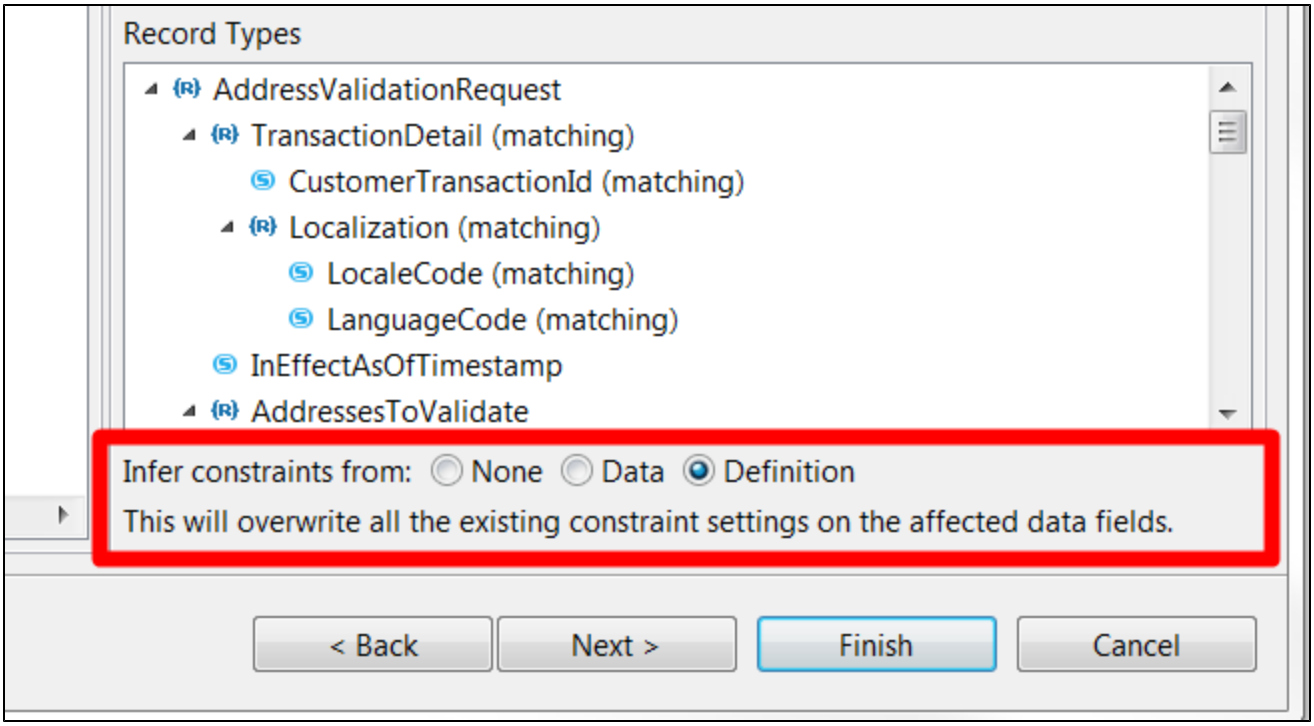


The right panel displays the record types and fields for a group. Any record types or fields that match existing data repository records will be marked with the (**matching**) label.

- **Data Set Import** options apply to the data sets listed in the left panel. They are described in Understanding Data Set Import Options.
- **Record Import** options apply to the record types shown in the right panel. They are described in Understanding Record Import Options.
- **Record Types tree** modification options are described in Modifying Record Type and Field Names and Mappings and Specifying Record Identities from the Wizard.

Inferring Constraints

If you are using CTP, you can also choose an option for inferring constraints. Constraints are additional properties associated with the data that enable you to create models and generate test data (see Data Modeling). For example, data can be constrained to strings, numbers, dates, or Booleans.



- Choose **None** if you do not want constraints to be defined.
- Choose **Data** if you want to automatically set constraints based on the data. You will have the option to populate metadata from the data into the data repository. For example, if the data set contains U.S. phone numbers, the following constraints may be set:

  ```
  Constraint:String

  Pattern:(###) ###-####

  Char map:0123456789
  ```

  See Inferring Data Constraints for additional information.

