Big Data and Analytics: A Conceptual Overview

Mike Park
Erik Hoel
In this technical workshop

• This presentation is for anyone that uses ArcGIS and is interested in analyzing large amounts of data

• We will cover:
  - Big Data overview
  - The Hadoop platform
  - How Esri’s *GIS Tools for Hadoop* enables developers to process spatial data on Hadoop
  - Looking ahead
Within ArcGIS, Geoprocessing was enhanced at 10.1 SP1 to support 64-bit address spaces
- This is sufficient to handle traditional large GIS datasets

However, this solution may run into problems when confronted with datasets of a size that are colloquially referred to as Big Data
- Internet scale datasets
Data is now central to our existence – both for corporations and individuals

Nimble, thin, data-centric apps exploiting massive data sets generated by both enterprises and consumers

Hardware era: 20 – 30 years
Software era: 20 – 30 years
Data era: ?
Big data
What is it?

• Big Data is a loosely defined term used to describe data sets so large and complex that they become awkward to work with using standard software in a tolerable elapsed time
  - Big data "size" is a constantly moving target, ranging from a few dozen terabytes to many petabytes of data
  - In the past three years, 90% of all recorded data has been generated
• Every 60 seconds:
  - 100,000 tweets
  - 2.4 million Google searches
  - 11 million instant messages
  - 170 million email messages
  - 1,800 TB of data
ArcGIS users have big data

- **Smart Sensors**
  - Electrical meters (AMI), SCADA, UAVs
- **GPS Telemetry**
  - Vehicle tracking, smartphone data collectors, workforce tracking, geofencing
- **Internet data**
  - Social media streams, web log files, customer sentiment
- **Sensor data**
  - Weather sensors, stream gauge measurements, heavy equipment monitors, ...
- **Imagery**
  - Satellites, frame cameras, drones
Value when analyzing data at mass scale

- As observations increase in frequency
  - Each individual observation is worth less
  - ...as the set of all observations becomes more valuable

- One single metric from the jet aircraft is much less useful than the analysis of that metric against the same metric from every known flight of that aircraft over time

- *Big Data* is the accumulation and analytical processes that uses this data for business value
Big challenges

• **Data acquisition**
  - Filtering and compressing
  - Planned Square Kilometer Array Telescope – one million terabytes per day

• **Information extraction and cleaning**
  - Extracting health info from X-rays and CT scans

• **Data integration, aggregation, and representation**
  - Heterogenous datasets

• **Modeling and analysis**
  - Nonstandard statistical analysis; very noisy, dynamic, and untrustworthy

• **Interpretation**
  - Decision making – metadata, assumptions, very complex
Big data
What techniques are applied to handle it?

• Data distribution – large datasets are split into smaller datasets and distributed across a collection of machines.

• Parallel processing – using a collection of machines to process the smaller datasets, combining the partial results together.

• Fault tolerance – making copies of the partitioned data to ensure that if a machine fails, the dataset can still be processed.

• Commodity hardware – using standard hardware that is not dependent upon exotic architectures, topologies, or data storage (e.g., RAID).

• Scalability – algorithms and frameworks that can be easily scaled to run on larger collections of machines in order to address larger datasets.

“Big data is not about the data.”
– Gary King
Harvard University
Director, Inst. For Quantitative Social Science

(Making the point that while data is plentiful and easy to collect, the real value is in the analytics)
Legacy system architecture

Processors

Disks

Network Bottleneck
Distributed system architecture
Until Now...

- Google implemented their enterprise on a distributed network of many nodes, fusing storage and processing into each node.

- Hadoop is an open source implementation of the framework that Google has built their business around for many years.
Apache Hadoop

Overview

- Hadoop is a scalable open source framework for the distributed processing of extremely large data sets on clusters of commodity hardware
  - Maintained by the Apache Software Foundation
  - Assumes that hardware failures are common

- Hadoop is primarily used for:
  - Distributed storage
  - Distributed computation
Apache Hadoop
Hadoop Clusters

Traditional Hadoop Clusters

The Dredd Cluster

20 flowdown PCs
5 – 7 years old
Quad-core, 3GHz
16 GB RAM
1 TB fast disk
Apache Hadoop

Distributed Storage

- The Hadoop Distributed File System (HDFS) is a hierarchical file system where datasets are organized into directories and files.

