IDEA Documentation

Versie 1.0 ● Proposed

Table of Contents

IDEA Documentation

This is the documentation of the IDEA AddOn. It is a combination of:

- Description of the user interface
- Source code of the AddOn

IDEA64

IDEA64

IDEA64

DeduplCompareItem ()
Class for comparing the elements in the deduplicator routine, it has two types of compare properties, one for the name and one for the total of an element.

DLANormalizeDataTable ()
Class for normalizing a datatable imported from an excel sheet

FrmArchimAID ()
Form for ArchiMate modeling that combines searching and a toolbox to prevent creating duplicates See also the screen description

FrmArchimAID ()
Form for ArchiMate modeling that combines searching and a toolbox to prevent creating duplicates See also the screen description. This is the definition part generated by the Visual Studio screen painter

FrmDiagramDuplicator ()
Form for the deduplication of diagrams. This is actually a duplicator since it duplicates all the elements in the diagram. This is the generated code by Visual Studio

FrmDiagramDuplicator ()
Form for the deduplication of diagrams. This is actually a duplicator since it duplicates all the elements in the diagram

FrmElementDeduplicator ()
Deduplication of an element with a number of subscreens in the tabpage to display the differences. This is the screen generated by Visual Studio

FrmIDEAClass ()
Form for IDEA routines specific for UML class entities. For every type of element a specific form is generated. This makes working with the IDEA AddOn easier.

FrmIDEAClass ()
Form for IDEA routines specific for UML class entities. For every type of element a specific form is generated. This makes working with the IDEA AddOn easier.

FrmIdeaDiagram ()
Form for IDEA routines specific for UML class entities. For every type of element a specific form is generated. This makes working with the IDEA AddOn easier.

FrmIdeaDiagram ()
Form for IDEA routines specific for diagram entities. For every type of element a specific form is generated. This makes working with the IDEA AddOn easier.

FrmIDEATable ()
Form for IDEA routines specific for database table entities. For every type of element a specific form is generated. This makes working with the IDEA AddOn easier.

FrmIDEATable ()
Form for IDEA routines specific for database table entities. For every type of element a specific form is generated. This makes working with the IDEA AddOn easier.

FrmImportExcel ()
Import data from excel sheets with helper routines for to make advanced import routines with associations and other entities

FrmImportExcel ()
Import data from excel sheets with helper routines for to make advanced import routines with associations and other entities

FrmPackageDeduplicator ()
Deduplication of an package and all the elements in the package with a number of subscreens in the tabpage to define the merging of the related elements and their features

FrmPackageDeduplicator ()
Deduplication of an package and all the elements in the package with a number of subscreens in the tabpage to define the merging of the related elements and their features

FrmQueryExport ()
Helper screen for an advanced user to do some advanced things with the database with SQL queries. You can retrieve data but also manipulate the data in the repository.

FrmQueryExport ()
Helper screen for an advanced user to do some advanced things with the database with SQL queries. You can retrieve data but also manipulate the data in the repository.

frmReleaseManager ()
Screen for doing release management in a DTAP configured environment

frmReleaseManager ()
Screen for doing release management in a DTAP configured environment

FrmUniqueElement ()
Form for displaying the elements that are duplicate in the repository. Give a warning and eventually add the duplicate element to the diagram.

FrmUniqueElement ()
Form for displaying the elements that are duplicate in the repository. Give a warning and eventually add the duplicate element to the diagram.

HTMLPublicator ()
Class for the HTML generator. Various functions to generate HTML pages and snippets from the elements in the repository

IDEADataTypes ()
Class for a list of datatypes for the conversion from one layer in the other in the data model (LDM - TDM)

IDEADefinitions ()
Class for processing definitions, such as connections, queries, etc. Is done based on a Dataset that is stored via XML in the config file

IDEAGenerator ()

Generating elements from one layer to the other (CDM - LDM - FDM) These are complex transformations tha tcan be modified when relevant. Important is to consider that these routines are based on the IDEA screens. In the scripts in the sample repository there are differences in implementation.

ReleaseContainer ()
Container class for storing release manager steps for transferring XML specifications from one environment to the other.

ReleaseManager ()
Bring the data of a release from one environment to the other in a DTAP implementation

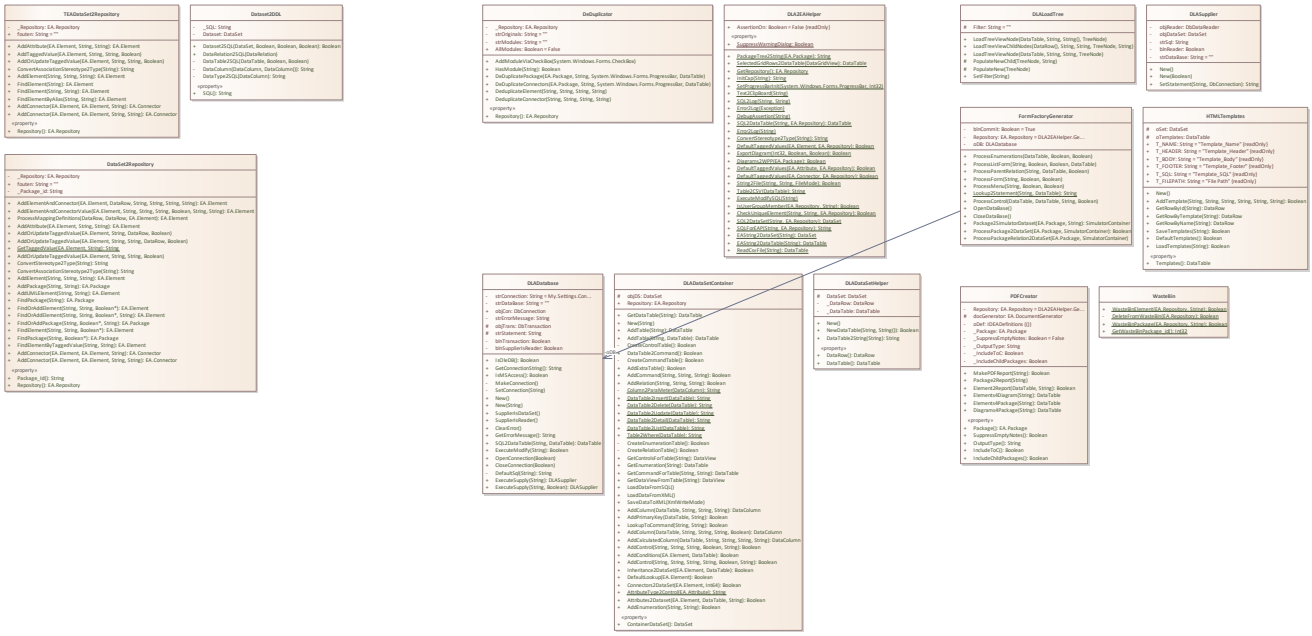
SimulatorContainer ()
Class added for inheritance and encapsulation of the routines

WndWelcome ()
Welcome screen of the IDEA AddOn, code generated by Visual Studio

WndWelcome ()
Welcome screen of the IDEA AddOn

DLAFormfactory

DLAFormfactory



Dataset2DDL ()
Transform the dataset content to DDL commands

DataSet2Repository ()
Helper class for transforming output from the Sparx API in a more usable format within a DotNet (windows) application.

