Anonymize Test Data with Anonymize-DB

Anonymization Challenges

In a properly managed DevOps environment, only anonymized data should be available in development and test databases.

The Dev/Test database environment should be open by nature, with few constraints, to avoid impairing the work of designers, Developers, and Testers.

The data on those environments should be as close as possible to the Prod data, but should be protected. The best way to handle this “dilemma” is to anonymize sensitive data for two primary purposes:

- Complying with legal and regulatory concerns about sensitive personal data protection
- Protecting business data from possible breaches

Which data should be anonymized?

Sensitive personal data should not be attributed to an identifiable person

The competition should not retrieve sensitive business data

What is “sensitive” is decided by legal, cultural, and business concerns, depending on the specific organization and country.

Anonymization technical constraints

When data is anonymized, it is imperative not to break existing application constraints and validations.

Preserving the inner coherence of composite data (e.g., address with street, city, zip…).

Preserving the structure rules of the data (e.g., Credit Card, IBAN… structure).

Preserving dependency rules (e.g., a Social Security number might contain birth date, which is also stored in a different column of the record)

Anonymize-DB: Anonymize your data in Dev/Test environments

Anonymize-DB provides a high level of freedom to Test and Development teams, allowing them to work with high-quality data without endangering compliance with legal regulations or internal business rules.

It's an automatic, flexible, and innovative process that provides the anonymization manager with a user-friendly and performant tool to anonymize sensitive business or personal data in your Dev/Test databases.

Your security and legal compliance are enforced.

Anonymize-DB supports multiple concurrent RDBMS (DB2, SQL Server, MySQL, Oracle, PostGreSQL…)

Anonymize DB main benefits:

High-quality anonymized data

- Usable (no hieroglyphs, pronounceable…)
- Coherent (repeated data is anonymized in the same way)
- Making sense (conditional separate anonymization, for example, female and male surnames)
- Culturally coherent (use addresses and names matching the organization's geographical location)
- Consistent (same anonymization is applied when data is reloaded, so the Test/Dev environment remains familiar)
- Non-reversible. It is not possible to deduce the original value from the anonymized one when unauthorized

The innovative anonymization process addresses even the most complex databases

- The anonymization consistency is ensured by the use of conversion dictionaries stored in a secure location
- 8 different anonymization methods are proposed, allowing the use of the most appropriate one for each type of data
- Anonymize-DB identifies groups of columns where the same data appears in multiple locations across the database, allowing you to recognize them as belonging to the same “domain”, and to anonymize them consistently
- Anonymize-DB creates unified, consistent dictionaries even when data is stored in multiple and heterogeneous databases

Eases the work and performance of your team

- Anonymize-DB produces an SQL script that can be run by the infrastructure operator, which anonymizes the data. This script can be used for multiple data sets and environments, resulting in a repeatable and automatable process
- With the quality anonymized data provided, testers feel as if they were in production, and data remains consistent even when reloaded
- Test-DB, if combined with Anonymize-DB, will generate only a small subset of the Prod database for Test/Dev, so that the anonymization process will have a minimal impact on the data delivery time

The whole process is documented

Anonymize-DB provides both textual and graphical documentation, which is essential for auditing reports and for follow-up and system evolution.

Anonymize-DB methodology

Identifying Column Groups: Anonymize-DB identifies groups of columns where data appears in multiple locations across the database, allowing you to identify them as belonging to the same “domain” and to anonymize them consistently.

Dictionaries: The consistency of anonymization is ensured by the use of conversion dictionaries, stored in a secure location. The anonymization manager configures the conversion method in these dictionaries.
Methods of Anonymization: Eight different anonymization methods are proposed, allowing for the use of the most appropriate one for each type of data and the combination of techniques to enhance security further. Data in multiple and heterogeneous databases: Anonymize-DB creates unified, consistent dictionaries even when data are stored in various and heterogeneous databases. The dictionaries are kept common and up to date via XML transfer of the updated dictionaries from the database to the database.
Activation: Anonymize-DB generates an SQL script that can be transmitted and executed by the infrastructure operator, thereby anonymizing the data. This script can be used with multiple datasets and environments, yielding a repeatable and automatable process. When setting up the anonymization methods, it is possible to build only the anonymization dictionaries. This allows for viewing and comparing the original and anonymized values without actually modifying the target data.
Performance: When correctly configured, the amount of data to be anonymized is generally small. Consequently, the time required to anonymize data does not significantly impact the delivery of the test database.

The User Interface

Setting the Model

The anonymization setting occurs within a model. This model should contain the tables that have columns requiring anonymization.
To create a new model, from the main menu of Xcase, select "File" and then "New".
To reverse-engineer the tables into the model, select Database from the main Xcase menu, then Reverse Engineer.

Main Dialog

To produce the “Anonymize Column Groups” Dialog, from the main Xcase menu select Database, Anonymize Column Groups.
The dialog contains five tabs, allowing the configuration of the various parameters for anonymization.

- Settings
  In this tab, you set general settings, such as the ODBC connection to the various databases and the location of the Anonymization Dictionaries.
- Functions
  In this tab, you set the custom SQL functions to be used to anonymize data.

- Grouped Columns
  In this tab, you set the groups of columns sharing the same domain that need to be anonymized coherently.
- Calculated Columns
  In this tab, you set the SQL code allowing you to update calculated columns from their components after they have been anonymized.
- Execution
  In this tab, you generate and optionally execute the anonymization script.

Anonymization Diagram

Anonymize-DB enables the automatic production of diagrams displaying the columns that have been set for Anonymization.

- Create a new empty diagram
  From the main Menu, select “Diagrams”, “Diagrams Management”, and click the “New” Button
- In the main toolbar on the left drop-down, set the display to “Anonymize”
- Run the Job named “Anonymization Diagram”
  From the main Menu, select “Code”, “Custom Scripts and Reports”, and double click “Anonymization Diagram” found under the “Anonymization” Category. Rerunning this Job will refresh the diagram if new anonymized columns have been added to the model.
- Optionally modify the Color and Font of the Anonymized columns
  From the main Menu, select “Options”, “Colors and Fonts”. On the left list of the dialog, select “Entity” and on the right list, select “Anonymized Field Font”. Click the “Edit” Button and select the desired font, size, and color.