- Choose **Definition** if you want the constraints to be defined according to a service definition file (WSDL/schema). This option is disabled if you did not provide a definition file was not specified in the traffic wizard.

See Creating Parameterized Message Responders from Traffic for information on using WSDLs in Virtualize and Creating Tests From a WSDL for information on using WSDLs in SOAtest.

Existing constraints will be overwritten if you select an existing repository in the traffic wizard. You can view and modify in the constraints in CTP. See Data Modeling.

Understanding Record Identities

A record identity is the subset of a record type's fields, which uniquely identifies that record type. For example, a bank customer record type might have 15 different fields, but its identity might use only social security number and account number. Or, a book record might have ISBN as its identity.

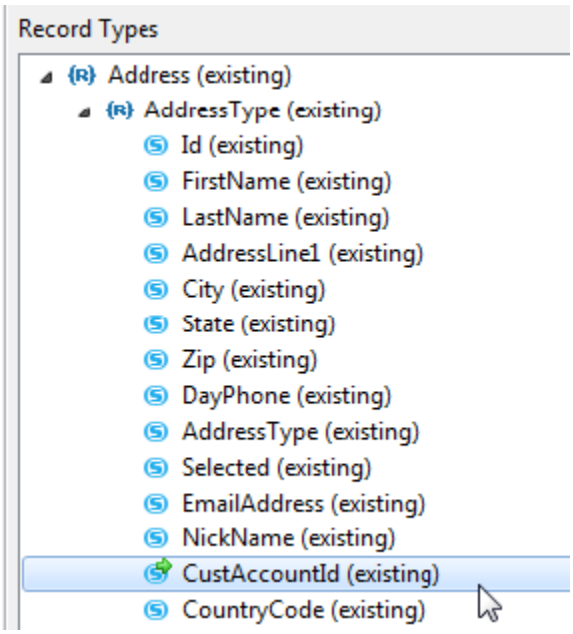
Identities enable you to correlate imported traffic data with existing data repository records. This matching helps determine which data from the traffic file is new and which match existing data. When matching data is detected, record import settings determine whether the existing record will be referenced/shared or whether it will be updated.

For more details on record identities, see Specifying Record Identities.

Specifying Record Identities from the Wizard

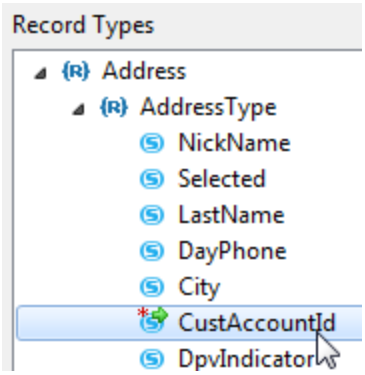
If record identities have not already been specified for the data set associated with this traffic file, they can be set in the record type tree shown in the wizard.

Fields that are already part of an identity are marked with green icons. Existing record types (annotated with (**existing**) labels in the tree) cannot have their identities adjusted in the wizard (you cannot remove identities or add new ones here). If you want to adjust identity settings on an existing data set then you need to do it using the Data Repository editor (as described in Specifying Record Identities).



If there are no existing identities and you want to specify that you want a field to be used as the identity:

- Right-click that field, then choose **Add to Identity**. The icon will change to indicate that it will become an identity upon wizard completion:



To remove a field from the identity:

- Right-click that field, then choose **Remove from Identity**.

