Article

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

Tutorial: Getting Started with XML

Aug 3, 2022 • Knowledge

Product Type
FME Desktop

FME Version
2022.0

Overview

The following tutorial series covers basic approaches for reading and writing both XML and GML. For XML we cover both the XML reader and XMLTemplater / Text File writer approach, and the new XSD based XML reader / writer which uses XML application schemas. For GML, we cover the GML reader / writer in application schema mode, which is the most common approach used by both the generic GML reader / writer and by the GML profile readers / writers such as AIXM and INSPIRE. Overall we try to take a problem-solving approach to make it easier to adapt these workflows to your own datasets and XML/GML profiles.

Background

XML is a form of structured text data that conforms to the open w3c standard. XML formats are commonly used on the web because it is text-based, supports complex data structures and multiple languages. GML, or geography markup language, is meant to support the storage and exchange of geospatial features.

XML formats primarily function as an exchange or messaging format. They carry a lot of schema and do not support spatial indexing, so they are not very efficient for access speed or updates.

Challenge

A big challenge with XML data translation to/from GIS is the fundamental differences in data modeling. GIS typically has relational structures, while XML is typically object-oriented. Practically speaking, this means that XML usually has a highly nested tree structure, and contains series/list features, external references, and multiple geometries. So it may take several tables and rows to model a given XML object in a relational system.

Tutorials

If you have time, before you begin, read through the FME Approach to XML (/s/article/tutorial-fme-approach-to-xml) and XML Reader Configuration (/s/article/xml-reader-configuration) articles.

Reading XML with XML and XSD XML Readers

XML is a powerful exchange format that allows for richly structured data models to support data sharing. XML can be used to store large, complex datasets compliant with a specific standard, or can be used to store small messages used in web communications such as HTTP POST bodies.

Most often, XML is read with FME using predefined XML profiles for specific XML formats, or FME's XML reader with Feature Paths. Other approaches to reading include XfMaps, or FME’s Text File reader using XQuery and other XML processing transformers. These approaches are considered advanced and are covered in other articles. In the sections below we will take a look at two styles of XML reading - using the standard XML Reader and using the XSD-Driven XML Reader.

- 1a. Converting from XML (Simple XML Reading Example) (/s/article/tutorial-getting-started-with-xml-reading)
- 1b. How to Read XSD-Driven XML (/s/article/How-to-read-XSD-Driven-XML)

Writing XML with XML and XSD XML Writers

FME has two approaches to writing XML. The first follows building the XML document structure within an XMLTemplater transformer and then writing the result out using the Text File Writer. fme:get-attribute functions are placed within the template at the locations where you want feature attribute values to be merged into the document. This functions much like a mail merge operation, where a letter template receives name and address values from each record to dynamically generate customized messages. Subtemplates can be used to include records from multiple feature types. Finally, the resulting XML document is written out using the Text File Writer.

The second approach consists of writing XML documents conforming to an existing (W3C XSD) XML Schema document. XSD was introduced in 2001, after being acknowledged by the World Wide Web Consortium (W3C) for being a beneficial way of structuring, defining, and restricting elements. Uniquely, in FME, XSD-Driven XML Reading/Writing gives you the advantage of interpreting XML elements as FME features.

The FME feature types must correspond to top-level element declarations, or elements reachable via top-level element declarations, in the XSD Schemas. The top-level element declarations from the XSD Schemas can be identified with the writer’s Feature Paths parameter.

- 2a. XML Writing with XMLTemplater (/s/article/xml-writing-with-xmltemplater)
- 2b. How to Consume and Produce XML using Application Schemas with the XSD-Driven XML Writer (/s/article/XSD-Driven-XML)

Reading / Writing GML

GML is a powerful exchange format that allows for richly structured data models that include both attribute properties and geospatial data. It is also an open standard maintained by the OGC - Open Geospatial Consortium that supports open data exchange and data sharing applications around the world.

The overall objective of these GML exercises is to demonstrate a complete example workflow for reading from GML and loading into a database, and then reading the database and writing back to GML, including full validation of the output against the app schema, visual inspection, and inspection of the XML. We use basemap data from UK OS OpenMap Local (https://www.ordnancesurvey.co.uk/business-government/products/open-map-local) for the Gloucester area.

- 3a. GML Reading (https://community.safe.com/s/article/reading-xmlgml)
- 3b. GML Writing (https://community.safe.com/s/article/gml-writing-with-application-schemas)

Additional Resources

See the following articles for more information:

- Understanding XML: The Human’s Guide to Machine-Readable Data (http://blog.safe.com/2016/07/understanding-xml-humans-guide-machine-readable-data/)

First Published Date

7/29/2020, 12:16 AM

Last Published Date

8/3/2022, 9:50 PM

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 3, 2022 at 9:50 PM (/s/feed/0D54Q00009ggPBiSAM)

Like Comment

Log In to Comment

Follow

Files (0) (/s/relatedlist/ka14Q000001DWxeQAG/AttachedContentDocuments)

Related Articles

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

Tutorial: Getting Started with JSON (/s/article/tutorial-getting-started-with-json)

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

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

Converting from XML (Simple XML Reading Example) (/s/article/tutorial-getting-started-with-xml-reading)

SAFE SOFTWARE®
(https://safe.com)

Getting Started (../s/topic/0TO4Q000000QKioWAG/welcome)
Ideas (../s/bridea/acideasULT brIdea c/00B4000000fWhEVUA1)
Feedback (https://forms.gle/LkoFWpziDYaWQkL78)
Forums (../s/forums/)
Groups (../s/group/CollaborationGroup/00Ba000000A0BxJEAV)
Knowledge Base (../s/knowledge-base/)
Support (../s/support/)

Register / Log In (/s/login/)

© Safe Software Inc | Legal (https://www.safe.com/legal/)

(https:[illegible]/))

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.