State of LocationTech GeoMesa

FOSS4G 2021 Buenos Aires
Jim Hughes
September 30, 2021
LocationTech
GeoMesa
Overview
What is GeoMesa?

A suite of tools for streaming, persisting, managing, and analyzing spatio-temporal data at scale
What is GeoMesa?

A suite of tools for **streaming**, **persisting**, **managing**, and **analyzing** spatio-temporal data at scale
What is GeoMesa?

A suite of tools for streaming, persisting, managing, and analyzing spatio-temporal data at scale
What is GeoMesa?

A suite of tools for streaming, persisting, managing, and analyzing spatio-temporal data at scale
What is GeoMesa?

A suite of tools for streaming, **persisting**, managing, and **analyzing** spatio-temporal data at scale.
Proposed Reference Architecture
GeoMesa 3.0

Released July 2020

Persistence / Databases:

- Accumulo 2.0 support with S3!
Accumulo 2.0 support with S3!

Accumulo 2.0 adds support for AWS S3.

- This allows for much cheaper cloud storage.
- Backend storage can be reduced by 90%+ in some cases.

Using S3 as a data store for Accumulo

Author: Keith Turner
Date: 10 Sep 2019

Accumulo can store its files in S3, however S3 does not support the needs of write ahead logs and the Accumulo metadata table. One way to solve this problem is to store the metadata table and write ahead logs in HDFS and everything else in S3. This post shows how to do that using Accumulo 2.0 and Hadoop 3.2.0. Running on S3 requires a new feature in Accumulo 2.0, that volume choosers are aware of write ahead logs.
GeoMesa 3.1
Released October 2020

GeoMesa NiFi
- What is Apache NiFi?
- GetGeoMesaKafka Record
- GeoAvro RecordSetWriter
What is NiFi?

From https://nifi.apache.org/

Apache NiFi supports powerful and scalable directed graphs of data routing, transformation, and system mediation logic. Some of the high-level capabilities and objectives of Apache NiFi include:

- **Web-based user interface**
  - Seamless experience between design, control, feedback, and monitoring
- **Highly configurable**
  - Flow can be modified at runtime
  - Back pressure
- **Data Provenance**
  - Track dataflow from beginning to end
- **Designed for extension**
  - Build your own processors and more
  - Enables rapid development and effective testing
What is NiFi?

From [https://nifi.apache.org/](https://nifi.apache.org/)

Apache NiFi supports powerful and scalable directed graphs of data routing, transformation, and system mediation logic. Some of the high-level capabilities and objectives of Apache NiFi include:

- **Web-based user interface**
  - Seamless experience between design, control, feedback, and monitoring

- **Highly configurable**
  - Flow can be modified at runtime
  - Back pressure

- **Data Provenance**
  - Track dataflow from beginning to end

- **Designed for extension**
  - Build your own processors and more
  - Enables rapid development and effective testing
How do we use NiFi?

We typically use NiFi for

- Managing data flows
- ETL
  - Extract
  - Transform
  - Load
How do we use NiFi?

We typically use NiFi for

- Managing data flows
- ETL
  - Extract
  - Transform
  - Load

As an example, one could:

Use a GetHTTP or ListenTCP processor to extract data from a source.

A processor like TransformRecord or TransformXML can be used to transform data in flow files.

Processors like PutJDBC, PutTCP, or PutS3 can load data into external systems.
How do we use NiFi?

We typically use NiFi for

- Managing data flows
- ETL
  - Extract
  - Transform
  - Load

As an example, one could:

Use a **GetHTTP** or **ListenTCP** processor to extract data from a source.

A processor like **TransformRecord** or **TransformXML** can be used to transform data in flow files.

Processors like **PutJDBC**, **PutTCP**, or **PutS3** can load data into external systems.

Let’s do this for SimpleFeatures and GeoMesa!
GeoMesa-NiFi

GeoMesa-NiFi is a GeoMesa community project to add NiFi processors and components to help NiFi users integrate GeoMesa into their NiFi flows.
GeoMesa-NiFi

GeoMesa-NiFi is a GeoMesa community project to add NiFi processors and components to help NiFi users integrate GeoMesa into their NiFi flows.
In addition to processors, there are

- **Configuration Services**
  - To manage the configuration for connecting to datastores

- **GeoAvroRecordSetWriter**
  - Allows any processor which write out NiFi RecordSets to create GeoAvro files.
GeoMesa 3.2

Released April 2021

GeoMesa-NiFi

- Support for modifying writes

Spark

- Added Scala 2.12 support
- Added Spark 3.0/3.1.1 support

Streaming

- Transactional Writes
- Metrics Integration
- Readiness Check (GM-GS)
NiFi: Support for modifying writes

UpdateGeoMesa*Record Processors are new in 3.2.0!
Spark: Added Spark 3.0/3.1.1 support

Required adding Scala 2.12 version of the build.