Anonymized Columns being part of a Unique Constraint

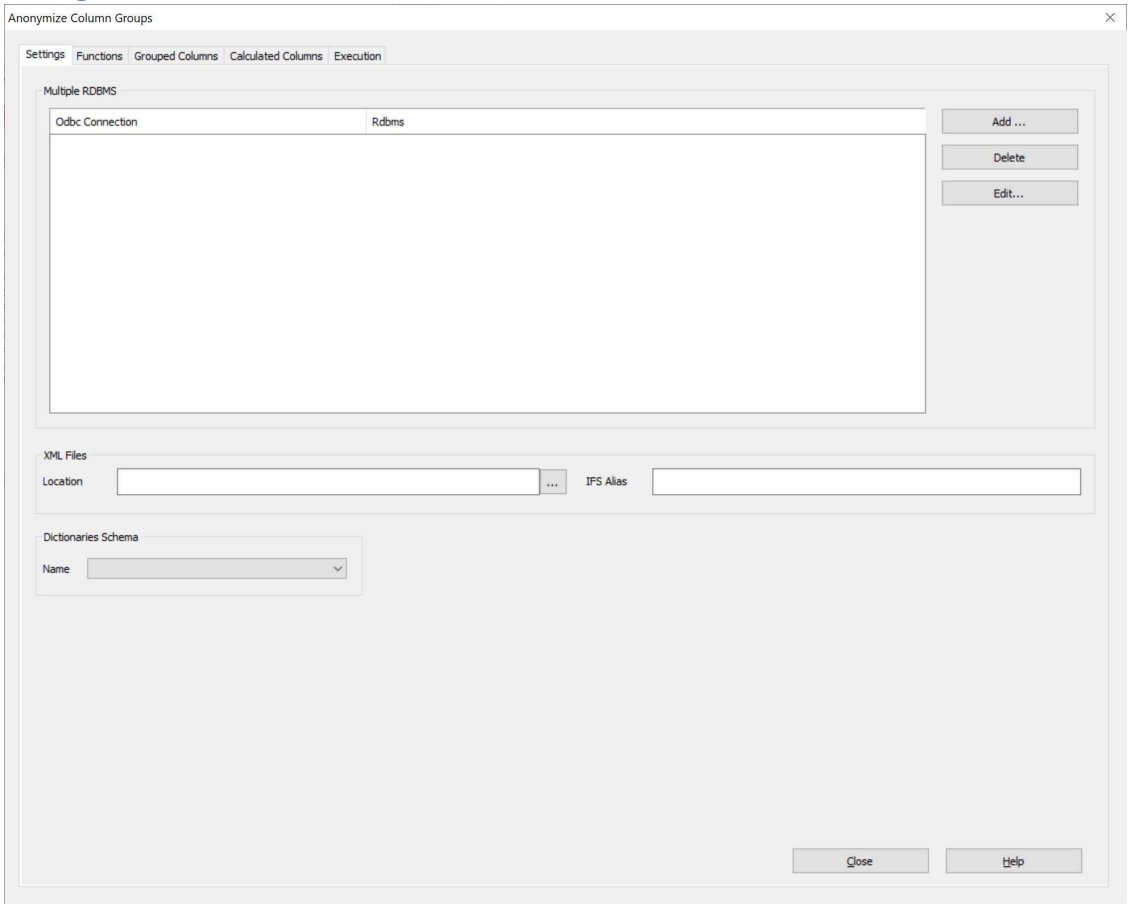
Special attention is required when a column to be anonymized is part of a Unique constraint, as the anonymized values might violate the unique constraint.

To produce a list of those columns and their Unique constraints, run the Job named “Anonymized Columns being part of a Unique Constraint”

From the main Menu, select “Code”, “Custom Scripts and Reports”, and double click “Anonymized Columns being part of a Unique Constraint” found under the “Anonymization” Category.

The different tabs of the “Anonymize Column Groups” Dialog are described in the following pages.

Settings Tab



This tab allows you to configure general anonymization settings.

“Multiple RDBMS” Group

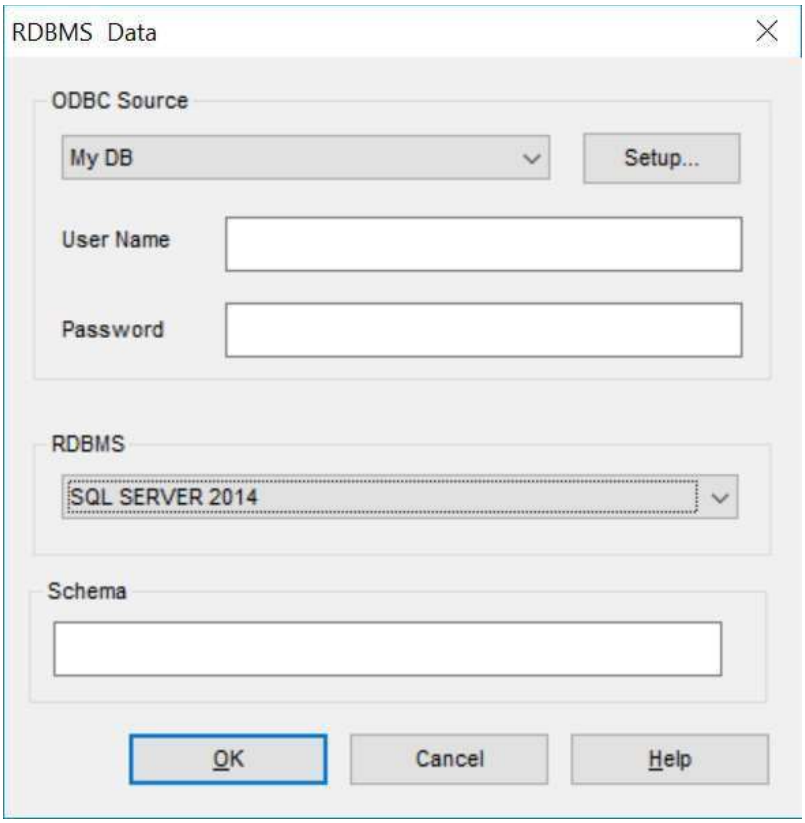
This group is relevant only if you need to anonymize data coherently stored in multiple databases. For example, the customer’s name might be stored in databases such as DB2, SQL Server, MySQL, PostgreSQL, and Oracle. If the scope of the anonymization is limited to a single database, you can disregard it.