DeDuplicator ()
Class for doing the generic routines of deduplication. This has various routines for deduplicating elements.

DLA2EAHelper ()
Class for all kinds of helper routines, it is the base for all the other routines and classes in the library When possible shared functions are used so this functions like a function library

DLADatabase ()
Communication with a database via oledb

@remark

DLADataSetContainer ()
Dataset container for the IDEA simulator

DLADataSetHelper ()
Helper for working with datatables and datasets in import and export routines

DLALoadTree ()
Class for loading a treeview in the explorer windows of the simulator

DLASupplier ()
Class for encapsulating navigation through a recordset can be used for example to fill list boxes etc.

@remark

FormFactoryGenerator ()
Class for generating source code for the forfactory ASP.Net application. (Specific routines not further documented)

HTMLTemplates ()
Dataset for creating templates for the HTML generator in IDEA Templates are filled with a default model but can be modified

PDFCreator ()
Report generator based on the sql statements used in the report generator screen (package helper) Currently only used in the HTML reporter

TEADataSet2Repository ()
Bring elements from a datatable to the repository with different functionalities like: Adding or Updating elements Adding or updating tagged values Adding or updating connectors Etc

WasteBin ()
Class for handling wastebin routines when a user deletes an item from the repository

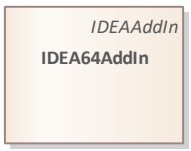
IDEA

IDEA

IDEAAddIn ()
Connector class for the menu options etcetera for the AddOn the functions are called from the EA application Here all the connection points from the EA application are defined

IDEA64

IDEA64



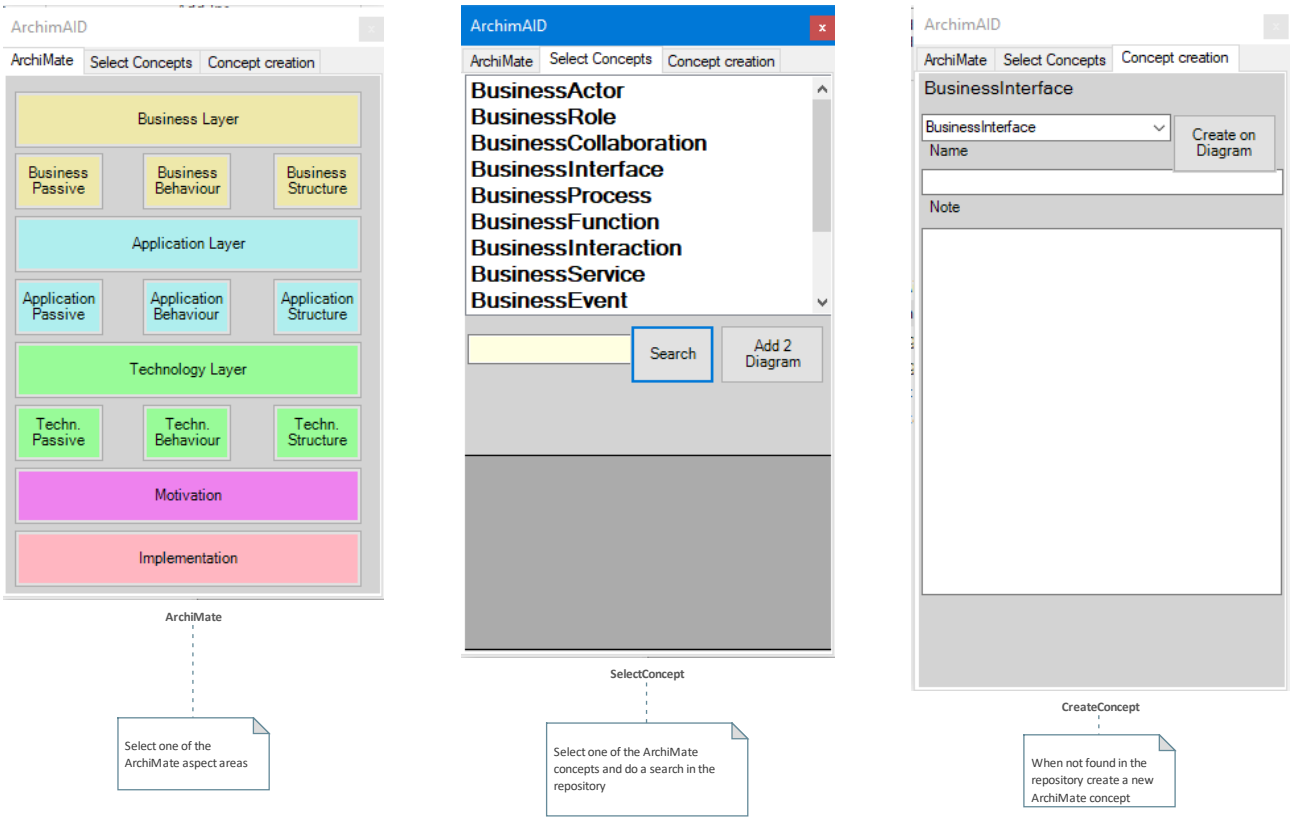
IDEAScreens

Screens documentation with a brief description of the various screens in the IDEA AddOn. Please see also the webvideos on the various screens and usage of functionalities

ArchimAid

ArchiMate is a part of the AddOn with a screen for a combination of search and toolbox for ArchiMate. It reduces the change for duplicate elements in the repository.

ArchimAid



ArchiMAID is a screen in EA that combines the toolbox and a simple keyword search diagram in a small screen. The screen has three tabpages related to each other and having a kind of wizard function

ArchiMate («Image»)
ArchiMate
Select one of the elements in the model (3*3 matrix) of ArchiMate or in one of the two extensions. You can select the elements in a layer or extension or select within the layer in on of the "columns"of ArchiMate (Active, Behaviour, Passive)

