

# SQL to UML transformation

SQL models and diagrams will be transformed into the platform-independent UML models and UML class diagrams. You can apply the SQL to UML transformation to SQL models of any database flavor.

## Type mapping

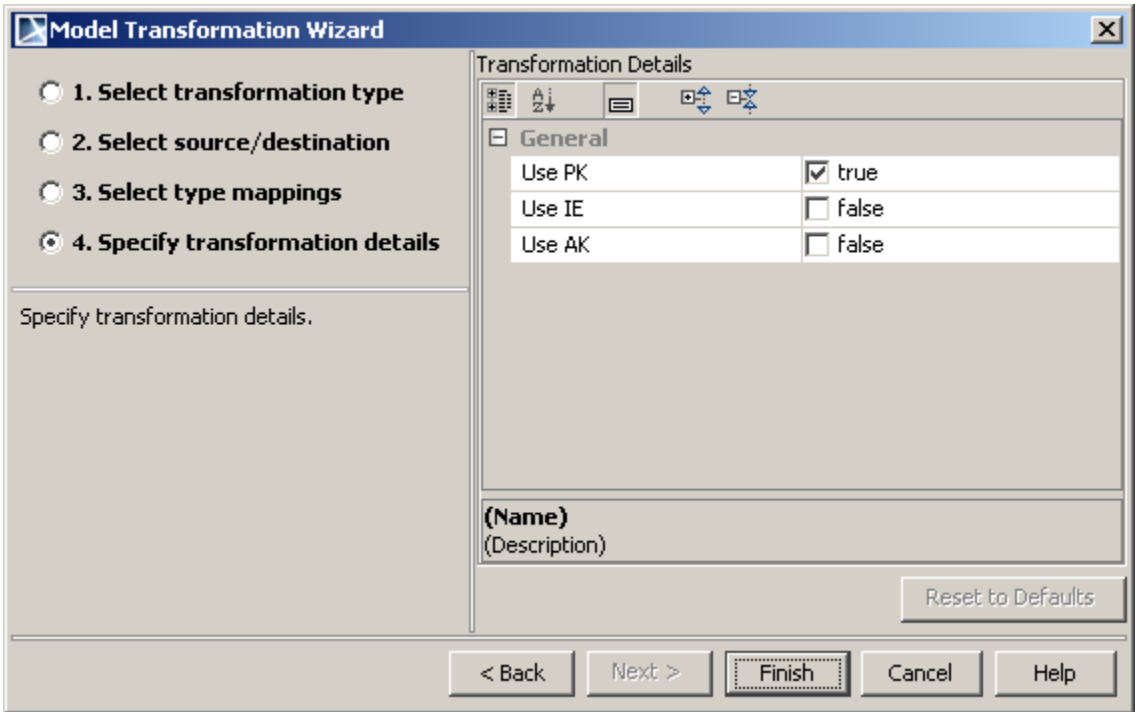
After transformation, SQL types should be converted to UML types if types are specified in the SQL model for elements. There is a type mapping from SQL types to UML types.

Mapping rules are based on dependencies, which contain the **SQL to UML Type Map** profile. This profile is automatically attached when SQL to UML transformation is performed.

## Transformation results

The SQL stereotypes are discarded from tables, views, fields, and associations (except the PK stereotype). Views are discarded in a transformed class diagram.

There are additional properties to choose for SQL to UML transformation in **Model Transformation Wizard**. For more information about the wizard, see [Model Transformation Wizard](#) in the MagicDraw User Guide.



Model transformation wizard for SQL to UML transformation with Specify transformation details wizard step.

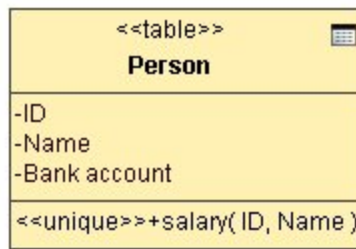
Option name	Type	Description
Use PK	Check box	If set to "true", appropriate columns with the primary key stereotype are marked after transformation.
Use IE	Check box	If set to "true", indexed columns with the inverted entity stereotype are marked after transformation.
Use AK	Check box	If set to "true", unique columns with the alternative key stereotype are marked after transformation.

The «IE» stereotype is applied to the columns in the UML model from *indexes* in the SQL.

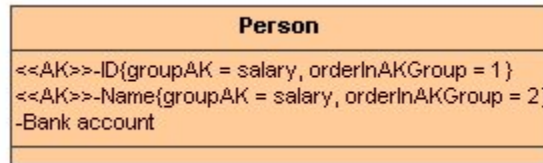
The «AK» stereotypes are applied to the columns in the UML model from *unique* constraints in the SQL.