Modifying Record Type and Field Names and Mappings

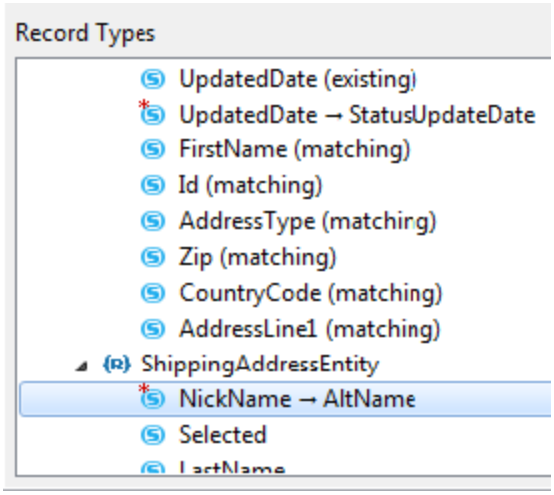
For greater control over the schema of the data being imported, you can rename record types and fields that will be created. This lets you customize names, as well as indicate that items marked with different names are actually the same.

Existing records and fields cannot be renamed, but new duplicate ones can be created.

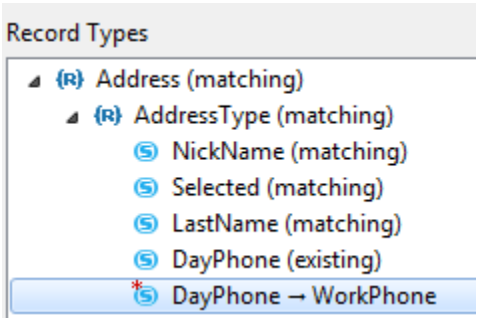
To rename a record type of field:

- Right-click the appropriate tree node, choose **Rename**, then enter the desired name.

The new name will be indicated in the tree (to the right of the arrow) and the icon will be marked with a red asterisk (*).