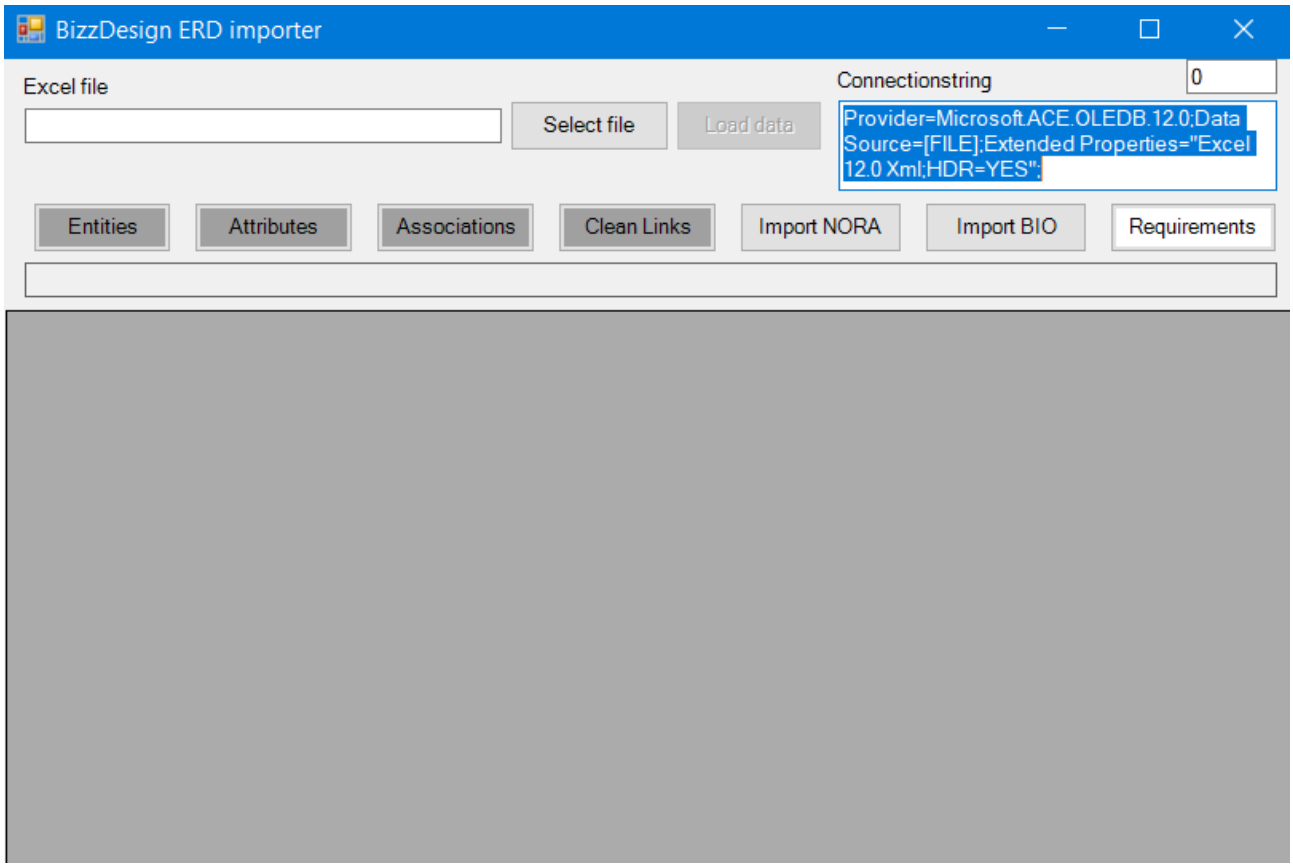
CreateConcept («Image»)
Concept creation
When no items are found you have the possibility to add an element in this last tabpage. When you push the Create on Diagram button the new element is created on the diagram.

SelectConcept («Image»)
Select concepts
In the second tab you can select which concept is relevant for you. When you selected one of the concepts you can search (on keyword) in the full repository to check the existance of the concept. If so a list of elements will be selected in the grid at the bottom of the screen. When selecting one of these elements you can use the Add2Diagram button to add the selected element to the diagram.

Assistant

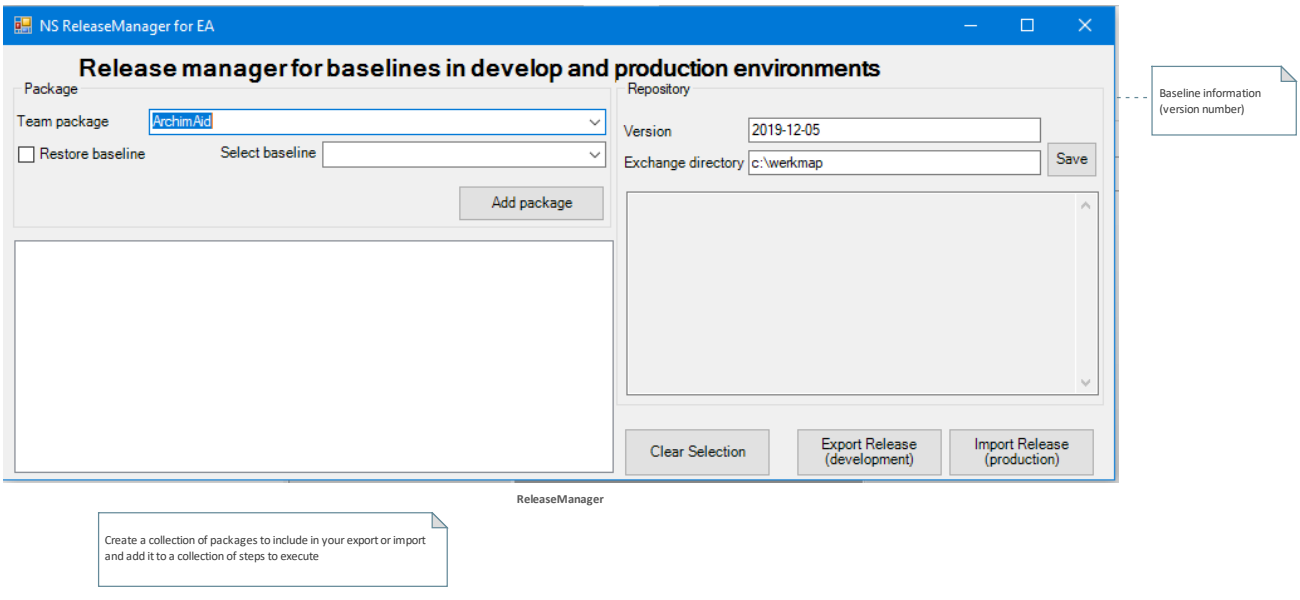
Assistant is a number of screens with extra functionality. It is available via the menu specialize and in the ribbon the IDEA button (under Add-Ins)

Importer



Importer

Release manager

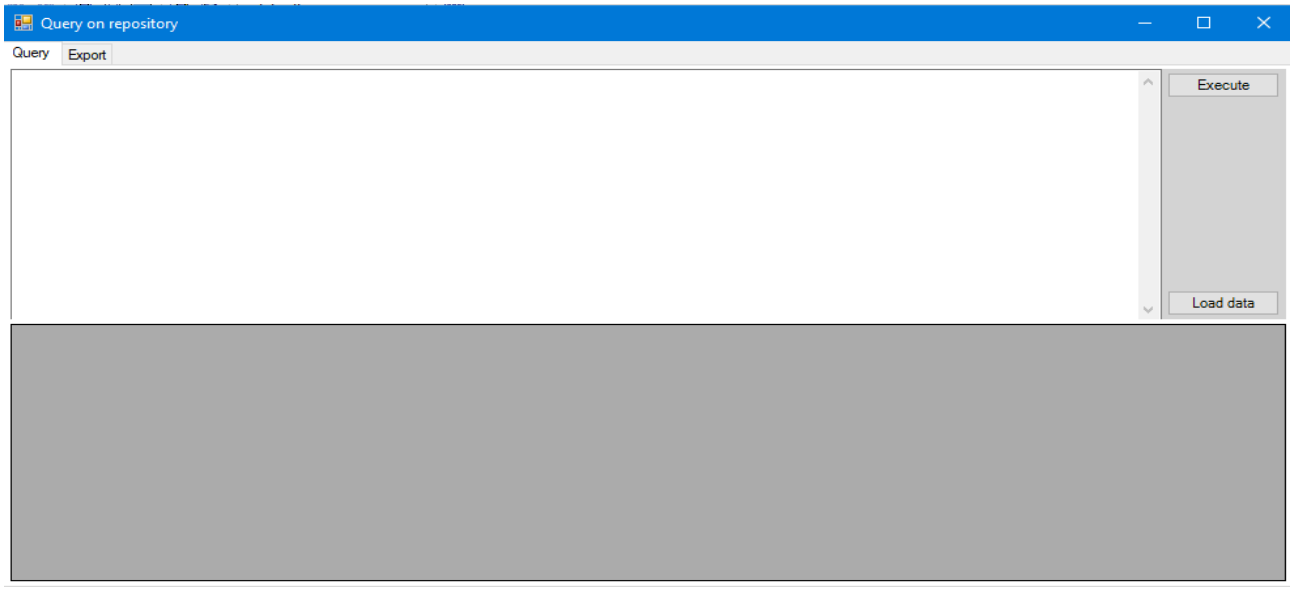


ReleaseManager

The release manager is a screen that makes it possible to easily transfer parts of your repository content in a DTAP street of repositories.

