The Big Data Puzzle -
Where Does the Eclipse Piece Fit?
About me:

J. Langley -


Developing applications and frameworks based on Eclipse technology since 2009.

Specialize in “Horizontal Integration” - combining existing technology to provide new capabilities.

Find me online:

Email – jlangley@cohesionforce.com
Twitter – @jperiodlangley
About CohesionForce:

Since its founding in 1998, CohesionForce has been providing innovative solutions to our customers’ most complex modeling & simulation, systems engineering, and software engineering challenges.

Core Values:

• Disciplined engineering should be easy
• Systems should be fulfilling to use
• Better tools are needed to solve the problems of tomorrow
The tools used for Big Data analytics seem to be converging on the Apache Software Foundation as a home. As an Eclipse and Apache integrator, CohesionForce has found a useful fit for Eclipse projects when used as tooling for an underlying Apache project runtime.

Using file formats such as Apache Avro and Parquet along with a compute system such as Apache Hive or Spark allows us to query the data using a SQL “like” language with Apache Hive and Spark.
Scope:

Big Data is well... a big topic. In this talk, we will focus on the following:

1. Sample data used to approximate the problem space
2. Apache Projects that CohesionForce has used for data analysis
3. Example Configurations tested
4. Eclipse Projects that CohesionForce has used/developed for tooling
5. Thoughts on future work with Eclipse for Big Data & Data Science
Sample Data:

A sizable list of data available for use can be found here: https://aws.amazon.com/public-data-sets/

These data sets are a mix of textual, spatial, image, and video. They closely approximate the size of the data that we needed, but did not have the combination of factors that we were looking for.

We needed data for events that have an ID, type, location, and a time. We were able to create a suitable data set by transforming taxi data available here: http://www.andresmh.com/nyctaxitrips/

After converting the data to a common format (DIS EntityStatePDU), the result was just over 174M events, but can be doubled if necessary.
Distributed Interactive Simulation:

Distributed Interactive Simulation (DIS)

DIS is an IEEE standard (IEEE-1278.1) developed by the Simulation Interoperability Standards (SISO) and approved by IEEE.

The EntityStatePDU contains fields such as
- EntityID – Site, Application, Entity
- EntityType – 7-tuple enumeration
- EntityLocation – x/y/z Geocentric WGS-84
- EntityOrientation – psi/theta/phi Euler angles
- EntityVelocity – x/y/z along the orientation axis

The EntityStatePDU is used as an example throughout this presentation.
**Apache Big Data Projects:**

<table>
<thead>
<tr>
<th>Apache Ambari</th>
<th>Apache Hama</th>
<th>Apache Tajo</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache Apex</td>
<td>Apache Helix</td>
<td>Apache Tez</td>
</tr>
<tr>
<td>Apache Avro</td>
<td>Apache Ignite</td>
<td>Apache VXQuery</td>
</tr>
<tr>
<td>Apache Beam</td>
<td>Apache Kafka</td>
<td><strong>Apache Zeppelin</strong></td>
</tr>
<tr>
<td>Apache Bigtop</td>
<td>Apache Knox</td>
<td></td>
</tr>
<tr>
<td>Apache BookKeeper</td>
<td>Apache Lens</td>
<td></td>
</tr>
<tr>
<td>Apache CouchDB</td>
<td>Apache MetaModel</td>
<td></td>
</tr>
<tr>
<td>Apache Crunch</td>
<td>Apache ORC</td>
<td></td>
</tr>
<tr>
<td>Apache DataFu</td>
<td>Apache Oozie</td>
<td></td>
</tr>
<tr>
<td>Apache DirectMemory</td>
<td>Apache Parquet</td>
<td></td>
</tr>
<tr>
<td>Apache Drill</td>
<td>Apache Phoenix</td>
<td></td>
</tr>
<tr>
<td><strong>Apache Edgent</strong></td>
<td>Apache REEF</td>
<td></td>
</tr>
<tr>
<td>Apache Falcon</td>
<td>Apache Samza</td>
<td></td>
</tr>
<tr>
<td>Apache Flink</td>
<td>Apache Spark</td>
<td></td>
</tr>
<tr>
<td>Apache Flume</td>
<td>Apache Sqoop</td>
<td></td>
</tr>
<tr>
<td>Apache Giraph</td>
<td>Apache Storm</td>
<td></td>
</tr>
</tbody>
</table>

**Interesting Note:**

Apache lists projects Cassandra, Hadoop, Hive, Pig, and ZooKeeper under the database category.
CohesionForce experience with:

- Apache Parquet
- Apache ZooKeeper
- Spark
- Kafka
- Zeppelin
- Avro
- Hive
- Apache Parquet
- Apache ZooKeeper