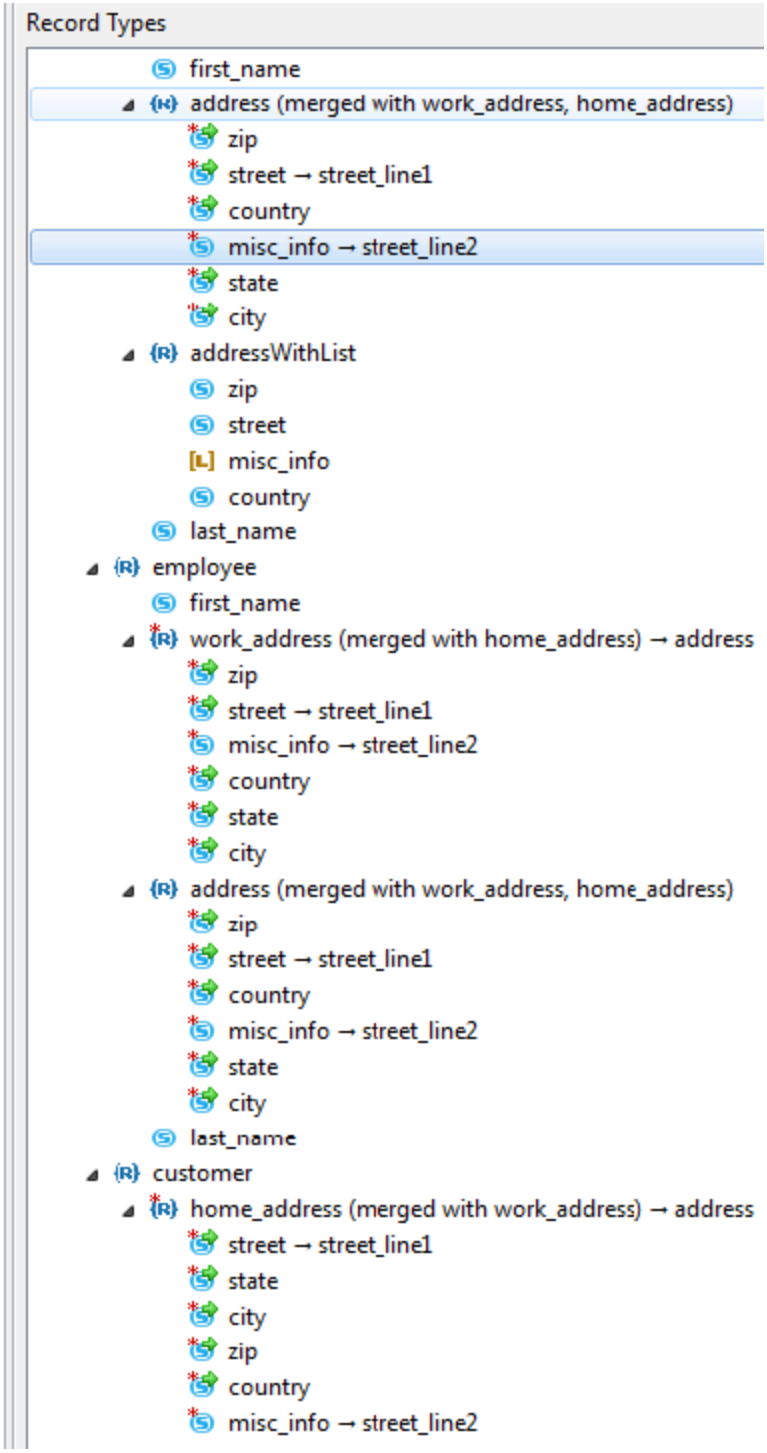


If you chose to rename an existing record type or field, note that the original entry will remain unchanged and a new one will be added immediately below it. Data from the traffic file will use the new record type or field.



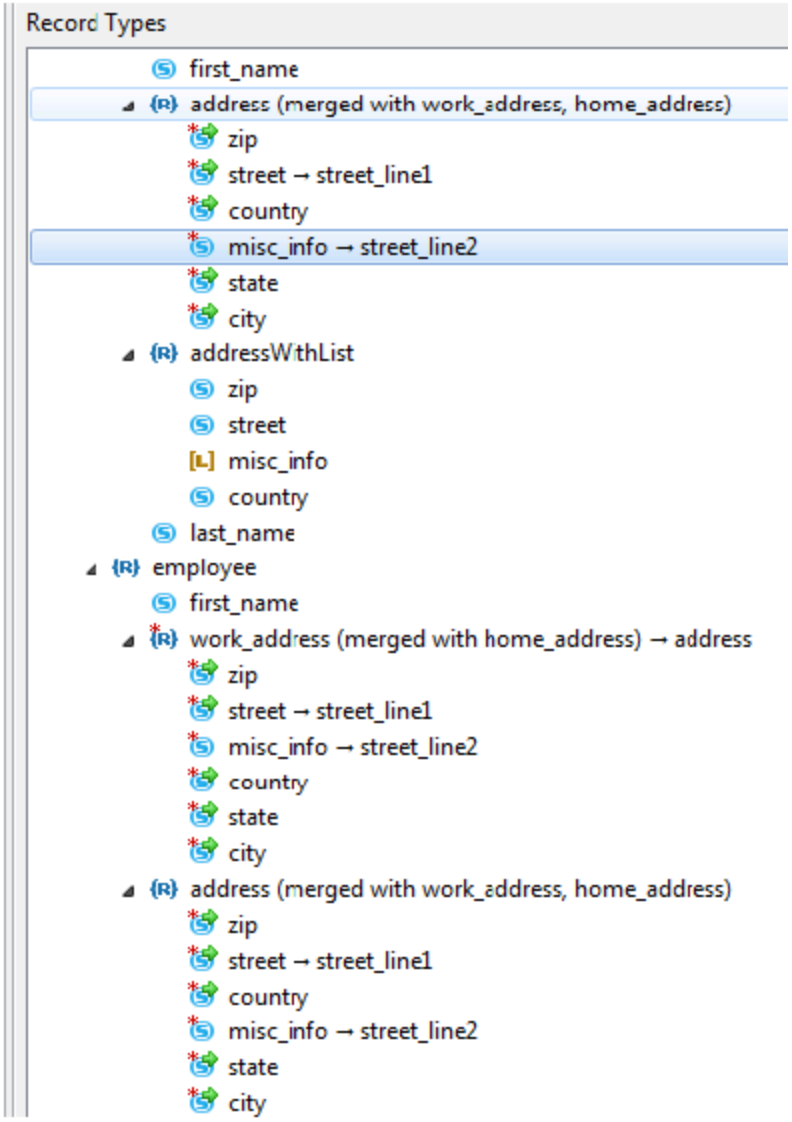
Example

Assume that the recorded traffic references multiple elements (e.g., address, work_address, home_address) that are actually of the same address type. By default, the wizard will treat them as different types (since they have different names). By renaming them, you can indicate that they should be treated as the same type.

