RDB via Table Introduction

The RDB via Table connector crawls content from any relational database that can be accessed using JDBC.

- The connector extracts data based on SQL statements and submits this data into Aspire for processing
- The connector directly extracts the data. This means that there is not a *fetch data* phase. However, if your database includes references to external data (say URLs to web sites or paths of external files), then a *fetch* stage may be invoked.

The RDB via Table connector features include the following:

- Connects to a database server using JDBC drivers (which must be downloaded separately)
- Performs full crawling
- Performs incremental crawling, so that only new or updated documents are indexed, using tables to hold identifiers of updated content
- Fetches data from the database using SQL statements
- Is search engine independent
- Runs from any machine with access to the given database

Content Retrieved by the Connector

The content retrieved by the connector is defined entirely using SQL statements, so you can select all or subsets of columns from one or more tables. Initially, the data is inserted into Aspire using the returned column names, but this may be changed by further Aspire processing.

JDBC Drivers

The RDB via Table connects to databases via JDBC, so you'll need the appropriate JDBC client (driver) JAR file for the database you want to connect to. These are available for most (if not all) major database vendors, and your first port of call for the driver should be the vendor's website.

Operation Mode

The connector can operate in two modes: **full** and **incremental**.

Important: *The data submitted to Aspire by this connector is dependent entirely on the SQL that's configured. Therefore, it is quite possible to submit all of the data in an incremental crawl, or only some of the data in a full crawl.*

Full Mode

In full mode, the connector executes a single SQL database statement and submits each row returned for processing in Aspire.

Incremental Mode

In incremental mode, there are three stages of processing: preprocessing, crawling, and post-processing.

1 - Pre-processing

(Optional) This stage runs a SQL statement against the database that can be used to mark rows to crawl (i.e., they have changed since the previous run).

2 - Crawling

This stage (similar to full mode) executes a single SQL database statement and submits each row returned for processing in Aspire. Typically, the result set is a subset of the full data that may be filtered using information updated in the (optional) pre-processing stage.

3 - Post-processing

(Optional) Each row of data submitted to Aspire can execute a SQL statement to update its status in the database. This may be to reset a flag set in the (optional) pre-processing stage, thereby denoting the item as *processed*. Different SQL can be executed for rows that were successfully processed versus ones that were not.