Article

← FME Desktop (/S/Topic/0TO4Q000000QL9uWAG/Fme-...

Tutorial: Getting Started with JSON

Aug 4, 2022 • Knowledge

Product Type
FME Desktop

FME Version
2022.0

Tutorial: *(Here)* | **Next:** Reading JSON (/s/article/json-reader-configuration)

Introduction

This tutorial series provides an overview of reading, writing, and transforming JSON data using FME.

JSON, or JavaScript Object Notation, is a file format that follows an open standard and has values that can be an object (attribute-value pairs such as { “a” : 5 }), an array (e.g. [1,2,3,4]), or a primitive (a string or number). It is widely used as a web exchange and messaging format. While it is popular on the web largely because it is the native object data structure in JavaScript, it is a language-independent data format.

FME Approach to JSON

A challenge with JSON data translation to/from tabular or GIS data is the fundamental difference in data modeling. JSON usually has a highly-nested tree structure, while tabular and GIS data usually use a relational database model composed of related tables. It may take several tables to model a given JSON object in a relational system.

FME can work with nested data, but its fundamental design lends itself more to working with flat tables composed of features and attributes. Therefore, bringing JSON data into FME requires considering the structure of JSON.

FME	Database	JSON
Feature	Record	Object

Attribute	Field	Property
List	Comma-delimited	Array

Terminology for data structures in FME, relational databases, and JSON

Consider this example of nested JSON data representing a point of geometry representing the location of San Francisco airport (SFO):

```
{
 "name" : "JSONFeature",
 "type" : "FeatureCollection",
 "features" : [
    {
      "ID" : "1001",
      "geometry" : {
        "type" : "Point",
        "coordinates" : [ -122.4194155, 37.7749295 ]
      },
      "properties" : {
        "name" : "SFO",
        "type" : "Airport"
      }
    }
 ]
}
```

JSON object example

If you are reading in JSON or composing a JSON message in FME, you might choose to store an entire JSON string as an attribute value:

_dataset

```
{
"name" : "JSONFeature",
"type" : "FeatureCollection",
"features" : [
{
"ID" : "1001",
"geometry" : {
"type" : "Point",
"coordinates" : [ -122.4194155, 37.7749295 ]
},
"properties" : {
"name" : "SFO",
"type" : "Airport"
}
}
]
}
```

If you planned to translate a JSON string into a relational database format or table, you could instead choose to “flatten” the JSON string into a table of FME features and attributes. Creating such a flattened JSON requires choosing an attribute from which to generate features, the unique values of which will each be a new feature, or row, in the resulting table. For example, flattening the above JSON string on the ID attribute would create the following features:

ID	geometry.type	geometry.coordinates	properties.name (http://properties.name)	properties.type
1001	Point	[-122.4194155, 37.7749295]	SFO	Airport

JSON object example represented in a table

FME offers flexible options for reading, writing, and transforming JSON data to get it in exactly the shape you need for your data integration workflow.

Articles

1. Reading JSON (/s/article/json-reader-configuration)

Learn how to read JSON data directly into your workspace using the JSON reader.

2. Extracting Location from JSON (/s/article/converting-from-json-to-a-spatial-format-gis)

Learn how to extract coordinate information from a nested JSON structure and use it to generate FME geometry.

3. Advanced JSON Reading (/s/article/json-reading-advanced)

Learn about more advanced JSON reading techniques, with an example of a multi-step operation where a URL is extracted from JSON and the data provided by the URL is read into the workspace, using the UK Environment Agency Real Time flood-monitoring API (https://environment.data.gov.uk/flood-monitoring/doc/reference).

4. Writing JSON (/s/article/json-writing-overview)

Learn about the options FME provides for writing to JSON and see a basic example using the JSON writer.

5. Writing JSON with the JSONTemplater (/s/article/json-writing-with-jsontemplater)

Learn how to write custom nested JSON with FME using the JSONTemplater transformer.

6. Transforming JSON using the JSONExtractor, JSONFlattener, and JSONFragmenter (/s/article/json-transformations)

Learn the use-cases for FME's most popular JSON transformers and how to use them.

First Published Date
7/29/2020, 12:18 AM

Last Published Date
8/4/2022, 1:58 AM

Transformation (/s/topic/0TO4Q000000...

FME Desktop (/s/topic/0TO4Q000000...

Sort by:

Latest Posts

LizAtSafe (/s/profile/0050c00000CeGV0AAN) (Employee) published a new version of this Knowledge.
August 4, 2022 at 1:58 AM (/s/feed/0D54Q00009ggVguSAE)

3 views

Like Comment

Log In to Comment

kailinatsafe (/s/profile/0054Q00000EwiZmQAJ) (Employee) published a new version of this Knowledge.
July 13, 2022 at 4:35 PM (/s/feed/0D54Q00009e1pVwSAI)

5 views

Like Comment

Log In to Comment

richardsr (/s/profile/0054Q00000EwvWRQAZ)
July 12, 2017 at 10:32 PM (/s/feed/0D54Q000080iRsHSAU)

Thank you for the clarification. After playing with it further it seems my experience it was impossible to edit the JSON Reader and to get the tree view. It seems to be only present at reader creation.

Alternatively at JSON reader creation, the tree browser option does not seem to work when authentication is required for the json service. Probably a support ticket but wanted to share why i struggled following the article.

Expand Post

2 comments 65 views

Like Comment

More comments 1 of 2

deanatsafe (/s/profile/00530000003Tn32AAC) (Employee)
3 years ago

Any time you have trouble getting the tree control to work with an online data source or feed, your best bet is to place the link in a web browser and download a sample JSON response locally. The local file can then be used to query the JSON with the tree control so you can find the correct JSON query, after which you can always point the reader back to the online source.
Note I just confirmed that in FME 2018 you can edit the 'Schema Scan Mode' and 'JSON Queries for Feature Objects' in the reader after adding the reader.

Expand Post

Like

Log In to Comment

richardsr (/s/profile/0054Q00000EwvWRQAZ)
July 3, 2017 at 1:51 PM (/s/feed/0D54Q000080iRkgSAE)

Can you provide a screen shot of the tool to help with the json query extraction? I have tried this in Desktop 2017 and do not see any changes to the JSON query for feature object area with this

"tree browser"?

2 comments 42 views

Like Comment

More comments 1 of 2

cbmgit (/s/profile/0054Q00000Ex2TKQAZ)
2 years ago

https://codebeautify.org/jsonviewer (https://codebeautify.org/jsonviewer) and https://jsonformatter.org/json-parser (https://jsonformatter.org/json-parser) tools helps to view data in tree.

Like

Log In to Comment

Follow

Files (0) (/s/relatedlist/ka14Q000001DWyDQAW/AttachedContentDocuments)

Related Articles

Tutorial: Getting Started with FME Desktop (/s/article/getting-started-with-fme-desktop-translate-data-be)

Tutorial: Getting Started with APIs (/s/article/getting-started-with-apis-1)

Tutorial: Getting Started with XML (/s/article/tutorial-getting-started-with-xml)

Tutorial: Getting Started with FME Server (/s/article/getting-started-with-fme-server)

Tutorial: Getting Started with Geodatabase (/s/article/tutorial-getting-started-with-geodatabase)

Land Acknowledgement —

Safe Software respectfully acknowledges that we live, learn and work on the traditional and unceded territories of the Kwantlen, Katzie, and Semiahmoo First Nations.