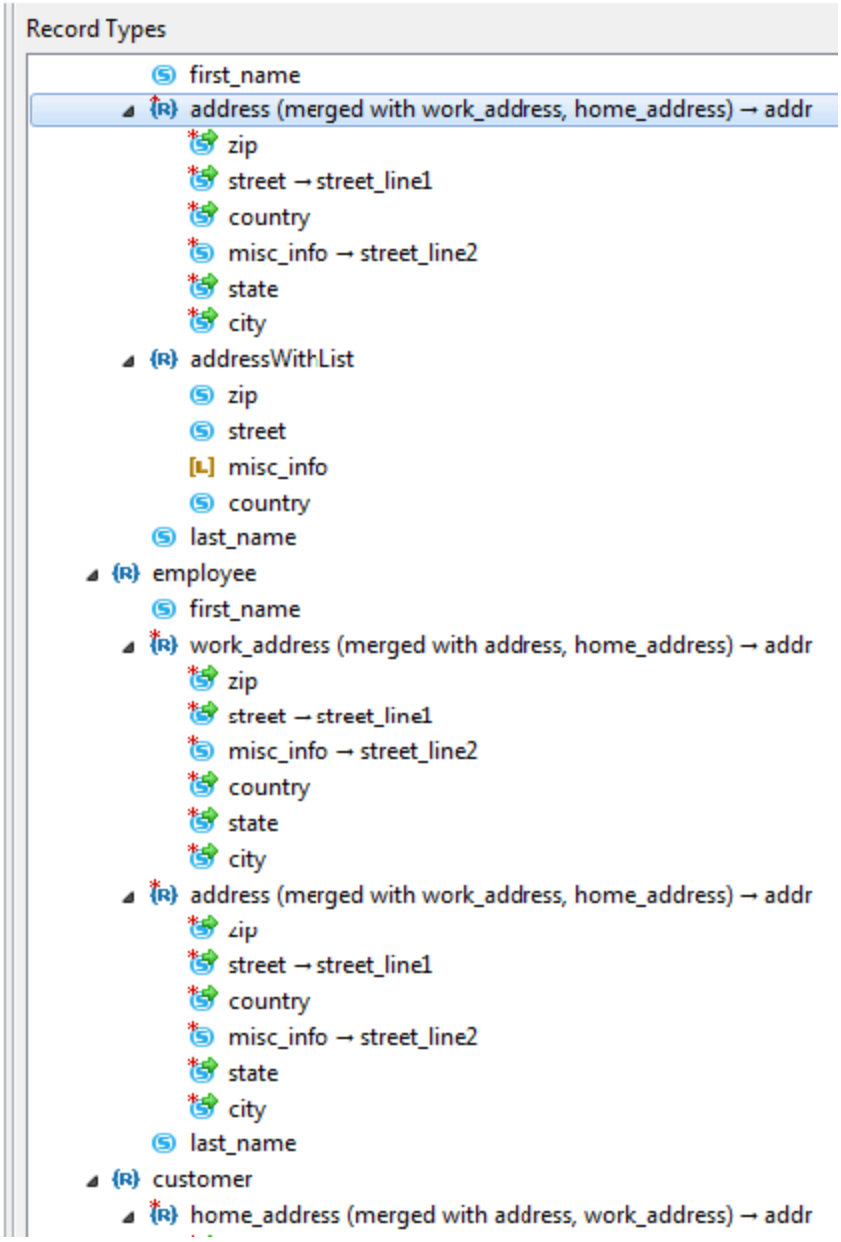


If you right-click **work_address**, and rename it to **address**, then **address** child nodes that were not in **work_address** will be added to **work_address**. Additionally, **work_address** child nodes that were not in **address** will be added to **address**.

