IATI Tool: Documentation

IATI Secretariat

28 janvier 2026

IATI Tables

IATI Tables transforms data published to the International Aid Transparency Initiative (IATI) into relational tables. These data are available to users in multiple formats, including PostgreSQL, CSV, and a Datasette instance.

We recommend IATI Tables for people wanting to :

— query IATI data using SQL
— download all IATI data in relational table format
— analyse IATI data using notebooks.

CHAPITRE 1

Getting Started

IATI tables transforms IATI data into relational tables, and provides multiple ways to access these data.

1.1 IATI Tables Datasette

Datasette is an open source tool for exploring data. For more information on the Datasette tool, see the Datasette documentation.

The IATI Tables Datasette instance allows you to explore IATI data your browser using SQL, and download the results of your query in multiple formats. You can also query IATI Tables Datasette from notebooks, such as Google Colab or Deepnote.

1.2 CSV Zip

The CSV Zip download is a compressed folder containing a CSV file for each table, which you can explore by importing into a spreadsheet viewer such as Excel or Google Sheets. For more information, see *Spreadsheets*.

1.3 SQLite Zip

The SQLite Zip download is a compressed SQLite database, which can be run and explored with `sqlite3`.

1.4 PG Dump

The PG Dump download can be loaded into a PostgreSQL database and are created by the pg_dump utility.

There are two options to choose from, gzip or custom :

Gzip

The "gzip" format gives you a compressed plaintext script of SQL commands, which can be restored using `psql`.

Custom

The "custom" format can be restored using the pg_restore utility. This option is more flexible if you want to perform any schema changes before restoring.

CHAPITRE 2

Spreadsheets

To work with all IATI data in a spreadsheet program such as Microsoft Excel or Google Sheets, you can use the CSV Zip Download. You can also prepare and filter IATI data using IATI Tables Datasette, and download the results in CSV format.

When working with IATI Tables data in spreadsheet programs, we recommend the following :

2.1 Disable Auto-Formatting

When opening CSV files, spreadsheet programs try to automatically assign types to each column, for example for numbers and dates. Auto-formatting removes leading zeros from numerical identifiers, for example transforming the identifier `00010` into `10`.

To prevent these incorrect transformations, disable auto-formatting when loading CSV data into spreadsheets. For example, in Google Sheets, take the following steps :

1. Create an empty spreadsheet
2. Select `File > Import`
3. Upload the file you want to import
4. On the `Import file` screen, uncheck the box labelled « Convert text to numbers, dates and formulas »
5. Click `Import data`

CHAPITRE 3

Data Retrieval

This section outlines how IATI data is retrieved by IATI Tables.

3.1 Data Source

IATI Tables retrieves a daily snapshot of all IATI data from the IATI Bulk Data Service. All data is replaced each time the process runs, so updates and removals are respected.

The metadata table shows the cut-off time for data to be included. The column `iati_tables_updated_at` shows the time at which the tables run finished. The column `data_dump_updated_at` shows the time of the Bulk Data Service snapshot that was used by the latest run. IATI data published or edited after this time will not be included in IATI Tables until the next run.

CHAPITRE 4

Data Processing

This section outlines how IATI data is processed by IATI Tables.

4.1 Flattening

4.1.1 Top-level elements

There are two top-level tables in IATI Tables :

1. The `activity` table is generated from Activity files, and contains rows of iati-activities/iati-activity elements.
2. The `organisation` table is generated from Organisation files, and contains rows of iati-organisations/iati-organisation elements.

All other tables are children of these top-level tables. Tables prefixed with `organisation_` contain child elements of the iati-organisations/iati-organisation element, and can be joined back to the `organisation` table using the column `_link_organisation`. The remaining tables contain child elements of the iati-activities/iati-activity element, and can be joined back to the `activity` table using the column `_link_activity`.

4.1.2 Singular child elements

Child elements which can appear zero or one times become columns in the parent table.

For example, the iati-activities/iati-activity/iati-identifier element becomes the column `iati-identifier` in the `activity` table.

4.1.3 Repeatable child elements

Child elements which can appear more than once are unnested into a new table.

For example, the iati-activities/iati-activity/transaction element becomes the table `transaction`.

4.1.4 Narrative elements

Elements which contain a `narrative` element are flattened into a single string column in the parent table.

For example, given the following iati-activities/iati-activity/title element :

```
<title>
    <narrative>Activity title</narrative>
    <narrative xml:lang="fr">Titre de l'activité</narrative>
    <narrative xml:lang="es">Título de la actividad</narrative>
</title>
```

This element will be transformed into the string : « Activity title, FR : Titre de l'activité, ES : Título de la actividad », and stored in the column `title_narrative` in the `activity` table.

4.2 Common columns

The following columns are present in all tables :

`_link`
: The primary key for each table.

`_link_activity` or `_link_organisation`
: The foreign key to the `activity` or `organisation` table respectively.

`dataset`
: The name of the dataset this row came from. This can be used to find the dataset in the IATI registry, using the URL : `https://www.iatiregistry.org/dataset/<DATASET_NAME>`.

`prefix`
: The registry publisher ID this row came from. This can be used to find the dataset in the IATI registry, using the URL : `https://www.iatiregistry.org/publisher/<PREFIX>`.

4.3 Codelists

Codelists are joined to the tables as part of the process.

For example, given the iati-activities/iati-activity/activity-status element, whose attribute `@code` uses the ActivityStatus codelist :

```
<activity-status code="2" />
```

This element will be transformed into two columns in the `activity` table :
- The column `activitystatus_code` with the value `2`.
- The column `activitystatus_codename` with the value `Implementation`.

4.4 Currency Conversion

IATI Tables converts the iati-activities/iati-activity/transaction/value element to USD.

This is calculated by pulling the Code for IATI IMF exchange rates dataset, and using the transaction's `@value-date` attribute to determine the exchange rate to use.

The result is stored in the transaction (`trans`) table in a new column `value_usd`.

4.5 Transaction Splitting

IATI Tables transforms the `transaction` (trans) table to proportionally split the transaction values across multiple sectors or countries/regions.

It follows the same methodology set out in the CDFD documentation.