IATI Tool: Documentation

IATI Secretariat

Jan 29, 2026

DATASTORE SEARCH

The International Aid Transparency Initiative (IATI) is a global effort to make data about development and humanitarian aid more transparent. IATI data is formatted according to the IATI Standard.

IATI Datastore Search is a tool which allows users to search IATI activity data published by reporting organisations. The results can be downloaded in IATI XML, JSON, or CSV format.

Datastore Search includes:

Simple Search
: This search behaves like any familiar search engine. Users can search the natural language elements of IATI data for any words or phrases of interest.

Advanced Search (using the "Switch to Advanced Search" button)
: This search allows users to make more complex queries using any IATI element.

Datastore API
: Users can also query IATI data using their own code via the API.

CHAPTER

ONE

DATA SOURCE

IATI Datastore Search provides a user-friendly online interface to the IATI Datastore.

The IATI Datastore is a repository of IATI activity data published by reporting organisations. In IATI data, an activity describes an individual development cooperation activity or project.

To be included in the IATI Datastore, data must:

- Be registered with IATI, and available via the IATI Bulk Data Service.
- Conform to the IATI XML Schema.
- Conform to the IATI Standard V2.0 or above.

CHAPTER

TWO

FREQUENTLY ASKED QUESTIONS

1. **Why is my data missing from the Datastore?**
   The IATI Datastore contains all available data registered with IATI which is version 2.0 or above, and conforms to the IATI XML Schema. Files that do not conform to the IATI XML schema are marked as critically invalid in the IATI Validator.
2. **When is the IATI Datastore updated?**
   The Datastore is updated dynamically as new data becomes available via the IATI Bulk Data Service. New files can take up to 24 hours to update.
3. **How do I search for a specific phrase/identifer?**
   To search for a particular phrase, use quotation marks. For example, **"rabbit production"** will return search results about rabbit production. **rabbit production** will return results about rabbits, and results about production. For more information on how to use the Datastore Search, please see *Simple Search: User Guide*.
4. **How does IATI Datastore clean IATI data?**
   The IATI Datastore indexes and represents IATI data precisely as it is was published. There are no transformations or layers of inferred meaning or metadata.

CHAPTER

THREE

SIMPLE SEARCH: USER GUIDE

The home page of Datastore Search contains the "Simple Search" bar. This performs a text search on every narrative element within IATI activity data.

For a summary of all elements in the standard, please see the Activity Standard Summary Table.

3.1 Search operators

Simple search includes search operators; special commands that make your search more precise. These operators and examples of their use are documented below:

Search operator	Function	Example
?	Wildcard, representing one character.	**latin?** returns results including latino and latina.
*	Wildcard, representing any number of characters.	**latin*** returns results including latin, latino, latina, latinize, etc.
“ ”	An exact match search.	**“inter-agency”** searches for inter-agency exactly. Without quotes, inter-agency would return matches to inter and agency.
AND	Boolean, requires both terms to be present. && can be used in place of AND.	**Congo AND Democratic** returns results that contain both the terms Congo, and Democratic.
NOT	Boolean, requires that the following term is not present. ! or - can be used in place of NOT.	**Congo NOT Democratic** returns results that contain Congo but do not contain Democratic.
OR	Boolean, requires that at least one of the terms is present. || can be used in place of OR.	**Covid OR vaccine** returns results containing Covid, vaccine, or both.
+	Boolean, requires that a term must be present.	**foreign domestic +aid** returns results that contain foreign or domestic, and must include aid.
~	Fuzzy search, finds similar results. Can be modified with a number (0,1, or 2) to limit the number of edits. **OR**	**roam~1** returns road, roma, etc.
~	Proximity search, requires that provided terms are found separated by a maximum count of words. **Note:** quote marks must be used for the search phrase.	**“protection rights” ~20** returns results that contain protection and rights, within 20 words of each other.
^	Term boosting operator, boosts the relevancy of the term.	**covid^4 vaccine health doctor** returns results that match any of the four terms. Results containing covid will be higher ranked.
()	Grouping, for use with other search operators	**(Congo AND Democratic) OR (Covid AND vaccine)** combines two AND searches.

3.2 Downloading Results

You can download the results of your search in different levels of detail and formats.