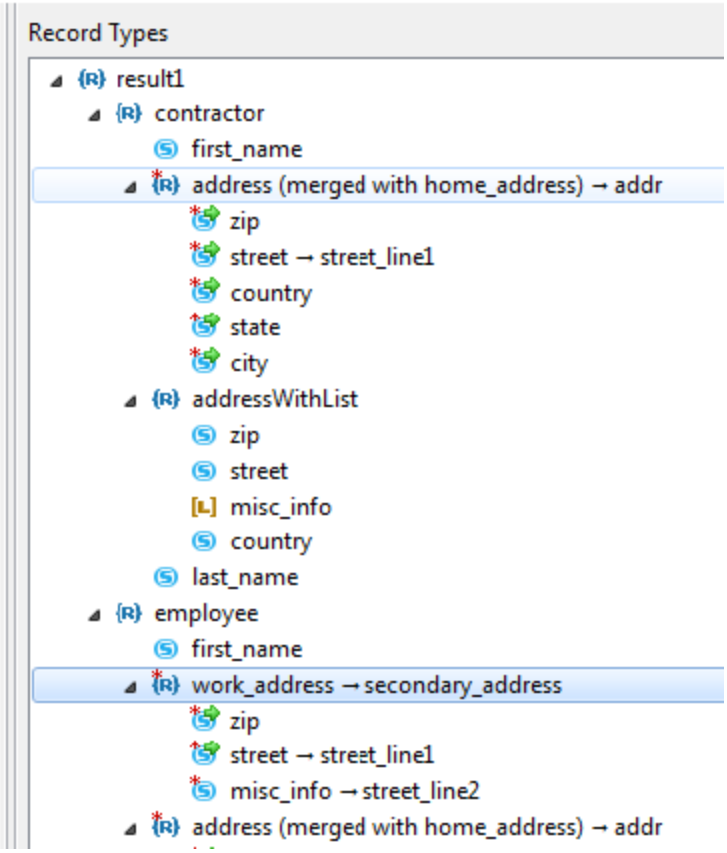
If you rename **home_address** to **address**, then **home_address**, **work_address**, and **address** are all merged together. The node text indicates which types are being merged. Any renaming and identity settings applied after the merge will be applied to all three types.



When you rename a merged type, you will be asked whether the renaming should apply to all the merged types, or only to the selected type. For example, renaming any of the merged types to **addr** and applying it to all the merged types would result in the following:



If you then renamed **work_address** to **secondary_address** and applied that change to only the selected node, it would be "unmerged" from the others—reverting its child list to the original—and then renamed. The wizard will treat it as a different type than **addr**.



Notes

- Any identity settings on a node being renamed to another type would be removed by the renaming. You can set the identity after the renaming.
- wo record types have a child field in common but those child fields have different types, they cannot be renamed. For example, **address WithList** and **work_address** cannot be merged because they both have a child called **misc_info** but they are of different types (one is a list, the other is a string).
- If two record types have a child field in common, they cannot be merged if those fields have been renamed in different ways, or if one has been renamed but the other has not.

Two record types cannot be merged if one type is an ancestor of the other anywhere in the traffic file.