Each database requires its own dedicated model. In each model, you define Groups of Columns representing the same “domain”, for example, the Customer Name. The Groups of columns are identified by their name. If the same Group name appears in multiple models, the anonymization will be handled coherently among the different databases for that Group.

For each database and each group of columns, a dictionary will be built in the database. A dictionary is a conversion table containing the original values and the anonymized values to be used. When updating a dictionary for a given group in a specific database, the most up-to-date dictionary across all databases must be used as the basis for the update to preserve coherence.

Anonymize-DB automatically detects which is the most up-to-date dictionary among the different databases and transfers it to the target database as an XML file.
If new data values are found on the target database, the local dictionary table will be automatically enriched to handle them. Existing values will be anonymized uniformly across all databases. This dictionary table then becomes the “most updated”.

The parameters of the other databases participating in the same anonymization process can be defined in any one of the models.
To add (or edit) the required information about a database, click the Add (or Edit) button in the Other RDBMS Group. The following dialog will be displayed:



“ODBC” Group

In the drop-down, select a predefined ODBC connection or click the “Setup” Button to define it.

In the Text Entries, enter the User Name and the Password.

“RDBMS” Group

In the drop-down, select the database type

"Schema" Group

In the Text Entry, enter the name of the Schema / Library where the dictionary tables are stored. Note that access to this schema should be restricted to authorized personnel, as the dictionaries enable the retrieval of the original values before anonymization.

"XML Files" Group

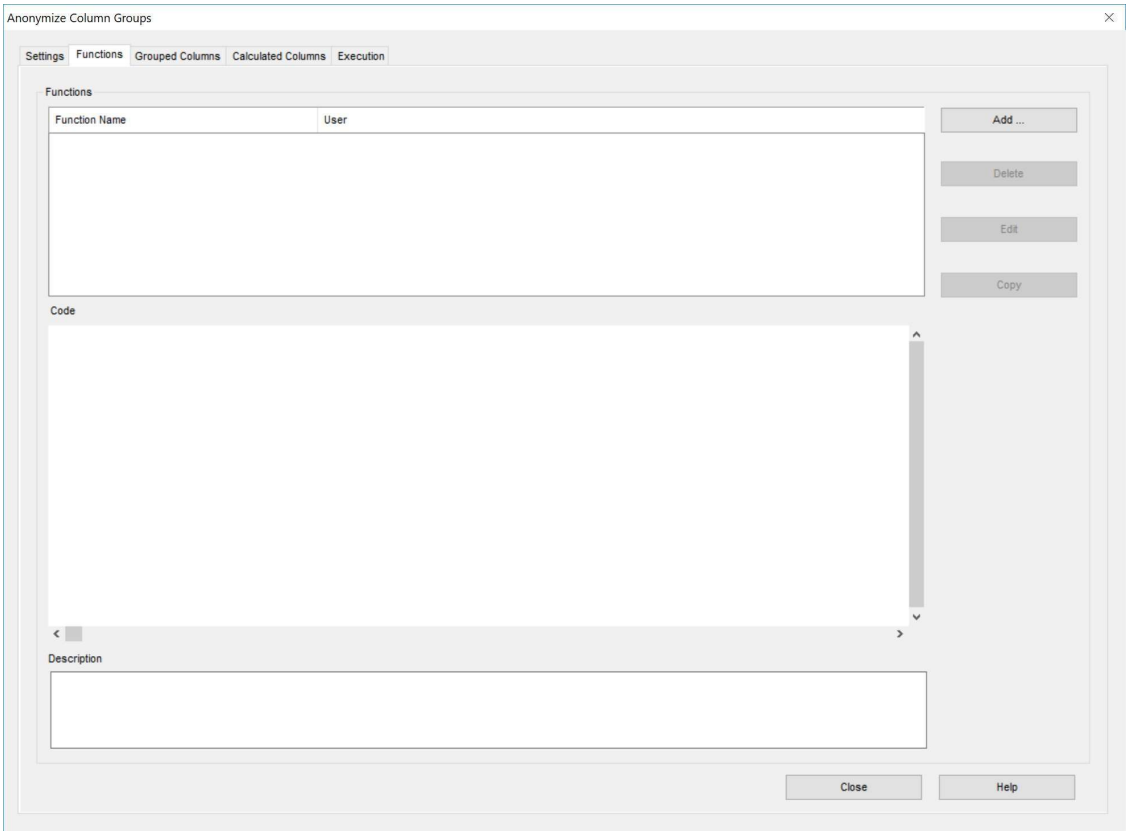
This group is relevant only if you need to anonymize data stored in multiple databases coherently. For the current model, specify the location where the XML version of the Dictionary Tables will be stored in the "Location" text field.
If the current Model target is iDB2 for i, it should be a drive mapping into the IFS, and the "IFS Alias" should also be specified.

"Dictionaries Schema" Group

The Conversion dictionary tables can be preserved in the schema you provide. This ensures that the anonymization for existing column values will be identical to the one previously obtained each time the anonymization process is run. If no schema is provided, a temporary one will be used, which de facto will produce a different anonymization each time the script is run. It is also possible to reset the dictionaries when you wish to get a "fresh" anonymization. This is set individually for each Column Group in the "Grouped Columns" Tab.