- These files are accessed like regular files, however they are actually distributed throughout the Hadoop cluster.
MapReduce

- A programming model for processing data with a parallel distributed algorithm on a cluster.
- A MapReduce program is comprised of:
  - A Map() procedure that performs filtering and comparing, and
  - A Reduce() procedure that performs a summary operation.
- The MapReduce system:
  - Marshals the distributed servers
  - Runs the various tasks in parallel
  - Manages all communications and data transfers
  - Provides for redundancy and failures
- MapReduce libraries have been written in many languages; Hadoop is a popular open source implementation.
MapReduce Walkthrough

- An instance of a MapReduce program is a job
- The job accepts arguments for data input and outputs

- The job combines
  - Functions from the MapReduce framework
    - “Splitting” large inputs into smaller pieces
    - Reading inputs and writing outputs
  - Functions that are written by the application developer
    - Map function, maps input values to keys
    - Reduce function, reduces many keys to one
High Level MapReduce Walk-Through

- An instance of a MapReduce program is a job
High Level MapReduce Walk-Through

- The job accepts arguments for data input and outputs

MapReduce Job

hdfs://path/to/input

data

hdfs://path/to/output
High Level MapReduce Walk-Through

- The MapReduce framework logically splits the data space
High Level MapReduce Walk-Through

- Many map() functions are run in parallel
High Level MapReduce Walk-Through

- The framework runs combine to join map() outputs
High Level MapReduce Walk-Through

- The framework performs a shuffle and sort on all data
High Level MapReduce Walk-Through

- The reduce() functions work against the sorted data
High Level MapReduce Walk-Through

• Reducers each write a part of the results to a file

MapReduce Job

Split
Combine
Shuffle / Sort

map()
map()
map()
reduce()
reduce()
reduce()
part 1
part 2

hdfs://path/to/input
hdfs://path/to/output
When the job has finished, ArcGIS can retrieve the results.
MapReduce – Word Count Example

Input text file:
- red red
- green
- blue red
- green
- green
- blue blue
- blue

Split: 1 / Record 1:
- red red
- red
- blue
- red
- green

Split: 2 / Record 1:
- green
- blue red
- green

Split: 3 / Record 1:
- green
- blue blue
- blue

Map:
- red 1
- red 1
- blue 1
- red 1
- green 1

Map:
- green 1
- blue 1
- red 1
- green 1

Shuffle / Sort:
- green
- green
- green
- red
- red
- red
- red

Reduce:
- green 4
- red 4
- blue 5

Output text file:
- green
- red
- blue

Output writer:
- green 4
- red 4
- blue 5
Apache Hive
Tables and Queries in Hadoop

- Part of the Hadoop ecosystem
- Provides the ability to interact with data in Hadoop as if it were tables with rows and columns
- Supports a SQL-like language for querying the data in Hadoop

Note: The source files are never modified by Hive. Tables are merely a view of the data.

SELECT * FROM earthquakes WHERE magnitude > 7
GIS Tools for Hadoop
Esri on GitHub

- Tools and samples using the open source resources that solve specific problems
  - Hive user-defined functions for spatial processing
  - JSON helper utilities

- Geoprocessing tools that...
  - Copy to/from Hadoop
  - Convert to/from JSON
  - Invoke Hadoop Jobs

- Java geometry library for spatial data processing

Hive user-defined functions for spatial processing
JSON helper utilities
Copy to/from Hadoop
Convert to/from JSON
Invoke Hadoop Jobs
Java geometry library for spatial data processing

https://github.com/Esri/gis-tools-for-hadoop
ST_Geometry
Hive Spatial Type

- Hive functions wrapped around the freely available Java Geometry API
- Supports simple geometry types
  - Point, polyline, polygon
- Distributed geometric operations
  - Topological - union, intersection, cut, …
  - Relational - contains, intersects, crosses, …
  - Aggregate - union, convex hull, intersection, …
  - Others - geodesic distance, buffer, …
- Supports multiple geometry formats
  - Well-known text/binary
  - GeoJSON
  - Esri JSON

```
SELECT count(*) FROM faa
WHERE ST_Contains('POLYGON ((-124 32, -124 42, -114 42, -114 32))', ST_Point(faa.longitude, faa.latitude))
```

Select a count of all points that are contained within the given polygon

https://github.com/Esri/spatial-framework-for-hadoop/wiki/Hive-Spatial
Spatial Aggregation
Making Big Data Manageable

- Reduces the size of the data
- Maintains a summary of numeric attributes
- Common aggregation patterns
  - Polygons
    - Aggregate points into polygons
  - Bins
    - Aggregate points into bins defined by a grid
I don’t need it. I’ll just draw everything.
FAA flight data – 17.5 million points
Spatial Aggregation

Why it’s important

- Mapping millions to billions of points in a small, dense area just isn’t feasible
  - It’s slow
  - It’s difficult to identify patterns

FAA flight data – 17.5 million points
Spatial Aggregation

Why it's important

- Mapping millions to billions of points in a small, dense area just isn't feasible
  - It's slow
  - It's difficult to identify patterns
FAA flight data – 17.5 million points (spatially binned)
Spatial Aggregation
With Bins

- Spatial bins are cells defined by a regular grid
- Each cell contains a set of attributes summarizing all points that fall within the cell boundary
Spatial Aggregation
With Bins

Points in bin 2

<table>
<thead>
<tr>
<th>Bin ID</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- speed = 10
- speed = 30