Understanding Data Set Import Options

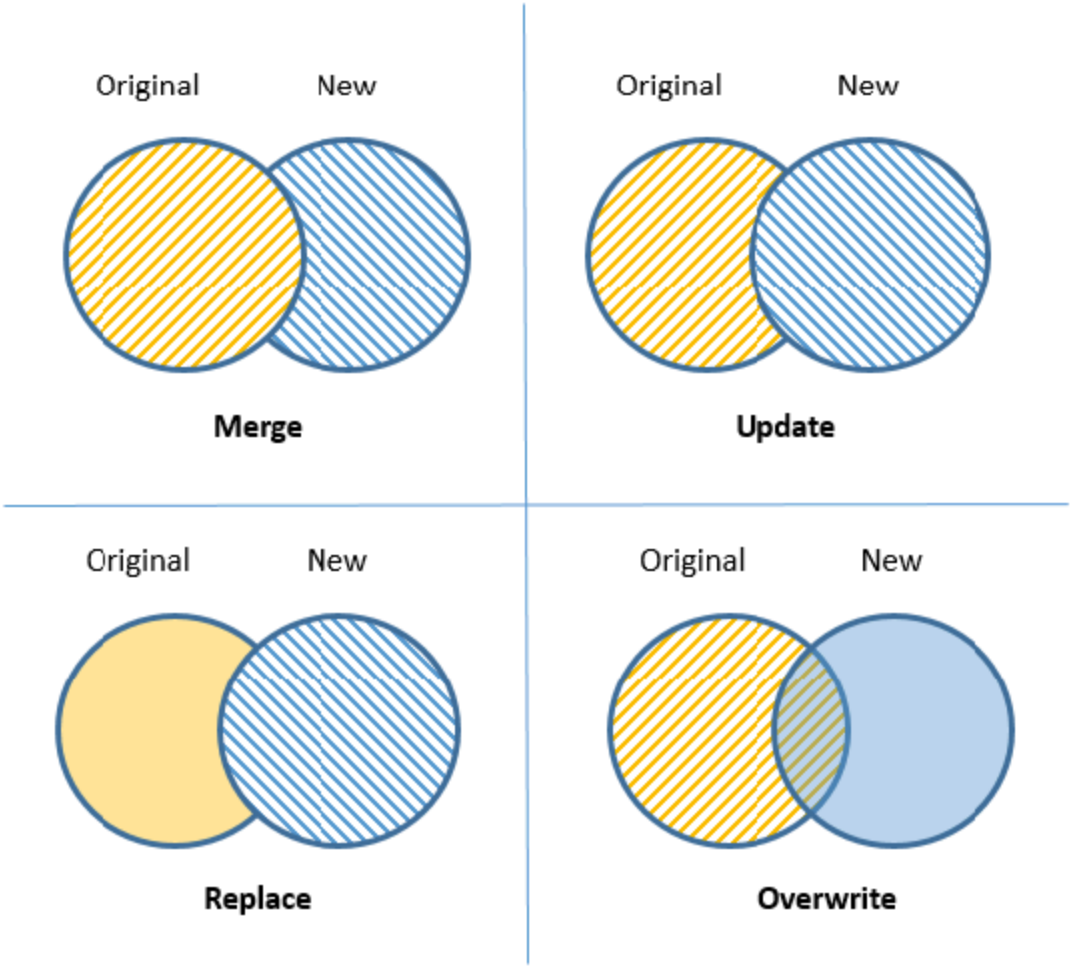
You can choose among the following options to control how new data from the traffic file will extend and/or update existing repository data sets:

SOAtest Options

- **Replace:** Erase existing data then add the new data.
- **Append:** Adds new records without first erasing the existing data.

Virtualize Options

- Replace: Erase existing data then add the new data.
- Merge: Import new data without modifying existing data.
- Update: Update matching records with new data and create new records as needed.
- Overwrite: Update matching records (with matching keys) with new data, do not create any additional records.



For a concrete example of how each strategy operates, assume that you have the following existing data set (where CustomerID is the key column):

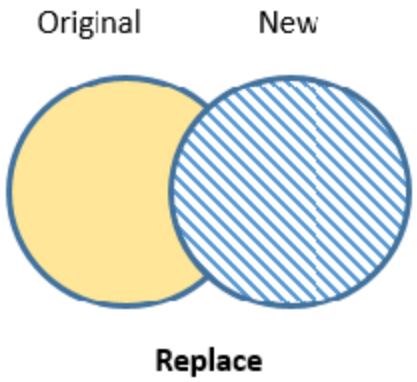
CustomerID	FirstName	LastName
1	Darth	Vader
2	Luke	Skywalker
3	Hans	Solo

Also assume that you have new traffic that contains the following data:

CustomerID	FirstName	LastName
1	Darth	Maul
2	Luke	Skywalker
4	Obi-Wan	Kenobi

Replace

Erases existing data then add the new data.