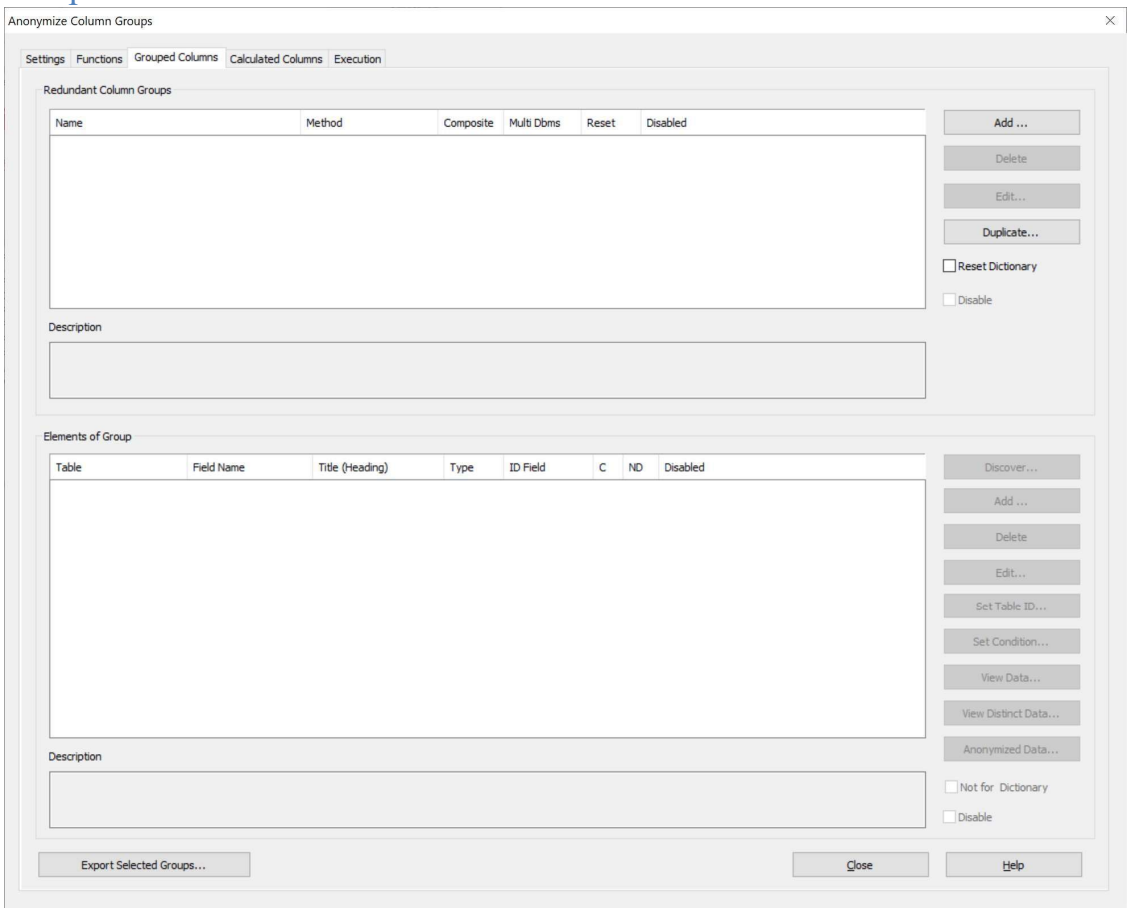
In the "Name" Text Entry, enter the name of the Schema / Library where the dictionary tables are stored for the database of the model. Note that access to this schema should be restricted to authorized personnel, as the dictionaries enable the retrieval of the original values before anonymization.

Functions Tab



This tab displays the list of SQL functions that can be used to anonymize columns. You can create your own or copy and modify the ones provided by the system. The functions defined in this Tab will appear in the generated script provided that they are used.

Grouped Columns Tab

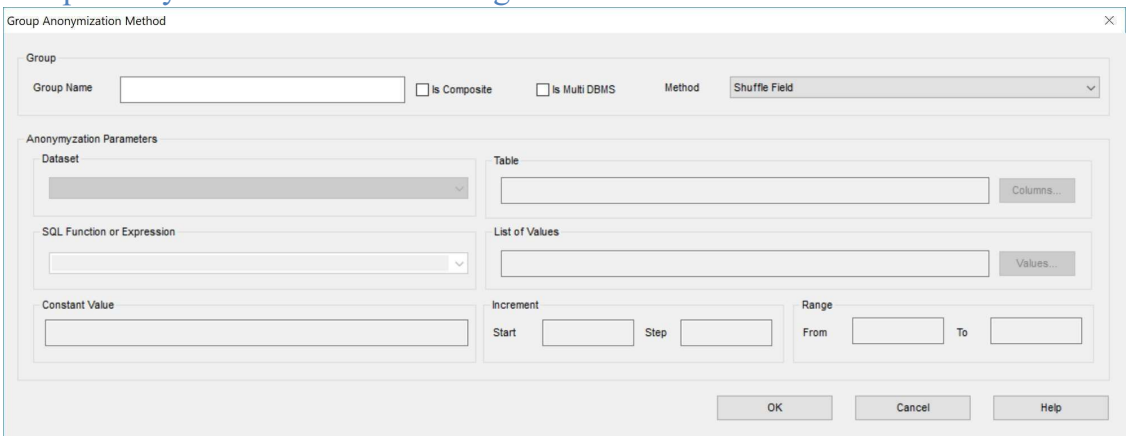


Redundant Column Groups

This list displays the Column Groups that the user has defined. A Column Group is identified by its name and contains one or more Columns belonging to it. A Column Group can be Disabled. This can be set using the Disable Check Box.

To add a new Column Group, click the Add button. This will produce the following dialog:

Group Anonymization Method Dialog



"Group" Group

The "Group name" edit text allows for identifying groups that need to be anonymized coherently among different databases in different models.

The "Is Composite" check box allows you to set that the columns of the group are composite.

A composite definition is helpful in two cases:

1. The columns composing the composite group need to be anonymized "together". For example, an address consisting of separate columns, such as street, province, city, and zip code, might need to be anonymized together, rather than separately, to preserve the coherence of the zip code with the rest of the columns. For this case, you can use the Shuffle Field or the Table methods. The complete address will be randomly replaced with another complete address (same record), either from the same table (Shuffle Field Method) or in a different one (Table Method)

2. To properly anonymize a column, you need the values or one or more additional columns in the same record. In that case, you need to define a custom SQL function to perform the anonymization. The function needs to return the anonymized value of the first element in the composite column group and takes as parameters the values of the different elements of the composite column group, named value1, value2, etc.
   For example, you wish to anonymize your customer's street address and want to select a different random street with the same zip code. In that case, define a composite column group having the street as the first element and the Zip as the second element. Define an SQL function (Myfunc) accepting as a parameter a Zip code and returning a random street (in the same or a different table) having the exact Zip as in the parameter. Define as an anonymization method, SQL Function or Expression, and set it as Myfunc(value2)

The "Is Multi DBMS" check box allows for automatically creating the same group with the same name in each model targeting a different DBMS.

