

Cube Load with Drillthrough

Related link: [Cube Load](#)

For huge data models, it may not be convenient to load the data at the most detailed level to an OLAP cube. However, access to this detailed data is required for the analysis. This is the scenario for drillthrough: the data are stored at an aggregated level in a cube. With a drillthrough request on a cube cell from the front end, the detailed data, including additional fields (e.g. Order Number, Line Item ID), can be displayed.

Drillthrough can be enabled in the **Advanced settings** area of a cube load. Select the desired [Drillthrough Mode](#) from the drop-down list, as shown below.

^ Advanced settings

Cube layout change mode: ▾

Deactivate while Loading:

Cube access

Rules calculation

SVS event

Bulk size:

Settings for drillthrough

Drillthrough mode: ▾

Relational connection: ▾

Optional schema: ▾

Optional table: ▾

Relational extract: ▾

Definition of Annex Fields

The columns of the source that should be part of the detailed drillthrough data request (but not part of the cube) have to be defined. This can be done in the Dimension Mapping table by assigning the entry "" instead of a dimension name to the particular column. It is not necessary for these additional fields to be at the end of the source. The order of the columns in the source defines the display order.

Drillthrough Modes

None The default setting for cube load is None, which means that drillthrough is disabled. To enable drillthrough, select one of the drillthrough modes described below.

Direct No additional persistence of the detailed drillthrough data is done during the cube load. The requirement for this is that all detailed data can be retrieved via one relational extract without any further transforms. Only the link from the OLAP cube to this relational extract is stored in Jedox Integrator.

Relation extract (for direct drillthrough only): all detailed data can be retrieved through a relational extract during a drillthrough request. It may contain an arbitrary SQL statement. The relational extract may have a field structure with a column mapping, but it must not contain default values for the columns.

Persisted During the cube load, the detailed drillthrough data, including the additional fields, is additionally persisted to a relational database. The link from the OLAP cube to this relational table is stored in Jedox Integrator.

Options for persisted drillthrough only:

Relational connection: A relational database connection where the detailed drillthrough data will be stored. They are recommended for very huge data volume. The same databases are supported as in Relational Load (e.g. Oracle or MS SQL Server) with the exception of MS Access. If no connection is indicated, the data are written to a filebased H2-database which is used internally in Jedox Integrator.

Optional schema: The name of the schema in the Relational database. Default: The name of the OLAP database

Optional table: The name of the table in the Relational database where Drillthrough data will get stored. Default: The name of the OLAP cube.

Differences between Persisted and Direct Drillthrough

	Persisted	Direct
Data transformations	Allowed	Not allowed (only in SQL query)
Performance	Slightly higher query performance (if direct drillthrough with complex SQL query)	Higher loading performance, as no loading to relational persistence
Data accuracy	Consistent with cube data if the relevant cube slice is always loaded in a Cube Load with activated Drillthrough (and the same Connection, Schema and Table)	Always up to date but potentially different from cube data. Consistency depends of the correct definition of the SQL query in the Relational Extract used for Drillthrough.
Possible systems for source data	Any	One relational database system

Needed requirements	<u>With external database:</u>	The Jedox Integrator project and the relational extract specified in cube load must not be removed or renamed in order to find the correct data in a drillthrough request.
	The Jedox Integrator project and the relational connection specified in the cube load must not be removed or renamed in order to find the correct data in a drillthrough request.	
	<u>With internal database:</u>	
	The Jedox Integrator project is not required.	

Notes

- Drillthrough requests are possible with Jedox Excel Add-in and Jedox Web. For the configuration of the drillthrough, you need to install and configure [Jedox Supervision Server](#).
- To retrieve the drillthrough data via Integrator, use the cube extract with advanced option “Retrieve drillthrough data”. You could also use the [SOAP API “drillthrough”](#), in which case no Supervision Server is necessary.
- For an example, see the Jedox Integrator Project *sampleDrillthrough*, which is included in every Jedox installation.
- If “Parallel Load in Database for Drill Through” is set and “Splash mode” is set to anything but “disabled”, there will be a warning message, because no splashing can be done on the relational Drillthrough data.
- It is not possible to run several cube loads with Drillthrough for the same cube in parallel jobs.
- Any manual changes to the cube slice relevant for

drillthrough and to the cube layout will lead to inconsistencies in the drillthrough request. The same goes for manual changes to the relational table used in Persisted Drillthrough.

- The link of a cube to a drillthrough table can be displayed and removed in the [Modeler](#).
- The default connection for Persisted Drillthrough is an internal filebased H2 database which is located in this directory:
<install_path>\tomcat\webapps\etlserver\data\db\etl_drillthrough. It can be configured in file
<install_path>\tomcat\webapps\etlserver\config\connections.xml
- The H2 database should only be used for a small or medium-sized data volume (filesize 4GB maximum). For huge data, an external client-server-based database should be used, e.g. Postgres. With H2 as persistence, parallel drillthrough requests are handled sequentially. See also the [H2 documentation](#).
- The cube load mode “insert” is not available in combination with Drillthrough, as it might lead to inconsistencies between the cube and the detailed Drillthrough data. Instead, the load mode “add” should be considered, possibly in combination with a preceding load mode “delete”. For a cube load *without* Drillthrough and a separate [relational load](#) for the Drillthrough data (using

relational load mode “insert” with “aggregation” option activated), it would be possible to achieve consistency but without linking the cube to the Drillthrough table.

- For Direct Drillthrough, a field structure should be defined in the corresponding Relational Extract (Button “Refresh” in table “Field Structure”). The column “Field name” may be filled or not. This increases the performance of the drillthrough request, especially in the case of complex SQL statements, as it omits an initial SQL query to the database to determine the output columns of the relational extract.
 - If a dimension element is not found, by default a warning is raised and the cube data won’t be loaded, although the drillthrough data will be persisted. To avoid this, the option “Handling of missing elements” could be set to “createUnderDefault” or “createUnderDefaultParent”.
-