Especially when working with more than one agile tteam in one repository this is a relevant feature. Since in taht situation it occurs that the agile team have different release rhytms. This makes it possible to do a release of one team package and temporary restore the packages of other teams still working on a release.

Simple query



Query on Repository

The query on repository is a helper for experienced users to query on the repository based on SQL statements.

In the query textbox you can enter a SQL query. When you push the Load data button the resultset (of a selectstatement will be displayed in the grid at the bottom of the screen.

When you want to do an update on the content you can create a manipulation command in the textbox and press the Execute button.

Eventually you can export the data to an excel file when relevant

Remark
When security is enabled you need to have administrator rights to execute these commands

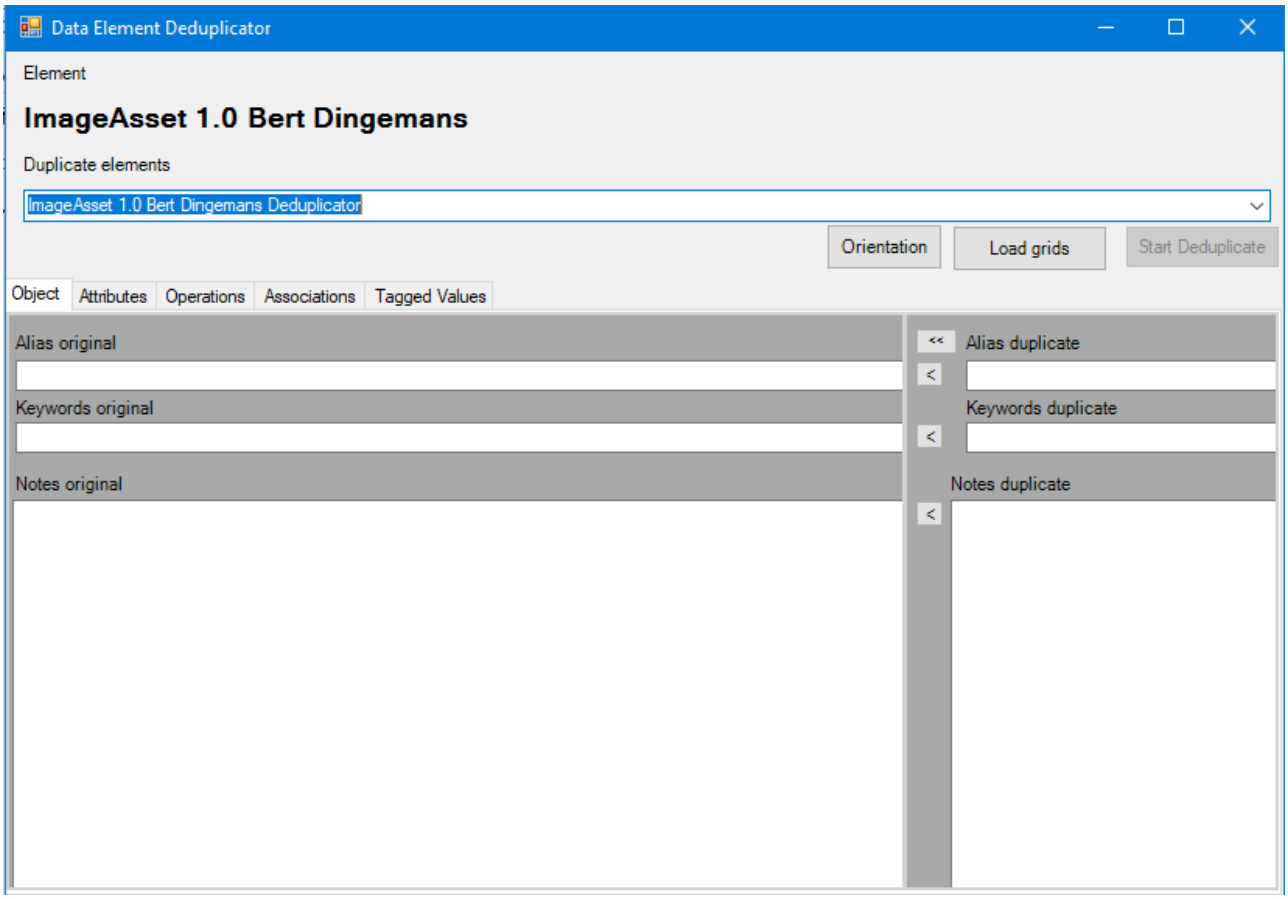
Importer («Image»)
In the importer window you have the possibility to import an excel file from a BizzDesign export. In the zip of the Add-On you can find an example of this excel sheet.

When you select this excelsheet in this window and load the data via the Load data button, the dat will be displayed in the grid at the button of the screen.

Deduplicator

The deduplicator is a rountine in the IDEA addon to merge elements with the same name and stereotype. This can be done on package and element level.