Innovative Solutions For Mission Critical Systems
Data Formats

Apache Avro

Apache Parquet
Apache Avro Data Format:

- Schema based with multiple language bindings (C, Python, Java, etc)
- Schema defined using JSON or IDL
- Useful for nested data structures
- File meta data can be added
- Schema stored with each serialized file
- Allows for dynamic typing or “type discovery”
- Multiple options for compression – currently using deflate (zlib) and snappy
  - Note: Snappy compression will be available for the C bindings with version 1.9.0

Example Schema for EntityType
Apache Avro Data Format:

```json
{"type": "record", "name": "EntityType", "namespace": "com.cohesionforce.dis.avro", "fields": ["entityKind", "domain", "country", "category", "subcategory", "spec", "extra"]}
```
Apache Parquet Data Format:

- Columnar storage format
- Also schema based
- Type specific compression
- Initially developed by Twitter and open sourced
- Promoted to an Apache Top-Level project on April 27, 2015
- Twitter is converting their data from Avro to Parquet

Traditional:
1. protoVersion, exID, type, family, time, length, pad
2. protoVersion, exID, type, family, time, length, pad
3. protoVersion, exID, type, family, time, length, pad
4. protoVersion, exID, type, family, time, length, pad
5. protoVersion, exID, type, family, time, length, pad
6. protoVersion, exID, type, family, time, length, pad

Parquet:
- protoVersion:1 – protoVersion:N
- exID:1 - exID:N
- type:1 - type:N
- family:1 - family:N
- time:1 - time:N
- length:1 - length:N
- pad:1 - pad:N
Data Storage
(Also Map/Reduce & Scheduling)

- Replicated storage using HDFS and commodity hardware
- YARN – Job scheduler application for managing Hadoop jobs
- Map/Reduce – programming paradigm for writing jobs that process large amounts of data in parallel
Execution Engines

Apache Hive
Apache Hive converts Hive SQL Statements into commands that run on another execution engine (Map/Reduce, Spark, Tez)
Provides an ODBC Connection

Apache Spark
Takes advantage of multiple cores and large amounts of memory to run massively multi-threaded jobs.
Has incorporated most of Hive capability into the SqlContext.
Apache Storm is a free and open source distributed realtime computation system. Storm makes it easy to reliably process unbounded streams of data.

A Storm topology consumes streams of data and processes those streams in arbitrarily complex ways, repartitioning the streams between each stage of the computation however needed.
Visualization Engines

Zeppelin

Eclipse BIRT

Innovative Solutions For Mission Critical Systems
Apache Zeppelin:

- Web based “notebook” that provides interactive sessions for data analysis
- Currently in incubation phase (We are running 0.7.1 at CFI)
Eclipse BIRT:

- Mature reporting capability for web and application use
- Exports to most supported office formats
- Used more for pre-determined reports than interactive analysis
Example Configurations
Initial Implementation:

- Eclipse BIRT
  - Apache Hive
    - Apache Yarn
      - Hadoop NN
        - Hadoop DN
        - Hadoop DN
        - Hadoop DN
Second Implementation:

- Added Spark Master
- Added Spark Workers
- Spark Worker
  - Hadoop DN
- Spark Worker
  - Hadoop DN
- Spark Worker
  - Hadoop DN
- Apache Hive
- Spark Master
- Apache Yarn
- Hadoop NN
- Eclipse BIRT
Innovative Solutions For Mission Critical Systems

Third Implementation:

- Removed Yarn
- Apache Yarn
  - Eclipse BIRT
  - Apache Hive
  - Spark Master
  - Hadoop NN
    - Spark Worker
      - Hadoop DN
    - Spark Worker
      - Hadoop DN
    - Spark Worker
      - Hadoop DN
Fourth Implementation:

- Removed Hive
- Switched from BIRT to Spark Shell
- Spark Shell
  - Spark Master
    - Hadoop NN
      - Spark Worker
      - Hadoop DN
      - Spark Worker
      - Hadoop DN
      - Spark Worker
      - Hadoop DN
Fifth Implementation: (Presented @EclipseCon/NA 2016)

Browser

Apache Zeppelin
Spark Master
Hadoop NN

Spark Worker
Hadoop DN
Spark Worker
Hadoop DN
Spark Worker
Hadoop DN

Introduced Zeppelin
Current Implementation:

- Browser
  - Apache Zeppelin
    - Spark Master
      - Hadoop NN
        - Spark Worker
          - Spark Worker
          - Spark Worker
  - Split Hadoop And Spark
  - Hadoop DN
  - Hadoop DN
  - Hadoop DN

Innovative Solutions For Mission Critical Systems
Don't forget to show Zeppelin!

Sorry! We did not have time to cover this during the session.

