Looking at the New Features of Apache NiFi

Timothy Spann
Principal Developer Advocate

Sunday October 8, 2023
4:10PM - 4:50 PM
Room 102
FLaNK Stack

Tim Spann
@PaasDev // Blog: www.datainmotion.dev
Principal Developer Advocate.
Princeton Future of Data Meetup.
ex-Pivotal, ex-Hortonworks, ex-StreamNative, ex-PwC
https://medium.com/@tspann
https://github.com/tspannhw
Apache NiFi x Apache Kafka x Apache Flink
Future of Data - New York + Princeton + Virtual

https://www.meetup.com/futureofdata-princeton/
https://www.meetup.com/futureofdata-newyork/

From Big Data to AI to Streaming to Containers to Cloud to Analytics to Cloud Storage to Fast Data to Machine Learning to Microservices to ...
This week in Apache NiFi, Apache Flink, Apache Kafka, Apache Spark, Apache Iceberg, Python, Java, AI, ML, LLM and Open Source friends.
My Talk List

Utilizing Real-Time Transit Data for Travel Optimization

Let’s Monitor the Conditions at the Conference
Apache NiFi has a lot of new features, processors and best practices that have arrived in the last year or so.

I will walk through building flows using the latest tips, techniques and processor.

I will and change a number of data flows utilizing the latest NiFi version and point out gotchas and some never dos. The deck will act as a take-away with notes, tips and guides to what we covered.

====> Any NiFi 1.23+ and 2.0 in progress features people want to see?
Records

New ExcelRecord Reader

AmazonGlueSchemaRegistry

https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12353320
# New to 2023 Processors

<table>
<thead>
<tr>
<th>GenerateRecord</th>
<th>ListGoogleDrive</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetAsanaObject</td>
<td>FetchGoogleDrive</td>
</tr>
<tr>
<td>PutSalesforceObject</td>
<td>PutGoogleDrive</td>
</tr>
<tr>
<td>QuerySalesforceObject</td>
<td>PutBoxFile</td>
</tr>
<tr>
<td>PutIoTDBRecord</td>
<td>DecryptContent</td>
</tr>
<tr>
<td>QueryIoTDBRecord</td>
<td>DecryptContentCompatibility</td>
</tr>
</tbody>
</table>

New to 2023 Processors

ExtractRecordSchema

RemoveRecordField

VerifyContentMAC

TriggerHiveMetaStoreEvent

“count” function added to RecordPath
AWS ML Service Processors

https://github.com/tspannhw/FLaNK-AWSML
AWS Translate
Deprecating for Removal

Deprecate Lua and Ruby Script Engines

Deprecate ECMAScript Script Engine

Deprecate the Ambari Reporting Task

Deprecate Kafka 1.x components and 2.0 components

XML Templates

Variables

See: https://cwiki.apache.org/confluence/display/NIFI/Deprecated+Components+and+Features
Start Using

ExecuteStateless -> run your stateless flows right in a regular NiFi cluster

Parameters

JSON Flow Serialization

Records everywhere
NiFi 2.0 Coming

- Python Integration
- Parameters
- JDK 17, maybe JDK 21+
- JSON Flow Serialization
- Rules Engine for Development Assistance
- Run Process Group as Stateless
- flow.json.gz

https://cwiki.apache.org/confluence/display/NIFI/NiFi+2.0+Release+Goals
https://medium.com/cloudera-inc/getting-ready-for-apache-nifi-2-0-5a5e6a67f450
Thanks to Pierre!

Pierre Villard
Apache NiFi Committer & PMC member | Working @Cloudera - ex-@Google | Twitter & Github — @pvillard31 | Blog @ www.pierrevillard.com
Python as First Class (NIFI-11241)

Graphical UI with custom Python based extensions

```python
import os
import sys
import json
from nifi.properties import PropertiesDescriptor
from nifi.properties import ResourceDefinition
from ni4java.filesystem import File

def _init_(self, name, namespace):
    self.name = name
    self.namespace = namespace

def GetPropertiesDescriptors(self):
    self.property_descriptors = []
    for descriptor in self.property_descriptors:
        if descriptor.name == 'python_class_file':
            self.property_descriptors.remove(descriptor)
    self.property_descriptors.append(PropertiesDescriptor(
        name='python_class_file',
        description='The path to the class file containing the classes that may be detected by the model. Expected format is one class name per line, newline terminated.',
        required=False,
        resource_definition=ResourceDefinition(allow_file=True)
    ))

def getName(self, file):
    return os.path.splitext(os.path.basename(file))[0]

def getClassNamesFromFile(self, file):
    class_names = self.property_descriptors['python_class_file'].getValue()
    return class_names
```

NEW in NiFi 2.0
Apache NiFi in a few numbers

A very active project with a dynamic community & comparison with ACEU 2019

- 2800+ members on the Slack channel (535+ - 4 years ago)
- 475+ contributors on Github across the repositories (260+ - 4 years ago)
- 65 committers in the Apache NiFi community (45 - 4 years ago)
- Apache NiFi 1.23.2 is the latest release, NiFi 2.0 coming soon (NiFi 1.10 - 4 years ago)
- 14M+ docker pulls of the Apache NiFi image (1M+ - 4 years ago)
NiFi Deploy Options from Open Source to Managed

- **MiNiFi C++**
  (small footprint)

- **MiNiFi Java**
  (headless version of NiFi)

- **NiFi Registry**

- **Stateless NiFi**

- **Cloudera Edge Flow Manager**
  (Command & Control of MiNiFi Agents)

- **Cloudera NiFi for Kafka Connect**

- **NiFi in Cloudera DataFlow Functions**

- **Cloudera DataFlow**
NiFi 2.0 is coming... https://medium.com/cloudera-inc/getting-ready-for-apache-nifi-2-0-5a5e6a67f450

- First-class citizen Python API
- Rules Engine
- NiFi Stateless at Process Group level
- Java 21 (virtual threads, perf improvements, etc)

Closing the gap between data engineers and data scientists...
- Export documentation (Sharepoint, OCR) to build the knowledge base powering your chatbot
- Scrape the internet (Sitemap) to build the knowledge base powering your chatbot
- Real-time streaming ingest of Slack to build the knowledge base powering your chatbot

THANK YOU