<table>
<thead>
<tr>
<th>GeoMesa Version</th>
<th>Supported Scala Versions</th>
<th>Supported Spark Versions</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.x</td>
<td>2.11</td>
<td>Up to 2.4.x</td>
</tr>
<tr>
<td>3.0 - 3.1.x</td>
<td>2.11</td>
<td>Up to 2.4.x</td>
</tr>
<tr>
<td>3.2+</td>
<td>2.11 support deprecated</td>
<td>2.3/2.4 support deprecated</td>
</tr>
<tr>
<td></td>
<td>2.12 support added</td>
<td>3.0 / 3.1+ support added</td>
</tr>
<tr>
<td>4.0</td>
<td>2.12</td>
<td>3.0+</td>
</tr>
</tbody>
</table>
Kafka: Transactional Writes


From https://www.confluent.io/blog/transactions-apache-kafka/

18.9. Transactional Writes

Kafka supports the concept of transactional writes. GeoMesa exposes this functionality through the GeoTools transaction API:

```java
import org.geotools.data.DataStore;
import org.geotools.data.DataStoreFinder;
import org.geotools.data.DataStoreTransaction;
import org.geotools.data.Transaction;

DataStore store = DataStoreFinder.getDataStore(params);
// the transaction will contain the Kafka producer, so make sure to close it when finished
try (Transaction transaction = new DataStoreTransaction(store);)
// pass the transaction when getting a FeatureWriter
    try (FeatureWriter<SimpleFeatureType, SimpleFeature> writer =
        store.getFeatureWriter(Append, "my-type", transaction)) {
        // write some features (aligned), then commit the transaction:
        transaction.commit();
        // if you get an error (aligned), then rollback the transaction:
        transaction.rollback();
    }
// re-using the transaction will re-use the Kafka producer
try (FeatureWriter<SimpleFeatureType, SimpleFeature> writer =
    store.getFeatureWriter(Append, "my-type", transaction)) {
    // write some features (aligned), then commit the transaction:
    transaction.commit();
}
```

![Diagram showing the interaction between Kafka and GeoMesa for transactional writes](https://www.geomesa.org/documentation/stable/user/kafka/transactional_writes.html)
GeoMesa-GeoServer is a project to hold code that helps integrate extra GeoMesa capabilities into GeoServer.

GeoMesa implements GeoTools interfaces, so GeoMesa-GeoServer is not needed for all deployments.
The `geomesa-gs-kafka-catalog-listener` does two things:

- Starts reading from Kafka
- Provides an endpoint to show when all topics are loaded

This allows for GeoServer in a container environment (e.g. Kubernetes) to wait until it has all the data before being “ready”
GeoMesa 3.3
Released September 2021

GeoMesa-NiFi

- PostGIS Write Support

Streaming

- Kafka Layer Views

Bulk Ingest into Accumulo

Improvements in FSDS Metadata
Kafka Layer Views

Expose your data in different ways to different users, without duplicating the storage and memory required.

```json
{
  type1 = [
    { type-name = "transformView", transform = [ "dtg", "geom", "name" ] },
    { type-name = "filterView", filter = "bbox(geom,0,0,10,10)" }
  ],
  type2 = [
    { type-name = "filterTransformView", filter = "bbox(geom,0,0,10,10)", transform = [ "dtg", "geom" ] }
  ]
}
```
Database tables in Accumulo and HBase consist of large, immutable files. During normal operations, these are written by *compactions*. When loading a large volume of data, compactions can slow down ingest.

GeoMesa 3.3 adds support for creating Accumulo RFiles via MapReduce jobs. Previous versions of GeoMesa had such support for HBase. This allows for building large tables more quickly (than using a traditional ingest).
NiFi: PostGIS Write Support

Data store connections have been moved out to NiFi controller services, simplifying NiFi configurations and allowing for re-use.

Postgis is now a supported data store controller service!

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Bundle</th>
</tr>
</thead>
<tbody>
<tr>
<td>GeoAvroRecordSetWriterFactory</td>
<td>GeoAvroRecordSetWriterFactory 3.3.0</td>
<td>org.geomesa.nifi - geomesa-datastore-service</td>
</tr>
<tr>
<td>KafkaDataStoreService</td>
<td>KafkaDataStoreService 3.3.0</td>
<td>org.geomesa.nifi - geomesa-kafka-nar</td>
</tr>
<tr>
<td>PostgisDataStoreService</td>
<td>PostgisDataStoreService 3.3.0</td>
<td>org.geomesa.nifi - geomesa-datastore-service</td>
</tr>
</tbody>
</table>
GeoMesa 3.4 / GeoMesa 4.0

TBD

Spark

- Integration with Apache Sedona

Deprecations

Java 11 Support
WIP: Spark Integration with Apache Sedona

From https://sedona.apache.org/

Apache Sedona (incubating) is a cluster computing system for processing large-scale spatial data. Sedona extends Apache Spark / SparkSQL with a set of out-of-the-box Spatial Resilient Distributed Datasets / SpatialSQL that efficiently load, process, and analyze large-scale spatial data across machines.
WIP: Spark Integration with Apache Sedona

Apache Sedona implements capabilities which optimize spatial joins in Spark.

There is a PR which provides a basic integration here: https://github.com/locationtech/geomesa/pull/2802
Deprecations

The following modules have been deprecated and are slated for removal in 4.0:

- GeoMesa Kudu
- GeoMesa Streaming (Camel integration)
- GeoMesa Web
- GeoMesa GeoJSON
Java 11 Support

In order to support Java 11, GeoMesa will need to:

- Fix a few split packages
  - The team is waiting for the next major release to make this breaking change
- Test each of the backends and update accordingly
- Add support for cross building in CI/CD
Thanks!

- jhughes@ccri.com
- https://www.geomesa.org/
- https://gitter.im/locationtech/geomesa
- https://github.com/locationtech/geomesa
- Twitter @CCR_inc

CCRi is hiring!

https://www.ccri.com/careers/

- DevOps
- Software Engineers
- Data Scientists
Questions?

https://www.geomesa.org
https://gitter.im/locationtech/geomesa
jhughes@ccri.com

Other Talks:

Introduction to Big Data Storage with LocationTech GeoMesa
2021-09-30, 15:30–16:00, Ushuaia

Streaming IoT sensor data with LocationTech GeoMesa, Apache Kafka, and NiFi
2021-10-01, 09:00–09:30, Bariloche