Cloud Computing – Lecture 12

Data streams, data flow pipeline management

28 April 2020

Chinmaya Dehury

Satish Srirama
Outlines

- Data streaming
- Data pipeline
- Amazon data pipeline
- Apache Nifi
Data streaming

• Continuous flow of data
• Usually thousands of data sources are involved
• Generated data are of small size
• Higher frequency of data generation
Stream data processing use cases

• **Anomaly Detection**
  o Detect problems in real time (cyber intrusions, financial fraud,
  o Continuously collect and analyse network traffic, transactions, 
    user behaviour

• **Predictive maintenance**
  o Collect and process performance data from deployed devices
  o Forecast potential faults and service disruptions, predict 
    maintenance cycles

• **Clickstream analytics**
  o Collect and analyse user clicks, routes and behaviour
  o Extract frequent patterns to improve user engagement
  o Personalized recommendations
Stream data processing frameworks

• Frameworks/extensions specifically designed for:
  o Low latency data processing
  o Dynamically process incoming data streams
  o Aggregate data processed at different time periods
  o Push results to external systems as output streams

• Two main approaches/models:
  1. Micro Batch processing
  2. Real Time stream data processing
Stream processing models

- **Micro Batch processing**
  - Collect incoming data into a batch/buffer
  - Processing one batch at a time
  - High throughput, High latency
  - Spark Streaming

- **Real time processing**
  - Process each incoming message right away
  - Low latency, lower throughput
  - Apache Storm, Apache Flink
Then What is Data Pipeline?
Data Pipeline

Pipeline approach for computer instruction execution:

![Pipeline diagram for computer instruction execution](https://slideplayer.com/slide/8207220/)

Pipeline approach in manufacturing:

![Pipeline diagram for manufacturing](http://www.ni.com/cms/images/devzone/tut/final.JPG)
Pipeline approach in logistic:

<table>
<thead>
<tr>
<th>International Shipping Company</th>
<th>Tracking Number</th>
<th>Remarks</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>菜鸟超级经济Global</td>
<td>S00000090969004</td>
<td>2019.11.26 19:37 (GMT-7): Departed country of origin</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>2019.11.26 14:37 (GMT-7): Shipment accepted by airline</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>2019.11.26 14:37 (GMT-7): Shipment left country of origin warehouse</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>2019.11.26 04:01 (GMT-7): Shipment at country of origin warehouse</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>2019.11.26 03:49 (GMT-7): Shipment dispatched</td>
<td></td>
</tr>
</tbody>
</table>

Tracking information is available within 5-10 days. You can track your order here 菜鸟超级经济Global.
Pipeline approach for handling the flow and processing of data.
Data Pipeline (DP)

Data in -> Data processing -> Data out

Data in -> Large data processing task -> Data out
• A system for moving data from one system to another.
• Encompasses ETL as a subsystem
• Transformation of data is optional
• May be processed in real-time or in batch manner
Data Pipeline properties

1. Low Event Latency
2. Scalability
3. Interactive Querying
4. Versioning
5. Monitoring
6. Testing
Types of data pipeline solutions

1. Batch
2. Real-time
3. Cloud native
4. Open source
Data Pipeline Technologies

1. Amazon Data pipeline
2. Apache Nifi
Data Pipeline Technologies

1. Amazon Data pipeline

2. Apache Nifi
Amazon Data Pipeline

• A web service for reliable process and movement of data
• Focus is on AWS compute and storage services
• AWS services such as Amazon S3, Amazon RDS, Amazon DynamoDB, and Amazon EMR
• Data processing workloads can be
  • fault tolerant
  • repeatable
  • highly available
Amazon Data Pipeline

1. Major components
   I. DataNodes
   II. Activities

2. Additional components
   I. Schedules
   II. Preconditions
   III. Resources

Cloud Computing - Lecture 12: Data Streams, Data Flow Pipeline Management
Amazon Data Pipeline

1. Major components

I. **DataNodes**: It specifies the name, location, and format of the data sources such as Amazon S3, Dynamo DB, etc.
   i. DynamoDBDataNode
   ii. SqlDataNode
   iii. RedshiftDataNode
   iv. S3DataNode

II. **Activities**: Activities are the actions that perform the SQL Queries on the databases, transforms the data from one data source to another data source.
Amazon Data Pipeline

1. Major components

   I. DataNodes

   II. Activities

      i. CopyActivity
      ii. EmrActivity
      iii. HiveActivity
      iv. HiveCopyActivity
      v. PigActivity
      vi. RedshiftCopyActivity
      vii. ShellCommandActivity
      viii. SqlActivity
Amazon Data Pipeline

1. Major components
   I. DataNodes
   II. Activities

2. Additional components
   I. Schedules: Schedule defines the timing of a scheduled event, such as when an activity runs.
2. Additional components

I. Schedules

II. Preconditions: A condition that must be true before an activity can run. E.g., check if the data is present on the source before attempting to run CopyActivity.