Levels:

- **Activity level:** each row in the data represents one activity. This can contain many transactions and budgets.
- **Transaction level:** each row represents one transaction. Activity level information is repeated for each transaction.
- **Budget level:** each row represents one budget. Activity level information is repeated for each budget.

Formats:

- **XML:** A valid IATI standard XML file containing all the results from your search. This format can only describe activity-level data.
- **JSON:** A flattened Javascript representation of your search results.
- **CSV:** A flattened, comma separated export of your search results. Fields with multiple values are combined into a comma separated list. To note: in this format, cell lengths can exceed the Microsoft Excel limit of

32,767 characters.

- **EXCEL:** A flattened, comma separated export of your search results, modified to be compatible with Microsoft Excel. Fields with multiple values are combined into a pipe (|) separated list. Cell values longer than 32,767 characters are truncated to 32,700 characters.

3.3 Advanced Search

If you want to expand a simple search to include non narrative elements, click the "Advanced" button on the results page. This will open the advanced search menu, auto-populated with your existing search. See *Advanced Search: User Guide* for more information.

CHAPTER

FOUR

SIMPLE SEARCH: EXAMPLE QUERIES

4.1 1: How do I investigate which countries have had SMART surveys?

A data user wants to investigate which countries have had Standardised Monitoring and Assessment of Relief and Transitions (SMART) surveys conducted.

A suitable query would be **"smart survey" ~5**. This will search every narrative IATI element for the words smart and survey, within 5 words of each other.

The user can explore the results by clicking on individual activities titles. This will take them to a basic summary page for the activity, with the options to download it or view it on d-portal.

If they are happy with the query, the user can download the full set of results in the EXCEL format, at Activity level. Scrolling across to the "recipient_country" and "recipient_region" columns will show the countries and/or regions that benefit from this activity.

If recipient countries and regions are missing for some activities, this may be because they have been declared for individual transactions. The user can find this information by downloading the results at transaction level.

4.2 2: How do I find activites related to diabetes?

A researcher wants to find IATI activities with a focus on diabetes.

They can use the query **diabet*** to search all narrative IATI elements for relevant activities. This uses the wildcard search operator *, meaning it will return results including the words diabetic and diabetes.

The researcher could make this query more specific by using the advanced search. For example, by filtering to activities with the DAC 5 digit sector code 12182, for medical research.

CHAPTER

FIVE

ADVANCED SEARCH: USER GUIDE

To access the advanced search, click on the "Switch to Advanced Search" button on the Datastore Search homepage. This tool can be used to search all elements within IATI activity data.

For a summary of all elements in the standard, please see the Activity Standard Summary Table.

Important

We recommend reading the "Simple Search" section of this guide first. The simple search documentation includes important information on search operators and download formats.

5.1 Building queries

Click on the "Build Query" button in the advanced search menu to begin building your search query.

Fig. 1: The advanced search menu.

Next, click "Add Rule", and use the dropdown to select a field to search against. "Standard fields" map directly to elements of the IATI standard. You can then enter the value you want to search for.

For some fields (e.g. transaction dates) you can look for dates less than, equal to, or greater than the entered date using the **<**, **=**, and **>** operators. For other fields (e.g. recipient-country codes) use the **==** and **!=** operators to find codes equal to or not equal to the selected code.

For more complex queries, you can add more rules or groups of rules. See the Example Queries page for examples.

When your query is ready, click "Run" to excute it. You can also import and export queries to rerun at a later date.

Tip

Quote marks must be used to search for organisation and iati-identifiers. For example, "XM-DAC-1234". Without quotes, **iati-identifer == XM-DAC-1234** would return matches to XM, DAC, and 1234.

5.2 Special fields

In addition to standard fields, advanced search includes special fields. These are:

All Narratives: All narrative elements within IATI activity data. This searches the same elements as in the Simple Search.

Geospatial Search: Searches for results in a given geographical region using the location element. Click "Open map", and pan and zoom the map to define the search area. You can also hold the shift key and click and drag to select the search area.

5.3 Multi-valued fields

If an element can occur multiple times within an IATI activity, it is stored as a multi-valued field. This is important to consider when using these elements in a query. For example:

Participating organisation type

The query **Participating Org Type == 11 - Local Government** returns all activities with at least one type 11 participating organisation. Other participating organistions referenced in the activity may not be of type 11.