The “Method” drop-down allows you to select the anonymization method for the group.
The anonymization method for a Column Group can be set as:

- Shuffle Column
  The content of the columns belonging to the same Column Group will be randomly shuffled among themselves.
  For each Column Group, a Dictionary or conversion table is automatically built. Each row of the Dictionary table represents a distinct value from the union of all values present in the columns of the Column Group. In this table, a column holds a distinct value, and another column holds its conversion, which has been randomly permuted from the first column.
- Dataset
  Allows you to select one of the provided or user-defined data sets as the source for the anonymization value replacement. The values from the dataset are chosen randomly.
  A dataset is a text file containing a single value per line. It should be located in the DATASET.FILES subfolder of your Xcase folder and should be named as xxxxxxxx.dts, where xxxxxxxx is a name of your choice.
  A free, easy-to-use generator of values that can be used in datasets is available at: https://www.fakenamegenerator.com/order.php
- SQL Function or Expression
  Allows you to select one of the SQL functions as defined in the Functions tab as the source for the anonymization value replacement. You can also enter a valid SQL expression.
- Table
  As in “Shuffle Column,” except that the conversion values are randomly gathered from a user-supplied table and column.
- List of Values
  Allows you to define a list of values as the source for the anonymization value replacement.
- Increment
  Allows you to define an incremental value as the source for replacing the anonymization value.
- Range
  Allows you to define a Range of values as the source for the anonymization value replacement.
- Constant Value
  Allows you to define a Constant value as the source for the anonymization value replacement.

Anonymization Parameters Group

In this group, you can set the parameters of the selected anonymization method.

To modify the General specifications of a Column Group, click the Edit Button.

To delete a Column Group, click the Delete Button.

To duplicate a Column Group, click the Duplicate Button.

"Elements of Group" Group

This list displays the Columns belonging to the selected Column Group in the Redundant Column Group list.

The Columns of the Column Group must belong to one of the Tables of the model and are identified by their table name and column name. Additionally, a column can be identified by an ID in the same table. This ID is usually a number less susceptible to spelling mistakes than a name. For example, let's say that in one table the name of a customer is "John Dow" and in another table, it appears as "John Dowe". The anonymization algorithm will associate those two names with two different values. On the other hand, if you associate an ID (for example, the insurance policy number of the customer) to those two columns and the value of the ID is the same, the anonymization algorithm will associate the SAME random name to both despite the spelling mistake.

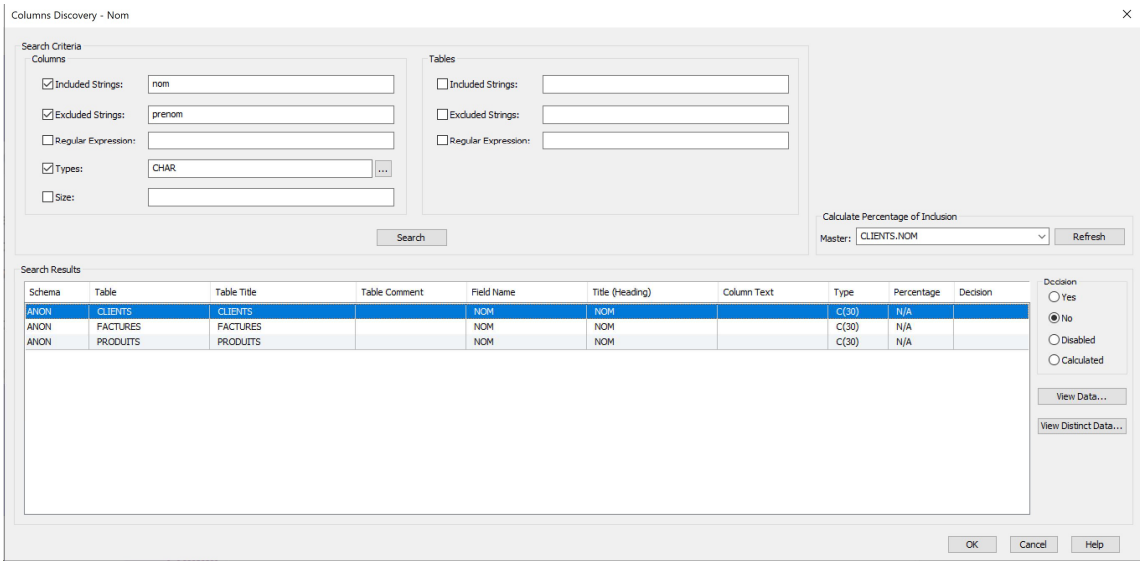
It is also possible to associate an SQL condition with a column. Only the records for which the SQL condition evaluates to True will participate in the anonymization process of the Column Group. This could be useful, for example, when you wish to anonymize the names of persons separately from the names of companies.

Click the "Discover…" Button to produce the "Columns Discovery" Dialog.

This dialog allows the discovery of which columns potentially belong to the same group based on:

- Semantic Data
- Metadata
- Physical Data

Columns Discovery Dialog



"Search Criteria" Group

Note that the discovery for composite columns addresses only the first element of the composite group. The other elements need to be added manually by editing the elements in the Grouped Columns Tab.

Check the "Included Strings" Check Box and specify in the Text Entry a comma-separated list of strings to be found in the Columns or Tables metadata (Name, Heading, Text…). Only if at least one of the strings is found will the column be included in the search.
Check the "Excluded Strings" Check Box and specify in the Text Entry a comma-separated list of strings not to be found in the Columns metadata (Name, Heading, Text…). If one or more of the strings is found, the column will not be included in the search.
Check the "Types" Check Box and specify in the Text Entry a comma-separated list of data types to be found in the Columns metadata. Only if one of the data types is found will the column be included in the search. Click the "…" Button to select one or more of the available Data types.
Check the "Size" Check Box and specify in the Text Entry the size of the Columns metadata. You can specify just a number or >#, >=#, <#, <=#, when # stands for the number. For a range, specify # .. #
Only columns having a matching size will be included in the search.
Check the "Regular Expression" Check Box and specify in the Text Entry a regular expression. Only if the regular expression evaluates to True, the column will be included.