Data element deduplicator



DataElementDeduplicator

Diagram Deduplicator

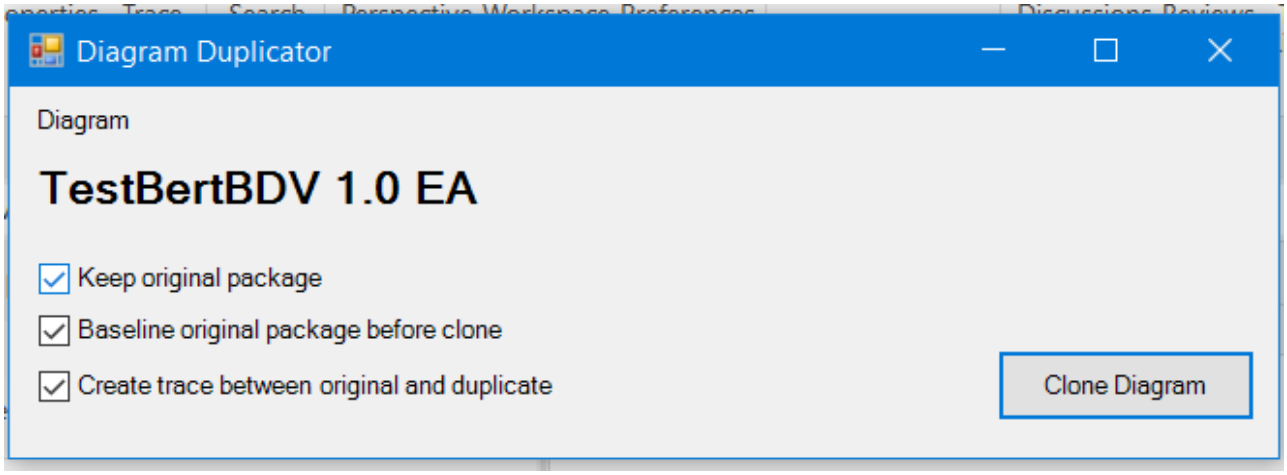
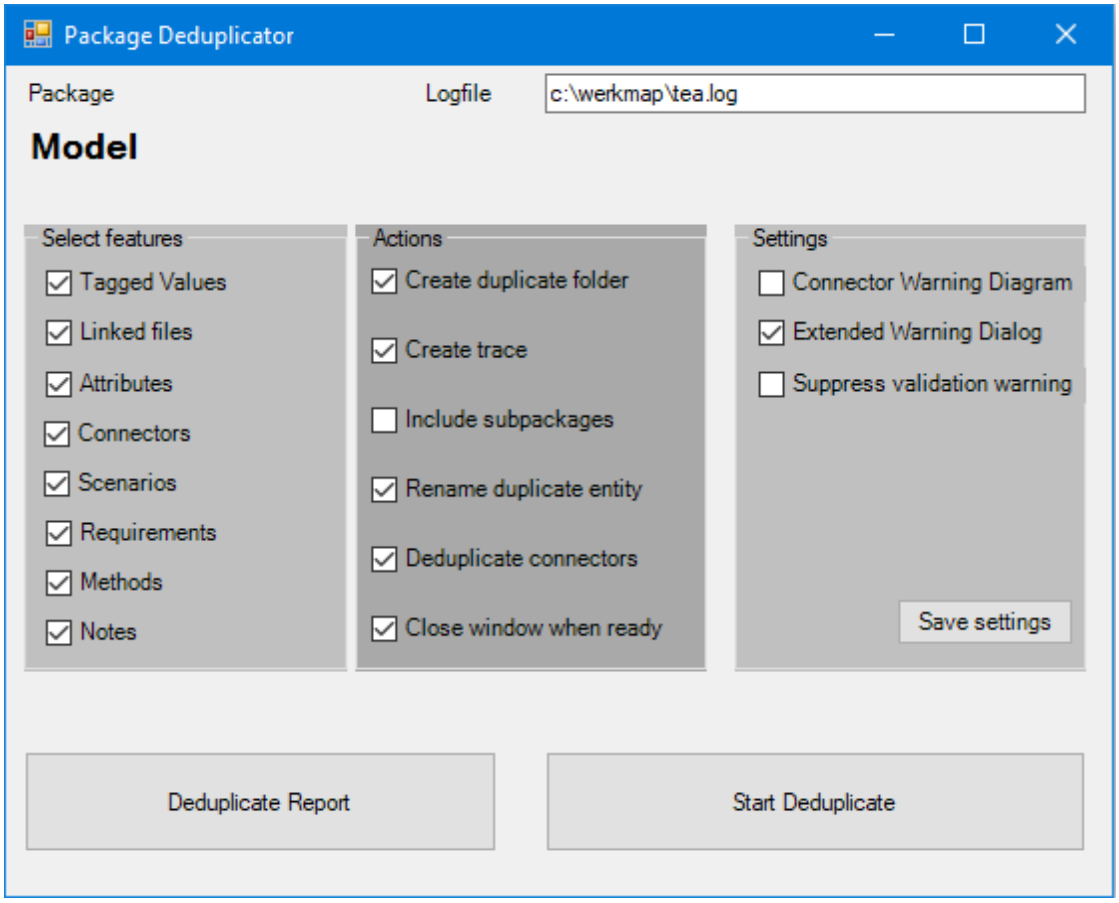


Diagram Deduplicator

The deduplicator on diagram level is actually a duplicator and is used in combination with the element deduplicator.

Package Deduplicator



Package Deduplicator

The package deduplicator is the most powerfull deduplicator. It can deduplicate elements in a package and its subpackages. So when selecting the highest level (the root) you will deduplicate the full repository.

DataElementDeduplicator («Image»)
When you have selected two elements you can load the grids. When the grids are loaded you can compare the features of the original and the duplicate with each other in the tabpages.
In the tabpages you have the possibility to modify the generated action
When ready you can press the start deduplicate button, the features from the duplicate to the original

Diagram Deduplicator («Image»)
After selecting a diagram you can duplicate the elements that are visible on this diagram. The routine will make a package for the originals, duplicate the package and eventually add some extra functionalities like:

- Keep the original elements (as available in the diagram) are kept in the original package after running this functionality (otherwise the elements are copied back to their original location)
- You can create a baseline of the elements in the original package for release reasons
- Creating a trace association between the original and the baseline for easy navigation in the traceability window for example.

Package Deduplicator («Image»)
Select features
Here you can select which features will be merged from the duplicate elements to the original element

Actions
With the actions you can define a number of functionalities when deduplicating:

- Create duplicate folder collects all the duplicate elements in one package with the name duplicates
- A trace association is created with create trace
- When include subpackages is checked the operation will be recursive
- When rename entity is checked the duplicate element is renamed to the name starting with an underscore
- Deduplicate connectors also deduplicates the connectors of an element

- This can be a long running process, with close window checked the window will close when the routine is ready

Settings

- Connector warning diagram gives a popup when closing the window (not yet active)
- Extended warning dialog is an advanced popup when a duplicate is found during diagramming
- Suppres warning shows a warming when you add a duplicate element to the repository.

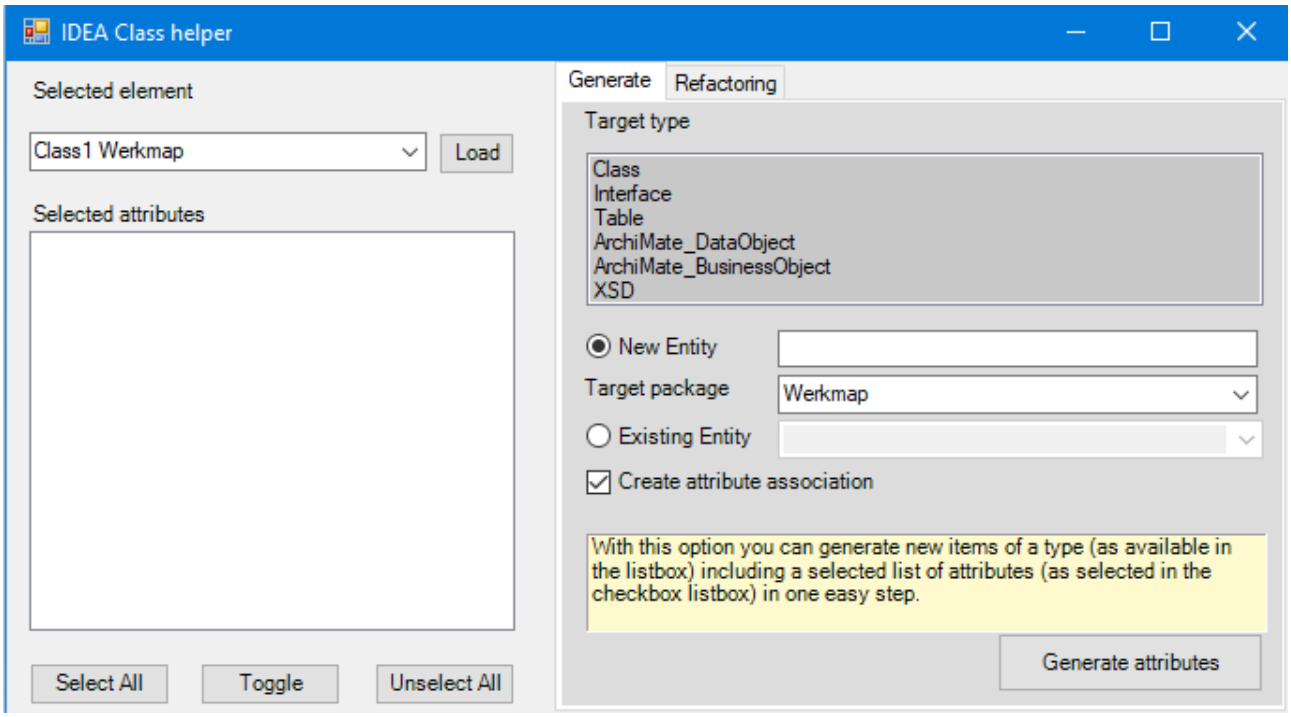
Duplicate report shows a report first of the duplicates (this can be shared with your team members before deduplication