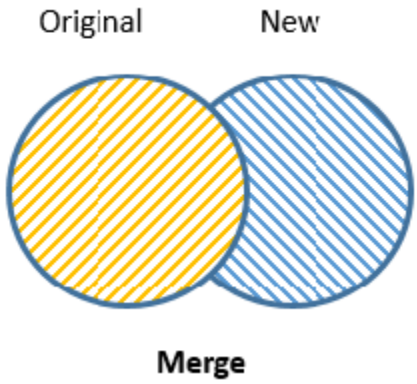


Given the example above, *replace* would result in the following:

CustomerID	FirstName	LastName
1	Darth	Maul
2	Luke	Skywalker
4	Obi-Wan	Kenobi

Merge

Imports new data without modifying existing data.

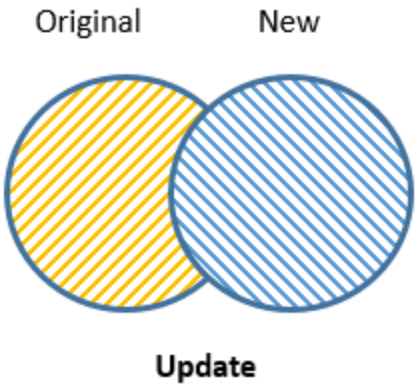


Given the example above, *merge* would result in the following:

CustomerID	FirstName	LastName
1	Darth	Vader
2	Luke	Skywalker
3	Hans	Solo
4	Obi-Wan	Kenobi

Update

Updates matching records with new data and creates new records as needed.



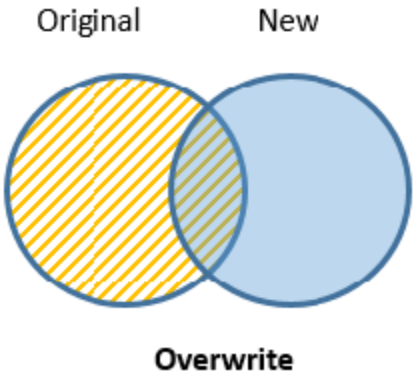
Given the example above, *update* would result in the following:

CustomerID	FirstName	LastName
1	Darth	Maul
2	Luke	Skywalker
3	Hans	Solo
4	Obi-Wan	Kenobi

Overwrite

Updates matching records (with matching keys) with new data, does not create any additional records.

If you import traffic into a new data set, new records will be created even though there are no matching keys.



Given the example above, *overwrite* would result in the following:

CustomerID	FirstName	LastName
1	Darth	Maul
2	Luke	Skywalker

3	Hans	Solo

Understanding Record Import Options

You can control whether matching data (data that matches existing record types, as determined by the identity) reuses existing record types or updates an existing record. Available options are:

- **Reuse:** Reuse/share the existing records that match.
- **Update:** Update the existing records' corresponding fields with data from the traffic and add new records for new record types.

For example, assume an identity for SocialSec and existing records with FirstName, LastName, SocialSec, and Email.

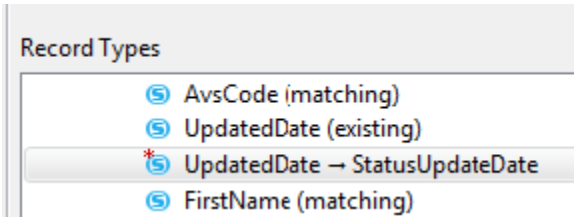
With the Update option...

- If the traffic has data with that same FirstName, LastName, SocialSec, but a different Email, Email will be updated.
- If the traffic has a field that references a CustomerPrefs record type that doesn't have an identity, then a new CustomerPrefs record will be created (in addition to the behavior described above).

With the Reuse option...

- If the traffic has data with that same FirstName, LastName, SocialSec, but a different Email, no fields are updated.
- If the traffic has a field that references a CustomerPrefs record type that doesn't have an identity, no fields are updated and no new records are created.

If you prefer to create a duplicate record type with a different name, right-click the name of the matching record type, choose **Rename** option, then enter the new record type name.



The tree will then show both the original record type and the new (renamed) record type.