Note that the search criteria are additive. When the Check Box is unchecked, it will be ignored.

Click the "Search" Button to perform the search and display the results in the "Search Results" List.

"Calculate Percentage of Inclusion" Group

Among the discovered redundant columns, one is probably the Master, and the others are redundant occurrences of the master column. If the values of a column are included in the master, this provides a good indication that it is a redundant column belonging to the same group as the master.
In the "Master" drop-down, select one of the discovered columns as the master column. Click the "Refresh" Button to display the percentage of inclusion in the "Search Results" list.

"Search Results" Group

The "Search Results" List displays the list of Columns discovered according to the criteria that were set. You can select one or multiple lines in this list and apply a decision to the selected ones.

"Decision" Group

You can click one of the four radio buttons to apply a decision to the selected lines.

- Yes
  The Column will be added to the Group
- No
  The Column will not be added to the Group
- Disabled
  The Column will be added to the Group as Disabled (when you are not yet sure if the column indeed belongs to the group and need to consult with others).
- Calculated
  The Column will be flagged as calculated and added to the list appearing in the "Calculated Columns" tab.

"View Data" Button

Click this button to view the physical data of the selected column.

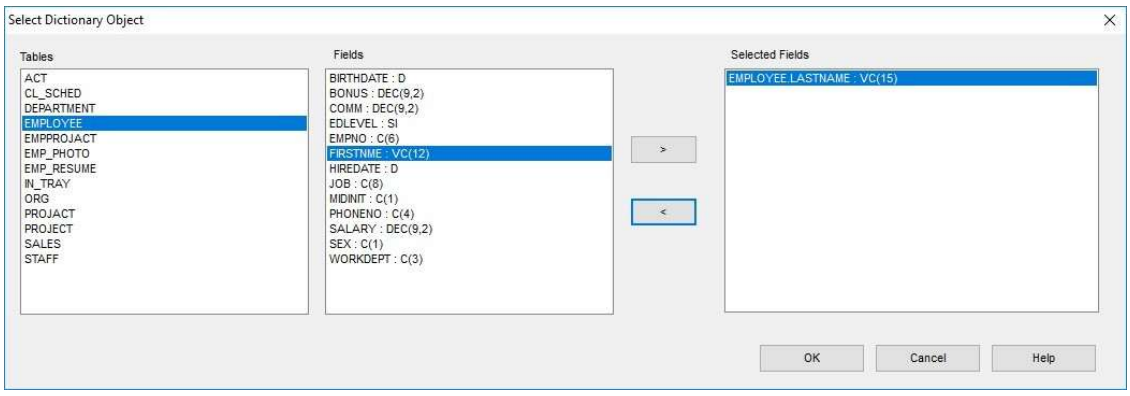
"View Distinct Data" Button

Click this button to view the distinct physical data of the selected column.

"Add" Button in the "Elements of Group" Group

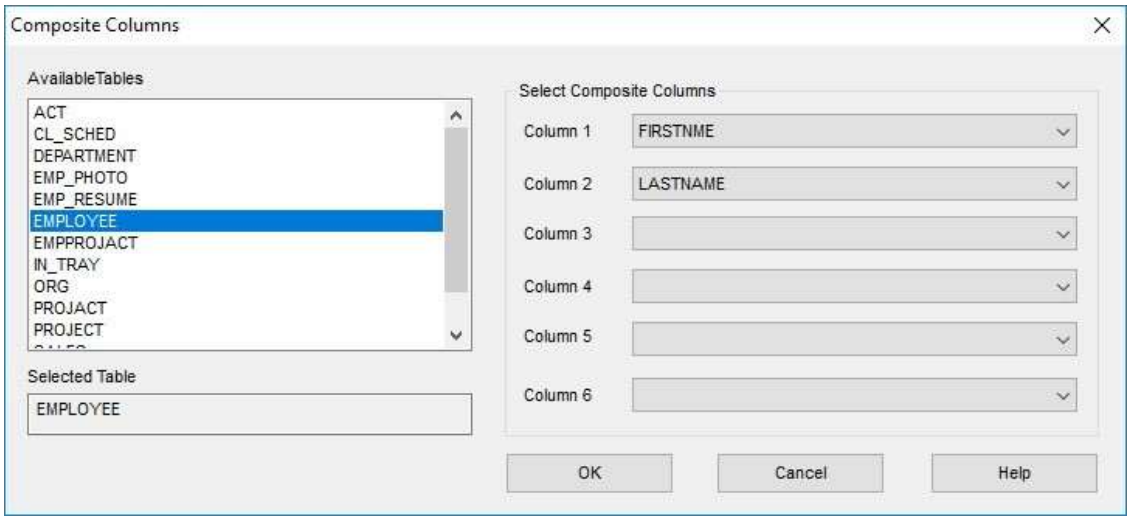
Click the "Add…" Button to select which columns from the model belong to the selected Column Group, in addition to the ones chosen using the "Columns Discovery" Dialog.

When the Column Group is not Composite, the "Select Dictionary Object" Dialog will be displayed. This dialog allows you to select multiple columns from different tables in the model.



If the Column Group was flagged as Composite, the "Composite Columns" Dialog will be displayed. In this dialog, you can set the columns composing the composite group.