Start deduplicate deduplicates the repository

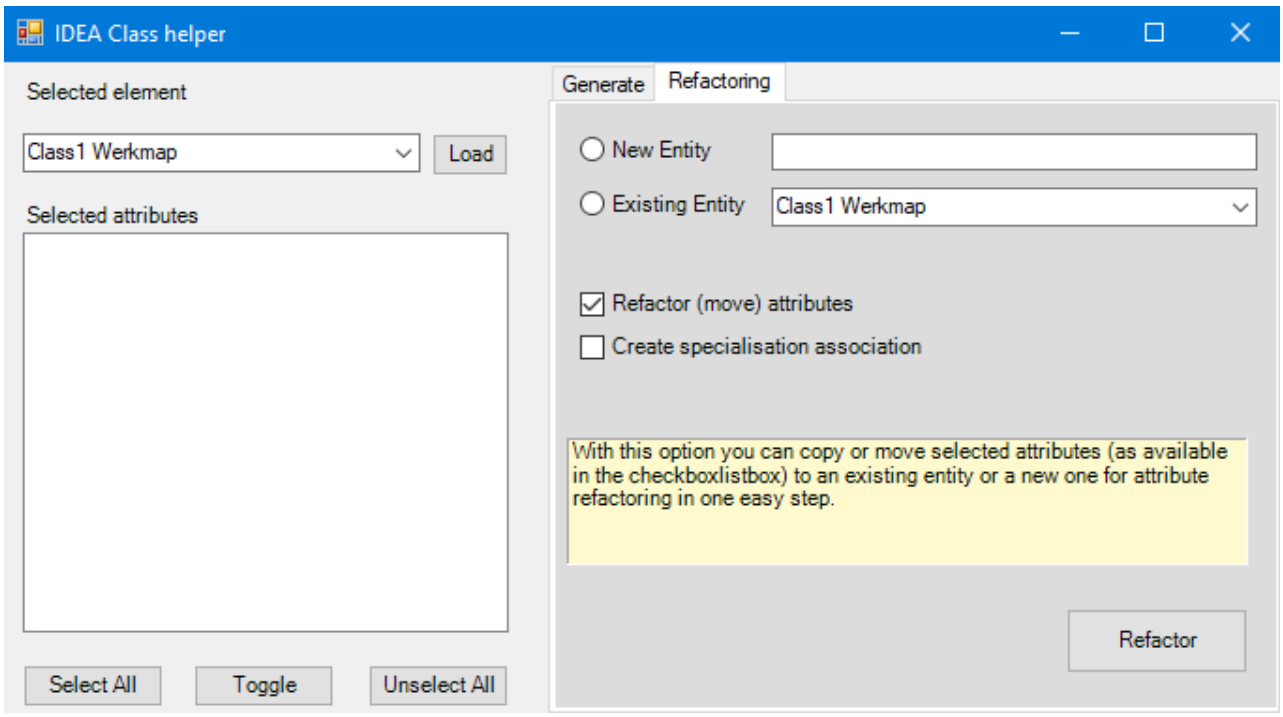
IDEA

IDEA is a combination of functionalities for data modeling and data architecture. It automates the tedious and repeating modeling activities.

IDEA Class helper



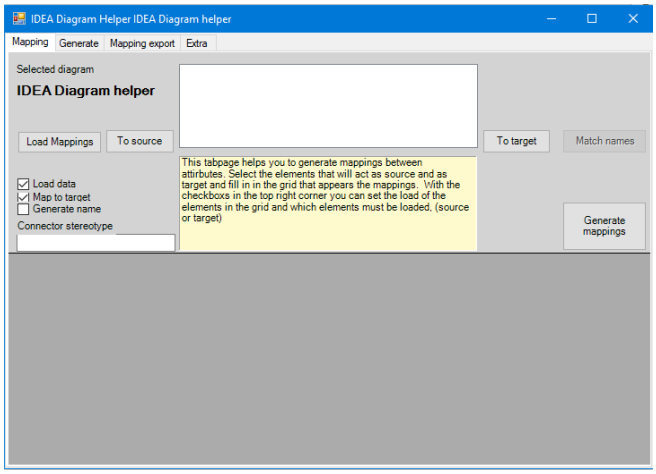
ClassHelperGenerate



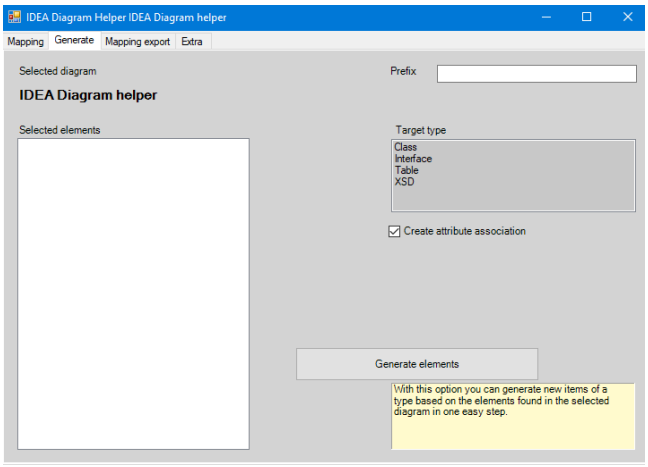
ClassHelperRefactor

The class helper is a specific screen for LDM modeling in UML class diagrams. When selecting a diagram and loading the attributes you can do a number of actions on the selected attributes.