A. System-managed Precondition:
   a) DynamoDBDataExists
   b) DynamoDBTableExists
   c) S3KeyExists, etc..

B. User-managed precondition
   a) Exists: Checks whether a data node exists.
   b) ShellCommandPrecondition: Unix/Linux shell command that can be run as a precondition
2. Additional components

I. Schedules

II. Preconditions

III. Resources: refer to the computational resource that performs the work that a pipeline activity specified

I. **Ec2Resource**: An EC2 instance

II. **EmrCluster**: An Amazon EMR cluster
Amazon Data Pipeline: An Example

Configuration -> Schedule
Schedule: Every 1 Day

S3Datanoode IntermediateDocDir -> CopyActivity CopyActivity

S3Datanoode DataDirectory -> EMRCluster DefaultEMRCluster1

PigActivity PigProcessDocuments

S3Datanoode OutputDirectory
Amazon Data Pipeline: An Example

1. Pipeline Configuration
2. Configuration Default
3. Execution Schedule
4. Schedule Every 1 Day
5. Data Node
6. S3Datnode IntermediateDocDir
7. CopyActivity
8. CopyActivity
9. S3Datnode DataDirectory
10. EMRCluster Default EMRCluster1
11. AWS Resource
12. Pipeline Activity
13. Data Node
14. PigActivity
15. PigProcessDocuments
16. S3Datnode OutputDirectory
1. Amazon Data pipeline

2. Apache Nifi
Apache Nifi Data Pipeline

• Open-source, under the Apache Software Foundation
• Automates and manages the flow of data between systems
• Web-based User Interface for creating, monitoring, & controlling data flows.
• Clients [src]:
  • Micron: Semiconductor Manufacturing
  • Payoff: Financial Wellness (fintech)
  • Slovak: Telekom Telecommunications
  • Looker: SaaS & Analytics Software
  • Hastings Group: Insurance
  • and many more....
• Latest version 1.11.4
Apache Nifi Data Pipeline

Key Features

Flow Management:
• Data Buffering
• Prioritized Queuing
• Guaranteed Delivery

Ease of Use:
• Flow Templates
• Data Provenance
• Fine-grained history
Apache Nifi Data Pipeline

Key Features

Security

• System to System
• User to System
• Multi-tenant Authorization

Extensible Architecture

• Extension
• Site-to-Site Communication Protocol
Apache Nifi Data Pipeline

1. Major components
   I. Processors
   II. Queue (between processors)

2. Additional components
   I. Input Port
   II. Output Port
   III. Process Group
   IV. Remote Process Group
   V. Template
Apache Nifi Data Pipeline

Key concepts

1. Process Group
2. Flow
3. Processor
4. Flowfile
5. Event
6. Data provenance
Apache Nifi Data Pipeline

1. Major components
   I. Processors

293 processors
1. Major components
   
   I. Processors

   Different States of a Processor:
   
   *Start, Stop, Enable, & Disable*

   Disable processor can not be started.

   When a group of Processors is started, this (disabled) Processor should be excluded.
Apache Nifi Data Pipeline

1. Major components

   I. Processors

Configuring a Processor

**SETTING:**

**Penalty duration:** Time to wait, when the data can not be processed for some reason.

**Yield Duration:** Time to wait, when the process can not progress.

**Bulletin level:** Level of bulletin, Nifi will display in the user interface.
Apache Nifi Data Pipeline

1. Major components

   I. Processors

Configuring a Processor

Scheduling:

Time vs Event vs CRON Driven

Concurrent Tasks: Number of FlowFiles should be processed by this Processor at the same time.
1. Major components

I. Processors

Configuring a Processor

Properties:

• Provides a mechanism to configure Processor-specific behavior.
• There are no default properties.
Different categories of processors

• **Data Ingestion Processors:** GetFile, GetHTTP, GetFTP, etc

• **Routing and Mediation Processors:** RouteOnAttribute, RouteOnContent, ControlRate, RouteText, etc.

• **Database Access Processors:** ExecuteSQL, PutSQL, PutDatabaseRecord, ListDatabaseTables, etc.

• **Attribute Extraction Processors:** UpdateAttribute, EvaluateJSONPath, ExtractText, AttributesToJSON, etc

• **System Interaction Processors:** ExecuteScript, ExecuteProcess, ExecuteGroovyScript, ExecuteStreamCommand, etc
Different categories of processors

• **Data Transformation Processors**: ReplaceText, JoltTransformJSON, etc.

• **Sending Data Processors**: PutEmail, PutSFTP, PutFile, PutFTP, etc.

• **Splitting and Aggregation Processors**: SplitText, SplitJson, SplitXml, MergeContent, SplitContent, etc.

• **HTTP Processors**: InvokeHTTP, ListenHTTP, etc.

• **AWS Processors**: GetSQS, PutSNS, PutS3Object, FetchS3Object, etc.
Apache Nifi Data Pipeline

1. Major components

II. Queue
Apache Nifi Data Pipeline

1. Major components

II. Queue

• To handle the large amount of data inflow.
• Possible to see the content, ID, Filename, FileSize etc of a flowfile
Apache Nifi Data Pipeline

Templates:
• Can be thought of as a reusable sub-flow.
• Any properties that are identified as being Sensitive Properties (such as a password that is configured in a Processor) will not be added to the template.
Apache Nifi Data Pipeline

Templates:

Upload Template

Add Template
Research on Data Pipeline

• We in RADON, focusing on developing the data pipeline platform for data intensive applications.
• For serverless applications
• TOSCA model for data pipeline
• Atop Apache NiFi, Amazon data pipeline.
What next ???
Let’s move to lab session...
References

Thank you