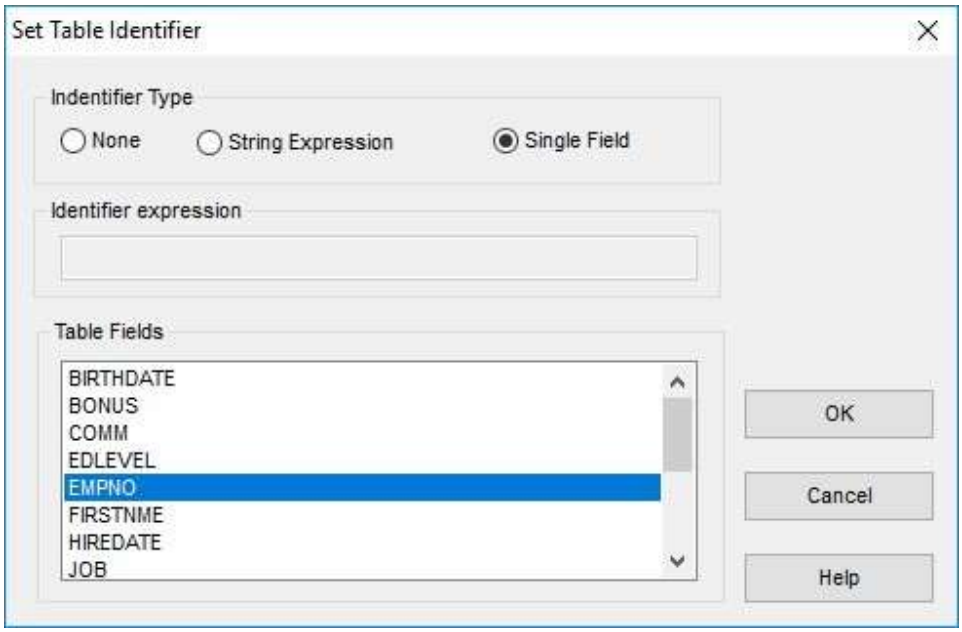
First, select the Table by double-clicking it, and then select up to six columns of the table that define the composite group.



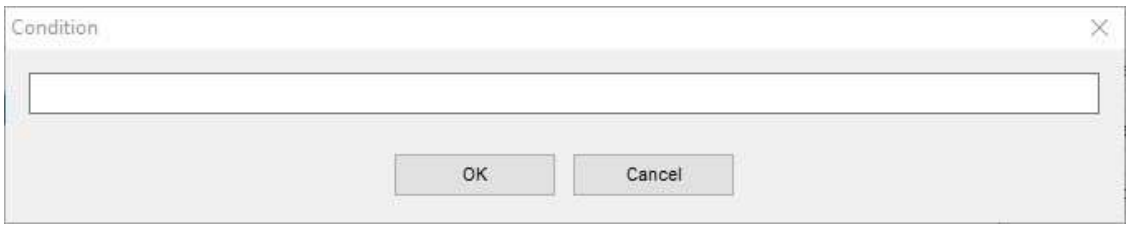
Click the “Delete” Button to delete the selected column from the group.

Click the “Edit...” Button to modify the selected column from the group.

Click the “Set Table ID…” Button to select an identifier from the columns of the table of the selected column. The following dialog is displayed:



Click the “Set Condition…” Button to set an SQL expression controlling which records of the table of the selected column should be considered by the anonymization algorithm.



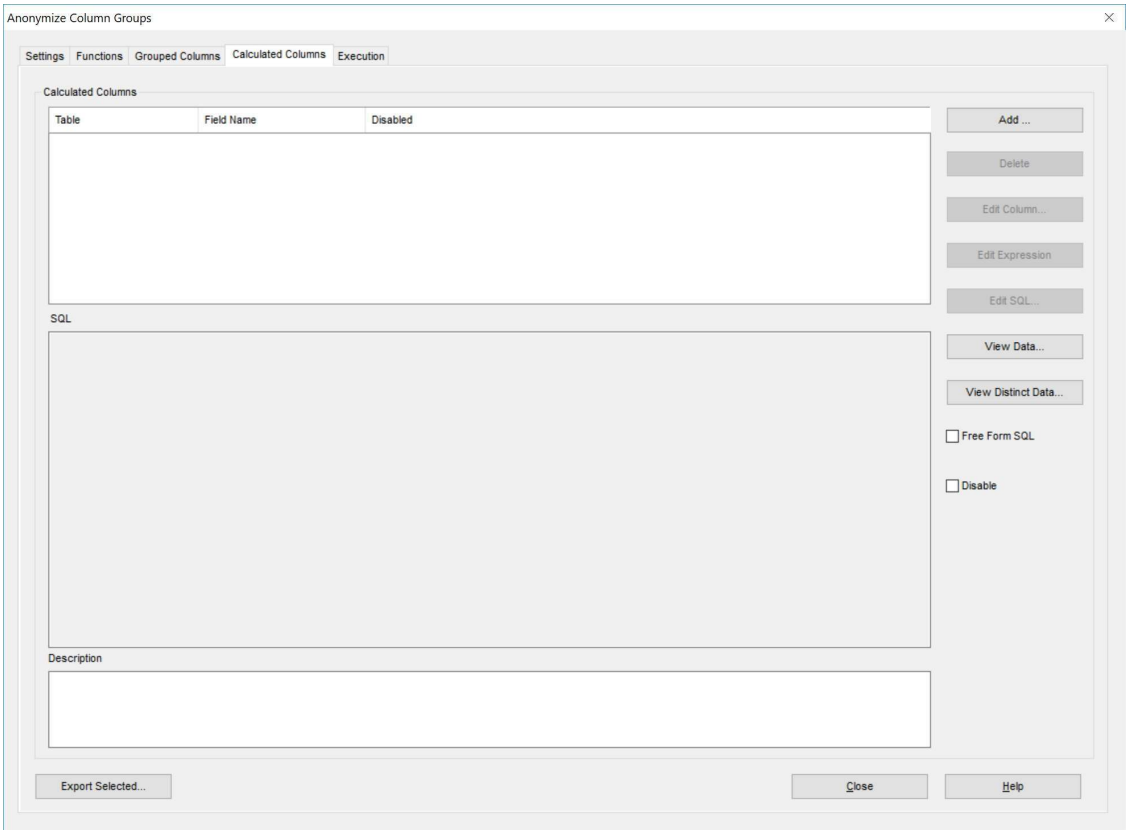
Click the “View Data…” Button to view the data of the selected column from the group. Click the “View Distinct Data…” Button to view the distinct data of the selected column from the group.
Click the “Anonymized Data…” Button to view the original and the simulated anonymized data of the selected column from the group. Note that this requires that the conversion dictionaries have been generated. The generation of the conversion dictionaries is done in the Execution Tab.

When a master column already provides the values of a group element, you can specify that it will not be taken into account when building the group conversion dictionary by checking the “Not for Dictionary” checkbox. In that case, you do not need to specify its condition.

A Column of a Group can be Disabled. This can be set using the “Disable” checkbox.