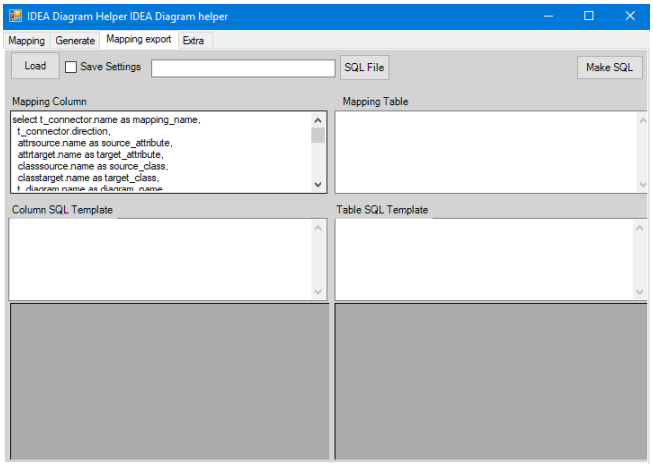
IDEA Diagram Helper



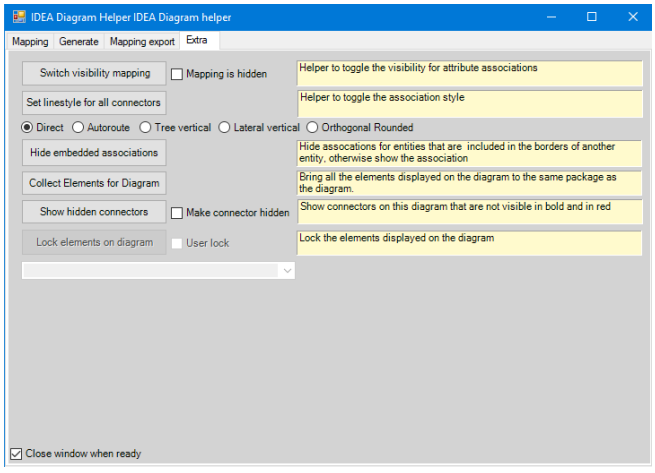
DiagramHelperMapper



DiagramHelperGenerate



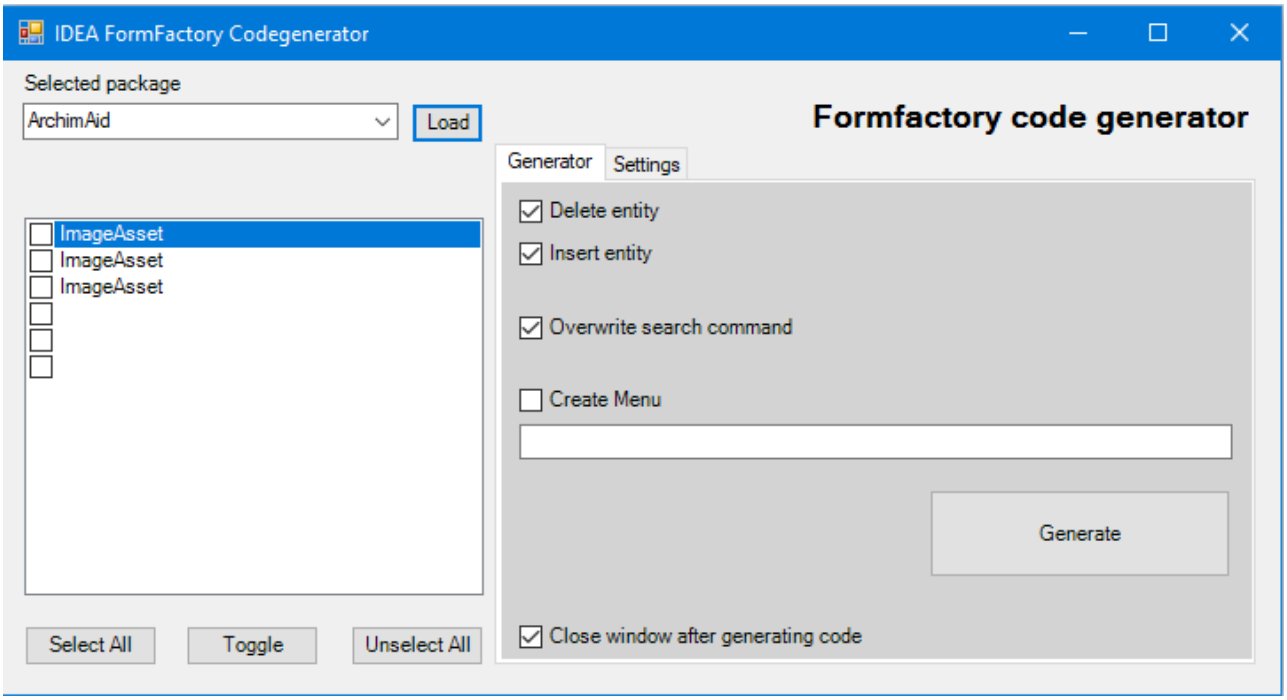
DiagramHelperMapperSQL



DiagramHelperExtra

The diagram helper is the most important screen in IDEA since most activities are done at diagram level in data modeling.