For a good description of Apache Zeppelin, please check out:

https://zeppelin.apache.org/docs/0.7.2/
Experimenting:

- Kafka Producer
- Kafka Producer
- Kafka Producer
- Zookeeper
- Kafka Broker
- Spark Master
- Spark Worker
- Spark Worker
- Spark Worker

Innovative Solutions For Mission Critical Systems
More Spark Info
Spark Resilient Distributed Datasets (RDD):

- The core concept in the Spark framework.

- RDDs are immutable. This allows multiple processes to evaluate the same piece of data – you can't change the data in an RDD, but you can create new RDDs.

- RDDs support two types of operations:
  - Transformations – map, groupByKey, etc. These return a new RDD based on the operation run.
  - Actions – these return a value – count, first, countByKey, etc.
Spark Core Libraries:

- Spark Streaming – batches RDD into a pipeline. Existing operations on RDD can be reused in a stream configuration.

- Spark SQL – exposes Spark datasets to allow SQL like queries.

- Spark MLib – machine learning library. Classification, regression, clustering, collaborative filtering, etc.

- Spark GraphX – API for graphs and graph-parallel computation.
Spark Shell:

- Provides an interactive session for evaluating and debugging transforms.
- Can attach to a separate cluster (based on the master URL) or include its own master in the JVM.
- Available in both Scala and Python.
Eclipse + Apache Workflow:

1. Log Data
2. Load into Hadoop
3. Execute Hive Queries
4. Generate Logger
5. Generate Queries
6. Export Results

Data Model

Innovative Solutions For Mission Critical Systems
Innovative Solutions For Mission Critical Systems
Eclipse Tooling:

**Eclipse-Avro:** The purpose of this repository is to provide the capability to store EMF data files in the Apache Avro format. The Acceleo project is used to generate an Avro Schema based on a given EMF schema along with an AvroResourceImpl that can be used in the place of the XMIResourceImpl to load and save data using the common EMF methodology. https://github.com/LangleyStudios/eclipse-avro

**DIS Toolkit:** The DIS Toolkit provides an EMF model based on the DIS schema provided by the OpenDIS codebase, along with a generated data logger that stores files in compressed binary using the Apache Avro format. https://github.com/CohesionForce/dis-toolkit

**AvroToParquet:** a simple command line converter for Apache Avro to Apache Parquet file formats. https://github.com/CohesionForce/avroToParquet
Other Eclipse Based Projects:

**Eclipse BIRT** - Visualizing Big Data with Hadoop and BIRT

**Talend Open Data Studio** – Start working with Hadoop and NoSQL databases today using simple, graphical tools and wizards to generate native code that leverages the full power of Hadoop.
https://www.talend.com/download/talend-open-studio

**DataStax DevCenter** - a free, Eclipse-based tool, which is designed to be a lightweight visual interface that provides developers and others with the ability to easily run CQL queries against Cassandra, view query results, and perform other related tasks.
http://www.datastax.com/what-we-offer/products-services/devcenter
More Eclipse Based Projects:

**Architect** - provides an Eclipse-based workbench in which data scientists can get their job done, in other words, an integrated development environment (IDE) targeted specifically at data scientists.  

**StatED** - an Eclipse based IDE (integrated development environment) for R. It offers a set of mature tools for R coding and package building. This includes a fully integrated R Console, Object Browser and R Help System, whereas multiple local and remote installations of R are supported.  
[http://www.walware.de/goto/statet](http://www.walware.de/goto/statet)  
[https://github.com/walware/statet](https://github.com/walware/statet)
Interested?

We would like to compile a working list of ideas on this subject.

We would also like to identify potential users of these types of tools to be sure that we implement the proper feature set.

If you need help with integrating any of the tools/concepts that have been covered, please:

Contact us:

CohesionForce
www.cohesionforce.com
Email: jlangley@CohesionForce.com
Twitter: @jperiodlangley
Thoughts on Future Direction:

Eclipse Sirius could be used to create graphical editors or visualizations of component deployment, schemas, queries, etc.

XText could be used to provide editors for Hive Query Language, while also providing binding to types retrieved from data schemas. This would allow a user to write queries with syntax highlighting, code completion, and validation before execution.

XTend could be used to generate loggers, transforms, and other tasks based on a data model. The config files needed for Hadoop & Spark could be generated based on a modeled laydown.

Provisioning bundles, starting containers.
Interested?

We would like to compile a working list of ideas on this subject.

We would also like to identify potential users of these types of tools to be sure that we implement the proper feature set.

If you need help with integrating any of the tools/concepts that have been covered, please:

Contact us:

CohesionForce
www.cohesionforce.com
Email: jlangley@CohesionForce.com
Twitter: @jperiodlangley
Evaluate the Sessions

Sign in and vote at eclipsecon.org

-1  0  +1

Innovative Solutions For Mission Critical Systems