If the *unique* or *index* of the SQL contains more than one column, the **group** tag is created on the corresponding columns. The value of the tag is the name of the *unique* / *index*.

If the PK, *unique* constraint or *index* of the SQL contains more than one column, the **orderInXXGroup** tag is created on the corresponding columns. The value of the tag is the place number of the column in the PK, unique constraint or index (first column gets tag value=1, second column - 2, etc).



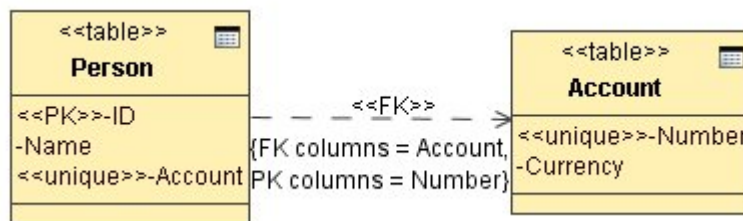
TBD 3 SQL before transformation.



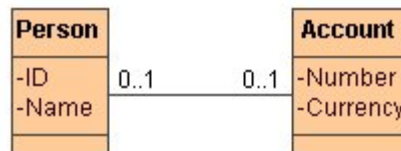
TBD 3 SQL after transformation.

There are some foreign key cases when an association with multiplicities is created in a class diagram after transformation, as shown below:

### Transforming foreign key, when the «unique» stereotype is set

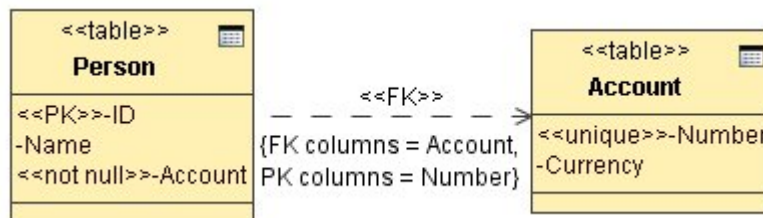


Transforming foreign key when the «unique» stereotype is set before transformation.

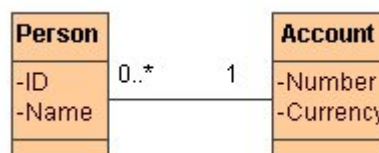


Transforming foreign key when the «unique» stereotype is set after transformation.

### Transforming foreign key, when the «not null» stereotype is set

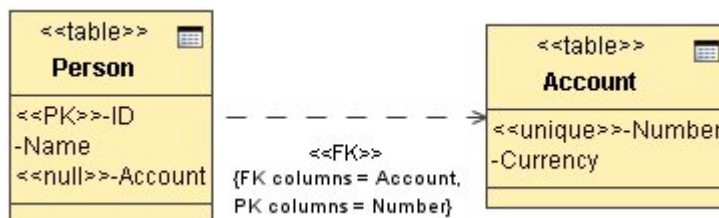


Transforming foreign key when the «not null» stereotype is set before transformation.

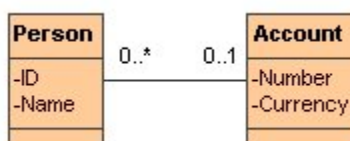


Transforming foreign key when the «not null» stereotype is set after transformation.

## Transforming foreign key, when the «null» stereotype is set

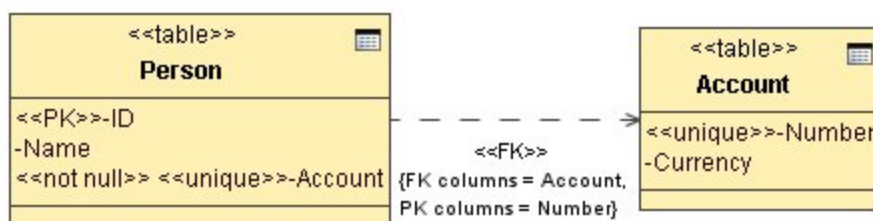


Transforming foreign key when the «null» stereotype is set before transformation.

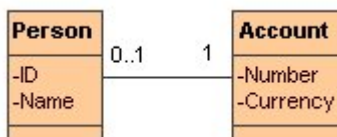


Transforming foreign key when the «null» stereotype is set after transformation.

## Transforming foreign key, when the «unique» and the «not null» stereotypes are set



Transforming foreign key, when the «unique» and the «not null» stereotypes are set before transformation.



Transforming foreign key, when the «unique» and the «not null» stereotypes are set after transformation.

### Related page

- [Model Transformation Wizard](#)