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

Big Data

- 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

Age of Data Ubiquity

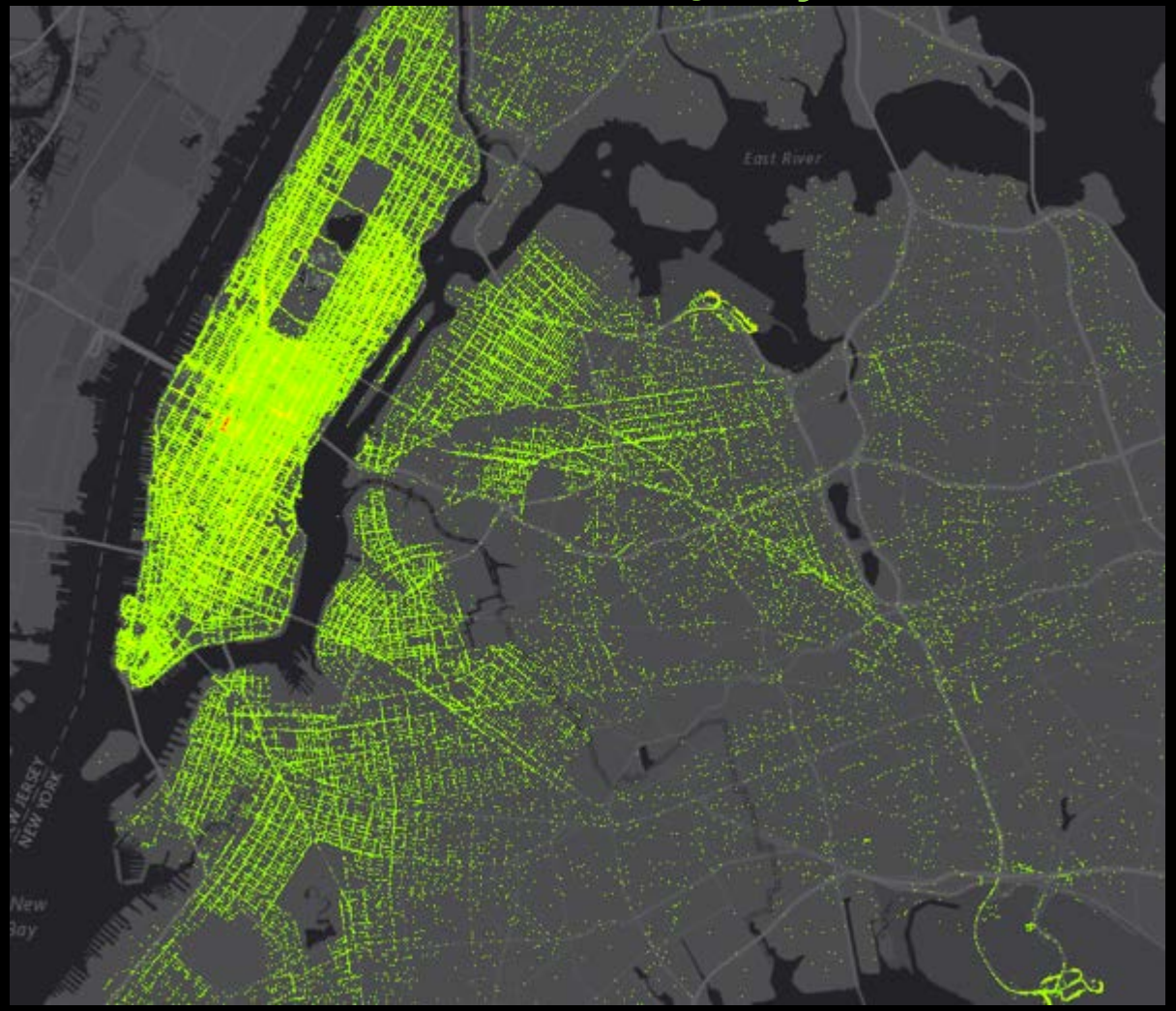
- Data is now central to our existence – both for corporations and individuals
- Nimble, thin, data-centric apps exploiting massive data sets generated by both enterprises and consumers
- Hardware era: 20 – 30 years
- Software era: 20 – 30 years
- Data era: ?