Click the “Export Selected Groups” Button to export the selected Column Groups definition into CSV.

Calculated Columns Tab



A calculated column is a column derived from other columns—for example, the concatenation of the first and last name of the customer. To coherently anonymize this column, it should be reevaluated after anonymizing its components.

In this Tab, you can specify which columns are calculated from anonymized columns and set the SQL code required to reevaluate them after anonymization has been executed.

Click the “Add” Button to select a calculated column.

Click the “Delete” Button to delete the selected calculated column.

Click the “Edit” Button to select a different calculated column.

Click the “Edit Expression” Button to set the expression that calculates the calculated column.

Click the “Edit SQL” Button to edit the complete SQL statement. Note that the “Free Form SQL” Check Box must be checked.

Click the “View Data…” Button to view the data of a calculated column.

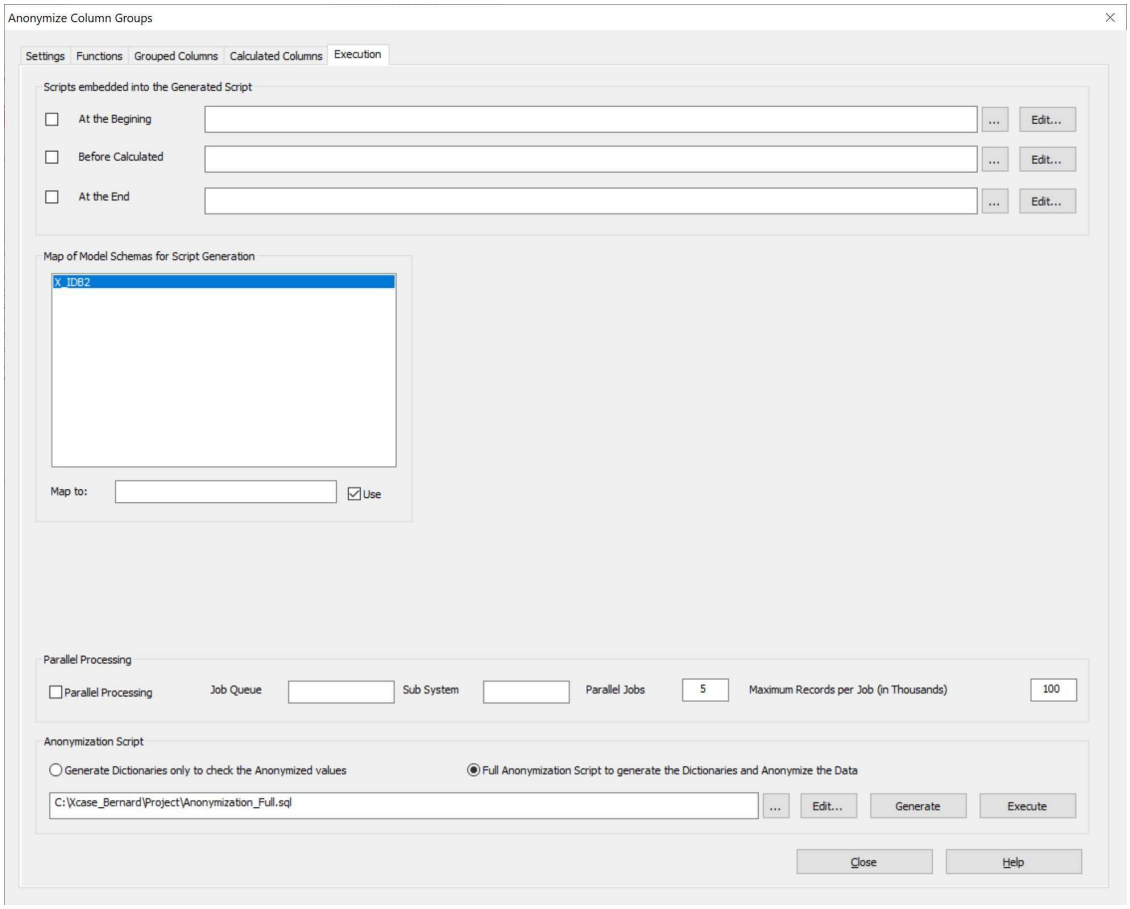
Click the “View Distinct Data…” Button to view the distinct values of the selected calculated column.

Check the “Free Form SQL” Check Box to be able to edit the complete SQL statement when clicking the “Edit SQL…” Button.

Check the “Disable” Check Box to turn off the handling of the calculated column.

Click the “Export Selected” Button to export the selected Calculated Columns definition into a CSV file.

Execution Tab



This dialog allows you to generate and optionally execute the anonymization script.

"Scripts embedded in the Generated Script" Group

The generated script is composed of two parts. The first is the anonymization script, and the second handles the calculated columns. If needed, additional manually defined scripts can be embedded with those two.

- At the Beginning
- Before Calculated
- At the End

Use the Text Entry to enter the qualified names of the manual scripts. Use the "…" Button to select them and the "Edit" Button to edit them.

"Map of Model Schemas for Script Generation" Group

If the generated script needs to reference different libraries/schemas than the ones in the model, you can map the ones in the model to the desired ones, and the generated script will use the mapped ones.

“Parallel Processing” Group (DB2 for i only)

To accelerate the Anonymization process, Anonymize-DB leverages the parallel processing capabilities of DB2 for i to its fullest advantage. Multiple parallel processes can anonymize a large file, each handling a relative part of the file according to its Relative Record Number.

“Anonymization Script” Group

You can generate a script only to build the conversion dictionaries or to perform the actual anonymization on the target data.

When building only dictionaries, they can be used to simulate data anonymization and display anonymized data in the Grouped Columns Tab by clicking the “Anonymized Data” button for a column. This allows you to inspect the resulting anonymized values and compare them with the original values without actually changing them.

Click the “Generate Dictionaries only to check the Anonymized values, or the “Full Anonymization Script to generate the Dictionaries and Anonymize the Data” radio buttons.

Enter the desired qualified script name in the Text Entry or click the “…” Button to select it.

Click the “Edit…” Button to edit the script.

Click the “Generate” Button to generate the anonymization script.

Click the “Execute” Button to run the generated script on the server, plus the additional scripts via an ODBC connection.