

# 30 Persistent Ids

Available since: 2017.01

## Introduction: why was this built and what does it do?

Any Delft-FEWS configuration is based on the linkages between unique IDs (identifiers for locations, parameters, qualifiers, module-instances etc.). In the early stages of the development of a configuration these IDs were chosen and from that moment on, these IDs are known in the database and cannot be 'renamed' anymore. While working with the configuration over time, it may happen that a chosen ID (in the past) turns out not to be very handy or consistent or may contain typos. For the readability and consistency of a configuration, functionality has become available to 'repair' inconsistencies in these IDs. The overall idea is that an (optional) configuration file can be added which contains a mapping between the required 'new' ID (= 'configuration-id') and the 'old' ID (= 'database-id'). Synonym for the 'database-id' is 'persistent-id'. When a new 'configuration-id' is chosen, this ID should be used in the complete configuration. This functionality will not 'rename' any ID in the database. It will leave the database intact. Since this functionality is a part of the configuration, it can also easily be rolled back (if needed). This functionality is compatible with the Deltares Open Archive.

The IDs in: LocationSets.xml, Parameters.xml, Qualifiers.xml and moduleInstanceDescriptors.xml can be overruled by a \*.CSV-file containing this mapping. When one of these files is used, make sure the mentioned configuration-id is used throughout the complete configuration.

## Use Cases: where to use this functionality for?

In the below mentioned cases this functionality can (not) be used.

Case	Description	Yes	No	Solution
1	A location-ID was changed by a data provider		X	Solve this by regular 'id-mapping' functionality for importing the data
2	I made a typo in a location-ID, parameter-ID or qualifier-ID	X		Create an overruling *.CSV file (for only those IDs with the typo) which is referred to in the LocationSets.xml, Parameter.xml or Qualifier.xml as in the examples below.  Change the rest of your configuration referring to this obsolete IDs as well.
3	My location-IDs are not consistent with the IDs used by my data-provider	X		Create an overruling *.CSV file (for only those IDs) which is referred to in the LocationSets.xml as in the examples below.  Change the rest of your configuration referring to this obsolete IDs as well.
4	A location-name is not up-to-date anymore		X	Solve this by editing the locations.xml or csv file and update the name attribute or field.
...	...			

## Configuration examples

### XML Files

The xml files to get this functionality to work should be appended with the <persistentIdsCsvFile> element.

#### Example for LocationSets.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<locationSets xmlns="http://www.wldelft.nl/fews"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.wldelft.nl/fews https://fewdocs.deltares.nl/schemas/version1.0/locationSets.xsd"
  version="1.1">
  <persistentIdsCsvFile>
    <file>persistentLocationIds.csv</file>
    <configId>%CONFIG_ID%</configId>
    <persistentId>%PERSISTENT_ID%</persistentId>
  </persistentIdsCsvFile>
  <locationSet id="csv">
    <csvFile>
      <file>locations.csv</file>
      <id>%ID%</id>
      <x>%x%</x>
      <y>%y%</y>
    </csvFile>
  </locationSet>
```

### Example for Parameters.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<parameters xmlns="http://www.wldelft.nl/fews"
             xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
             xsi:schemaLocation="http://www.wldelft.nl/fews https://fewsdocs.deltares.nl/schemas
/version1.0/parameters.xsd" version="1.0">
  <persistentIdsCsvFile>
    <file>persistentParameterIds.csv</file>
    <configId>%CONFIG_ID%</configId>
    <persistentId>%PERSISTENT_ID%</persistentId>
  </persistentIdsCsvFile>
  <parameterGroups>
    <displayUnitConversionsId>DisplayUnitConversions</displayUnitConversionsId>
```

### Example for Qualifiers.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<qualifiers xmlns="http://www.wldelft.nl/fews"
             xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
             xsi:schemaLocation="http://www.wldelft.nl/fews https://fewsdocs.deltares.nl
/schemas/version1.0/qualifiers.xsd">
  <persistentIdsCsvFile>
    <file>persistentQualifierIds.csv</file>
    <configId>%CONFIG_ID%</configId>
    <persistentId>%PERSISTENT_ID%</persistentId>
  </persistentIdsCsvFile>
  <qualifier id="a"/>
  <qualifier id="b"/>
</qualifiers>
```

### Example for moduleInstanceDescriptors.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<moduleInstanceDescriptors xmlns="http://www.wldelft.nl/fews" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://www.wldelft.nl/fews https://fewsdocs.deltares.nl/schemas/version1.0
/moduleInstanceDescriptors.xsd" version="1.0">
  <persistentIdsCsvFile>
    <file></file>
    <configId></configId>
    <persistentId></persistentId>
  </persistentIdsCsvFile>
  <!-- importing -->
  <moduleInstanceDescriptor id="ImportQMA">
    <moduleId>TimeSeriesImportRun</moduleId>
  </moduleInstanceDescriptor>
  <moduleInstanceDescriptor id="ImportMM3PCSV">
    <moduleId>TimeSeriesImportRun</moduleId>
  </moduleInstanceDescriptor>
```

## CSV file(s)

To make it still possible for Delft-FEWS to resolve the location/parameter/qualifier and module instances in the database or archive, you have to create CSV files with two columns that maps the config-id to the original persistent database/archive id. The overruling \*.CSV file(s) should look like this:

#### Example for a persistentIdsCsvFile

```
CONFIG_ID, PERSISTENT_ID
correct-id1, database-idA
correct-id1, database-idB
correct-id2, database-idC
...
```

the *correct-id\** is the ID you want to use in the configuration from now on. The *database-id\** is the persistent-id in the database.

When data is written to the database / archive the initial old persistent-ids are still used instead of the updated config-id. You only have to list the config-id and persistent-id when they are different from each other. You can split the CSV files if you want.

**IMPORTANT REMARK: When one of these files is used, make sure the mentioned configuration-id is used throughout the complete configuration!!**