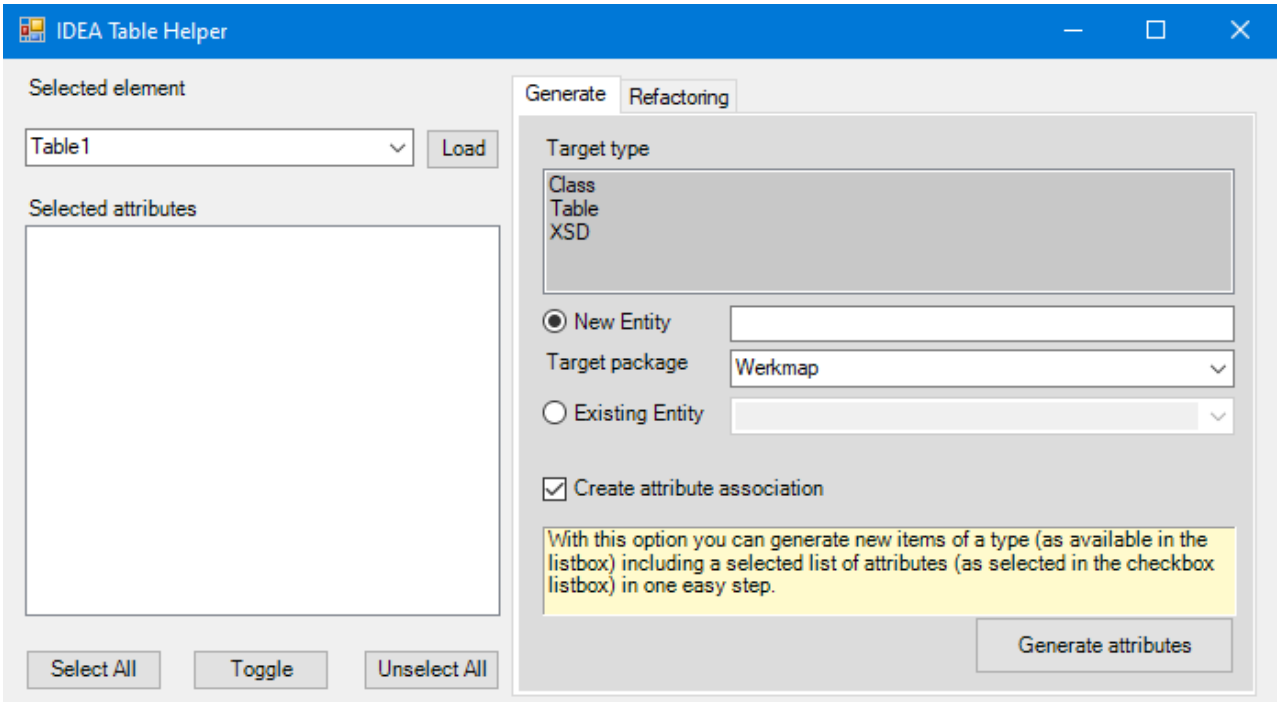
IDEA Package helper



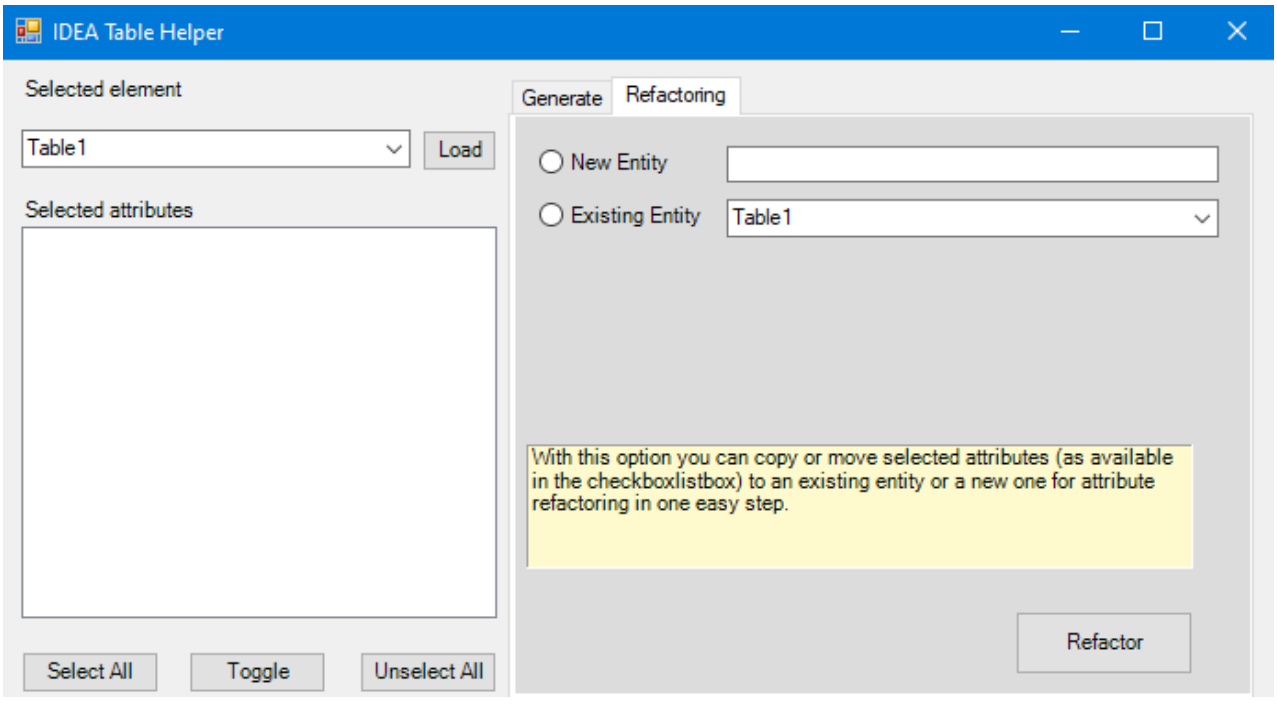
FormFactory

This screen is used for generating source code for the FormFactory Content Management System. This CMS is specific for EAxpertise and therefore the documentation is not described in detail

IDEA Table helper



TableHelperGenerate



TableHelperRefactor

The class helper is a specific screen for TDM modeling in data modeling diagrams. When selecting a diagram and loading the attributes you can do a number of actions on the selected attributes.

ClassHelperGenerate («Image»)
With the generate tab you can create a number elements in the TDM and the CDM.
In the generate you can bring the elements to an existing entity or create a new entity.

With create attribute association you create a trace at attribute level for navigation and modeling mappings in FDM - LDM modeling

ClassHelperRefactor («Image»)
Refactoring is a technique to move attributes in your datamodel. With this screen this is made easy.

It has two features:

- Refactoring is moving attributes in stead of copying the attributes
- Create a specialisation association for speciializing or generalizing elements

DiagramHelperExtra («Image»)
This is a number of extra functionalities that makes data modeling easier. The explanation of the functions is given in the yellow boxes.

DiagramHelperGenerate («Image»)
With this tab page you can select a number of entities on the diagram and create a new entity based on the target type listbox.

When create attribute associations there are mapping associations created on the background.

With the prefix field you can add a number of characters to the name of the new elements. Relevant for organisations with naming conventions

DiagramHelperMapper («Image»)
The mapping tab in the diagram helper makes it possible to define a grid of source and target attributes and define a mapping association between these attributes. It can be used for mapping in a layer (LDM or TDM) and for creating mappings between LDM and TDM.

There are a number of extra functions and features:

- Load mappings is used to load the existing mappings in the grid after the source and target attributes are loaded
- Match names is a routine to create mappings between source and target attributes with the same name (useful for DataVault modeling)
- Load data is default active, when not active only the comboboxes in the grid are loaded and not the grid
- Map to target is default checked, when not check the data in the grid is loaded based on the source attributes (useful for creating mergers)
- Generate name when checked creates a mapping with the name of the attribute
- With connector stereotype you can define the stereotype of the association.

DiagramHelperMapperSQL («Image»)
This tab page makes it possible to load data for the elements and attributes or tables and columns and create a SQL file for the SQL statements.

It makes it possible to create a specific SQL statement for the table and the columns. For more information at NS there is a specific powerpoint presentation describing the usage of this tabpage.

TableHelperGenerate («Image»)
With the generate tab you can create a number elements in the TDM and the LDM.
In the generate you can bring the elements to an existing entity or create a new entity.

With create attribute association you create a trace at attribute level for navigation and modeling mappings in FDM - LDM modeling

TableHelperRefactor («Image»)
Refactoring is a technique to move attributes in your datamodel. With this screen this is made easy.

It has two features:

- Refactoring is moving attributes in stead of copying the attributes
- Create a specialisation association for speciializing or generalizing elements

Begrippenlijst

Begrip	Definitie
Analytics	Analytics is the systematic computational analysis of data or statistics.[1] It is used for the discovery, interpretation, and communication of meaningful patterns in data. It also entails applying data patterns towards effective decision making. It can be valuable in

Begrip	Definitie
	areas rich with recorded information; analytics relies on the simultaneous application of statistics, computer programming and operations research to quantify performance.
Big Data	Big data is a field that treats ways to analyze, systematically extract information from, or otherwise deal with data sets that are too large or complex to be dealt with by traditional data-processing application software. Data with many cases (rows) offer greater statistical power, while data with higher complexity (more attributes or columns) may lead to a higher false discovery rate.[2] Big data challenges include capturing data, data storage, data analysis, search, sharing, transfer, visualization, querying, updating, information privacy and data source. Big data was originally associated with three key concepts: volume, variety, and velocity. When we handle big data, we may not sample but simply observe and track what happens. Therefore, big data often includes data with sizes that exceed the capacity of traditional software to process within an acceptable time and value.
Data warehouse	In computing, a data warehouse (DW or DWH), also known as an enterprise data warehouse (EDW), is a system used for reporting and data analysis, and is considered a core component of business intelligence.[1] DWs are central repositories of integrated data from one or more disparate sources. They store current and historical data in one single place[2] that are used for creating analytical reports for workers throughout the enterprise.[3]