- count = 2
- speed sum = 40
- speed avg = 20
Spatial Aggregation
Binning with Hive and GIS Tools

- Hive functions for spatial binning
  - **ST_Bin** – given a point, return an integer value that uniquely identifies the containing bin
  - **ST_BinEnvelope** – given a point (or bin id), return the envelope (bounding box) of the bin

- With these functions together, we can create a spatially binned table using Hive

```
ST_Bin(0.5, 'POINT (10 10)')             ->  4611685954723114540
ST_BinEnvelope(0.5, 'POINT (10 10)')     -> POLYGON ((10 9.5, 10.5 9.5, 10.5 10, 10 10, 10 9.5))
ST_BinEnvelope(0.5, 4611685954723114540) -> POLYGON ((10 9.5, 10.5 9.5, 10.5 10, 10 10, 10 9.5))
```
Spatial Aggregation
Spatial Query Example

Bin size is 0.5 degrees

Source data is 17.5 million FAA flight points
Spatial Aggregation
Spatial Query Example

Subquery to augment each flight record with the ID of the bin that contains it
Spatial Aggregation
Spatial Query Example

Select the envelope of the bin and a count of each point in that bin.

<table>
<thead>
<tr>
<th>shape</th>
<th>count</th>
</tr>
</thead>
<tbody>
<tr>
<td>POLYGON((144.44 13.85, 144.54 13.85, 144.54 13.95, 144.44 13.95, 144.44 13.85))</td>
<td>33</td>
</tr>
<tr>
<td>POLYGON((144.34 13.75, 144.44 13.75, 144.44 13.85, 144.34 13.85, 144.34 13.75))</td>
<td>27</td>
</tr>
<tr>
<td>POLYGON((144.54 13.75, 144.64 13.75, 144.64 13.85, 144.54 13.85, 144.54 13.75))</td>
<td>49</td>
</tr>
<tr>
<td>POLYGON((144.94 13.75, 145.04 13.75, 145.04 13.85, 144.94 13.85, 144.94 13.75))</td>
<td>39</td>
</tr>
<tr>
<td>POLYGON((144.34 13.65, 144.44 13.65, 144.44 13.75, 144.34 13.75, 144.34 13.65))</td>
<td>29</td>
</tr>
<tr>
<td>POLYGON((144.94 13.65, 145.04 13.65, 145.04 13.75, 144.94 13.75, 144.94 13.65))</td>
<td>75</td>
</tr>
<tr>
<td>POLYGON((144.34 13.55, 144.44 13.55, 144.44 13.65, 144.34 13.65, 144.34 13.55))</td>
<td>25</td>
</tr>
<tr>
<td>POLYGON((144.94 13.55, 145.04 13.55, 145.04 13.65, 144.94 13.65, 144.94 13.55))</td>
<td>71</td>
</tr>
<tr>
<td>POLYGON((144.64 13.45, 144.74 13.45, 144.74 13.55, 144.64 13.55, 144.64 13.45))</td>
<td>189</td>
</tr>
</tbody>
</table>
Spatial Aggregation
With Polygons

- Aggregate points into polygons instead of bins
- Works best with a small set of polygons

```
1 SELECT counties.shape, count(*)
2 FROM counties JOIN faa
3 WHERE ST_Contains(counties.shape, 
4    ST_Point(faa.longitude, faa.latitude))
5 GROUP BY counties.shape;
```
GIS Tools for Hadoop
Visualizing your data in ArcGIS

- A model was used to bring the output of our Hive queries back into ArcGIS as a local feature class
  - Models let you chain geoprocessing tools together
  - The output of one tool becomes the input to the next tool
GIS Tools for Hadoop
Visualizing your data in ArcGIS

Copy From HDFS – Copy a file or directory from Hadoop to a local file

JSON To Features – Create a feature class from a JSON file
GIS Tools for Hadoop
Visualizing your data in ArcGIS

Copy From HDFS – Copy a file or directory from Hadoop to a local file

Hadoop service information

Hadoop path to the result of the Hive aggregation query

Path to temporary file on local hard disk

JSON To Features – Create a feature class from a JSON file

Output feature class to create

JSON file type (usually unenclosed)
Other relevant sessions

• **Big Data and Analytics: Getting Started with ArcGIS**
  - Th 10:15 – 11:30 (Room 4)

• **Big Data and Analytics: Application Examples**
  - We 1:00 – 1:30 (Tech Theater 16)

• **Real-Time GIS: Applying Real-Time Analytics**
  - We 8:30 – 9:45 (Room 14B)

• **Real-Time GIS: The Road Ahead**
  - We 1:30 – 2:45 (Room 14B)

• **Road Ahead: ArcGIS for Server**
  - We 10:15 – 11:30 (Room 7/8)
Thanks for coming

- Be sure to fill out the session survey using your mobile app
  - Select the session (search works well)
  - Select Technical Workshop Survey
  - Answer a couple questions and provide any comments you feel appropriate