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

NYC Taxis by Day



Manhattan Taxis Friday after 8pm

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 Suare 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**
  - Heterogenious 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 distribu** …oss a collection of
- **Parallel proc** …, combining the partial re
- **Fault toleran** …fails, the dataset can s
- **Commodity** …c architectures
- **Scalability** – …ollections of machines in order to address larger datasets

“Big data is not about the data.”

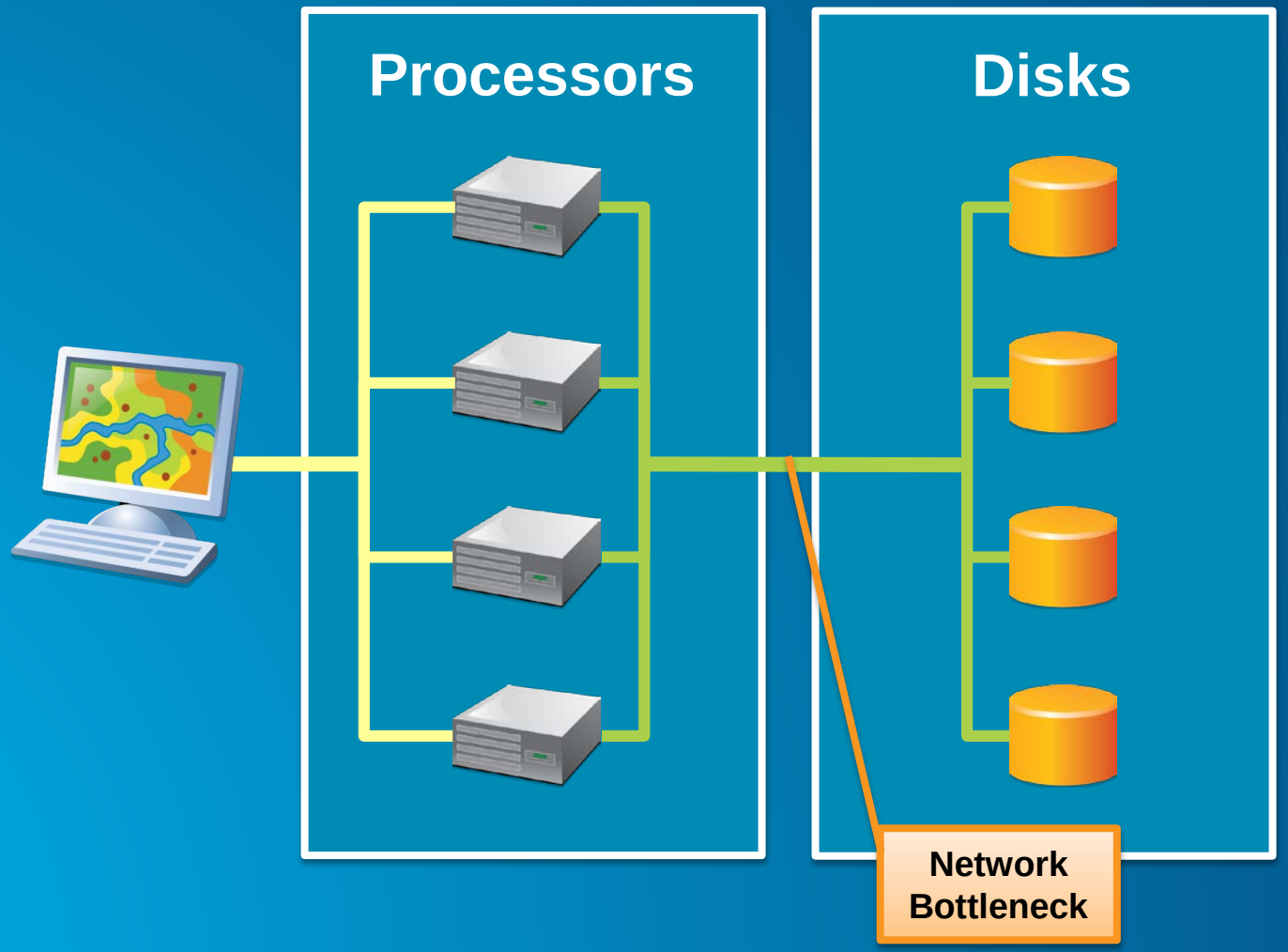
– Gary King

Harvard University

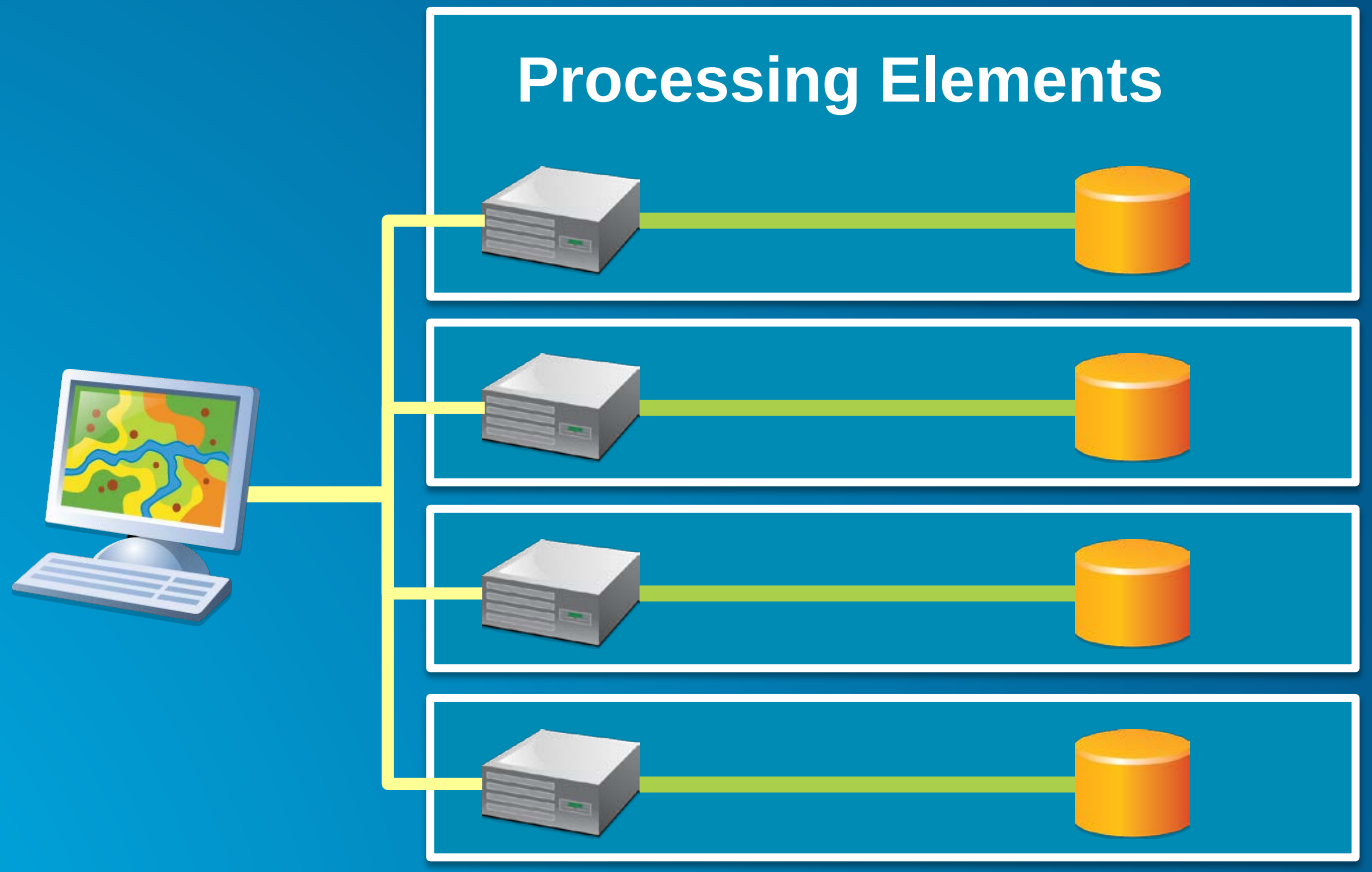
Director, Inst. For Quantitative Social Science