The query **Participating Org Type != 11 - Local Government** returns all activities with no type 11 participating organisations.

Transaction type

The query **Transaction Transaction Type Code == 1 - Incoming Funds** returns all activities with at least one incoming fund transaction. Other transaction types may be included in the activity.

However, the transaction-level download can be used to only download incoming fund transactions from these results. See the simple search user guide for more information on download options.

5.3.1 Element Wise Searching

It is not possible to query multi-valued fields in an element-wise fashion.

For example, consider the query: **Recipient Country Code == AF - Afghanistan AND Recipient Country Percentage == 30**.

This returns all activities with a recipient country code of AF, and a recipient country with a percentage of 30%.

However, this does not mean that all the activites will have 30% going to Afghanistan.

The 2 clauses may not necessarily apply to the same recipient-country element. So there could be an activity with 30% going to Pakistan and 70% to Afghanistan in the results.

In some cases you may be able to work around this limitation by downloading results at the transaction or budget-level. This is because a single transaction can only have a single recipient-country, for example.

CHAPTER

SIX

ADVANCED SEARCH: EXAMPLE QUERIES

6.1 1: How do I find activities reported by the FAO?

A researcher is interested in agricultural funding in IATI data, spefically in activities reported by the Food and Agriculture Organization of the United Nations (FAO). They visit the IATI Dashboard and determine that the organisation identifer of FAO is XM-DAC-41301.

They therefore need to search for activities with the reporting organisation reference XM-DAC-41301, using the query **Reporting Org Ref == "XM-DAC-41301"** . Note the quotation marks around the organisation identifer.

Fig. 1: Query 1: Activities reported by the Food and Agriculture Organization of the United Nations

If the researcher wanted to expand this search to include other reporting organisations, such the United States Department of Agriculture (US-GOV-2), they can use a comma separated list of organisation identifiers.

For example, **Reporting Org Ref == "XM-DAC-41301", "US-GOV-2"**. You can `download this query` and test it out yourself by using the "Import Query" option in advanced search.

6.2 2: How do I find organisations involved in the humanitarian response in Lebanon?

A Local NGO is looking to coordinate with other organisations. They want to find a list of organisations who are supporting the humanitarian response in Lebanon.

First, they need to find IATI activities flagged as "Humanitarian" which list Lebanon as a recipient country. Both of these codes can be declared at activity or transaction level, so they need to create a grouped query.

- Group A. will look for Humanitarian flags, declared at activity **OR** transaction level.
- Group B. will look for the recipient-country code LB for Lebanon, declared at activity **OR** transaction level.

These groups are combined with the **AND** group operator, so the search returns results with the Humanitarian flag, **AND** Lebanon as a recipient.

This creates the query **(Humanitarian == TRUE OR Transaction Humanitarian == TRUE) AND (Recipient Country Code == LB - Lebanon OR Transaction Recipient Country Code == LB - Lebanon)**. You can `download this query` and test it out yourself by using the "Import Query" option in advanced search.

Fig. 2: Query 2: Humanitarian activities in Lebanon

Next, they need to find organisations involved in the activities. They download the output in the EXCEL format, at Activity level, and open the file in their preferred spreadsheet software. All of the data about the IATI activities will be flattened into a single spreadsheet.

There are two places to look for organisations at the activity level:

1. **Reporting Organisations**
   A reporting organisation is the organisation reporting an IATI activity. Their names are in the "reporting_org_narrative" column. Filtering using a "remove duplicates" option will give a list of all reporting organisations.
2. **Participating Organisations**
   Participating organisations are all organistions involved in an activity. Their names are in the "participating_org_narrative" column. This will often contain multiple participating organistion names, separated by a pipe (|) character. Using a "split text to columns" option and specifying | as the separator will split the names into individual cells. These can then be combined into a single column and deduplicated as above.

Note - some duplication may remain as different names can be used for the same organisation. For example "World Health Organisation" and "WHO".

The NGO could also download the Transaction level file to investigate receiver and provider organistions.

6.3 3: How do I find activites related to diabetes in Botswana?

A researcher wants to find IATI activities with a focus on diabetes in Botswana.

They need to find IATI activities which mention diabetes in the title or description, and list Botswana as a recipient country. Recipient country codes can be declared at activity or transaction level, so they need to create a grouped query.

- Group A. will search for phrases related to diabetes in all narrative IATI elements, using the query **diabet***. This uses the wildcard search operator *, meaning it will return results including the words diabetic and diabetes.
- Group B. will look for the recipient-country code BW for Botswana, declared at activity **OR** transaction level.

Group B is nested in Group A with the **AND** operator, so the search returns results which mention diabetes, **AND** have Botswana as a recipient.

This creates the query **All Narratives include diabet* AND (Recipient Country Code == BW - Botswana OR Transaction Recipient Country Code == BW - Botswana)**. You can `download this query` and test it out yourself by using the "Import Query" option in advanced search.

Fig. 3: Query 3: Activities related to diabetes in Botswana

CHAPTER

SEVEN

API

An API is available with methods to query the datastore from your code and get results programatically.

To use this, you must sign up for an API key.

The Developer website allows you to get a key and gives you details of the API methods you can call and their parameters.

The rest of this page describes details of the API and should be read alongside the Developer website.

7.1 Field names and content in activity collection

Field names are constructed from the the XML tag and (if applicable) the attribute. For example field names you can query include:

- sector_percentage
- sector_vocabulary
- sector_code
- sector_narrative

These all come from the sector element of the standard.

As the sector element can appear multiple times in an activity, the contents of these fields can be a list of all the values that appear in the activity.

Here is an example:

- *"sector_code":["16050","3","10"]*
- *"sector_percentage":[100.0,50.0,50.0]*
- *"sector_vocabulary":["1","7","7"]*

It is not possible to tell from these lists which element in one field applies to an element in another list.

For example, you can't tell which code is from which vocabulary or which percentage applies to which code. (You may be able to guess from context in this example, but you can't tell for sure).

7.2 Field names and content in transaction and budget collection

In these collections, one data item is returned for each transaction or budget.

Any field names that start with the name of that collection come from that item only.

Any other field names come from the actvity and include all data in the activity. This makes it easy to construct complex queries.

For example, on an activity collection result you may see these fields:

- *"transaction_value":[10954.0,14086.0]*
- *"transaction_value_value_date":["2023-03-05T00:00:00Z","2023-08-15T00:00:00Z"]*
- *"sector_code":["16050","3","10"]*
- *"sector_percentage":[100.0,50.0,50.0]*
- *"sector_vocabulary":["1","7","7"]*

However on the transaction collection you will get one result with:

- *"transaction_value":[14086.0]*
- *"transaction_value_value_date":["2023-08-15T00:00:00Z"]*
- *"sector_code":["16050","3","10"]*
- *"sector_percentage":[100.0,50.0,50.0]*
- *"sector_vocabulary":["1","7","7"]*

And a second result with:

- *"transaction_value":[10954.0]*
- *"transaction_value_value_date":["2023-03-05T00:00:00Z"]*
- *"sector_code":["16050","3","10"]*
- *"sector_percentage":[100.0,50.0,50.0]*
- *"sector_vocabulary":["1","7","7"]*

7.3 Getting other data formats back from the activity collection

You can pass *iati_xml* to the *fl* parameter to get back a string of the XML of the activity.

You can pass *iati_json* to the *fl* parameter to get back a JSON representation of the XML, with structure.

Here is a truncated example with the same sector data as the above example:


```
{
"iati-activity": [
    {
    "@last-updated-datetime": "2024-11-27T11:23:09+01:00",
    "@xml:lang": "fr",
    "iati-identifier": [
        {
        "text()": "AB-123-123-123-123"
        }
    ],
    "sector": [
        {
        "@vocabulary": "1",
        "@code": "16050",
        "@percentage": "100"
        },
        {
        "@vocabulary": "7",
```


(continues on next page)

(continued from previous page)


```
        "@code": "3",
        "@percentage": "50"
        },
        {
        "@vocabulary": "7",
        "@code": "10",
        "@percentage": "50"
        }
    ]
    }
]
}
```


Unlike the flattened lists returned in previous examples, in this example it is possible to tell which sector code relates to which percentage and which vocabulary.