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

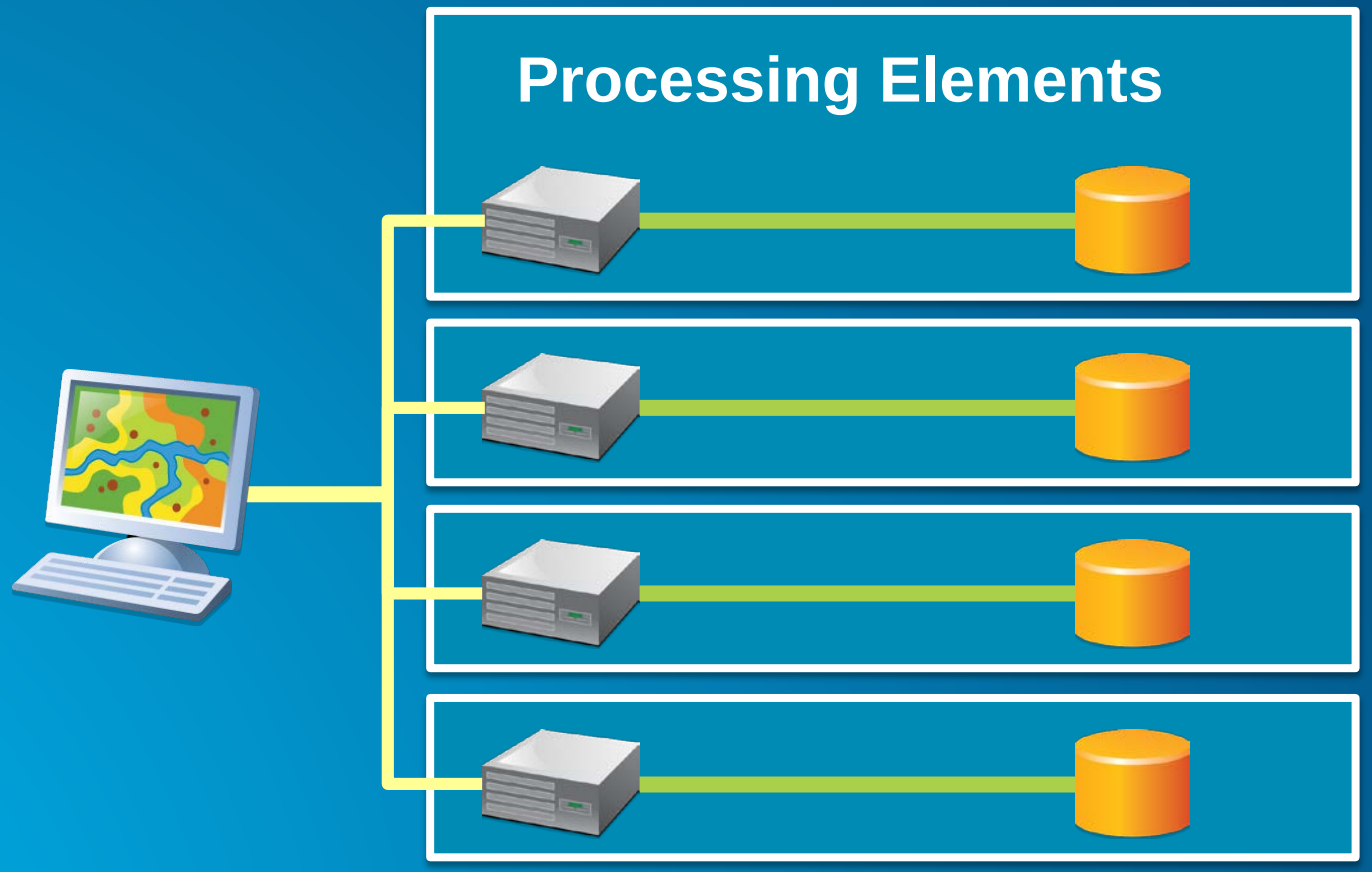
Legacy system architecture



Distributed system architecture

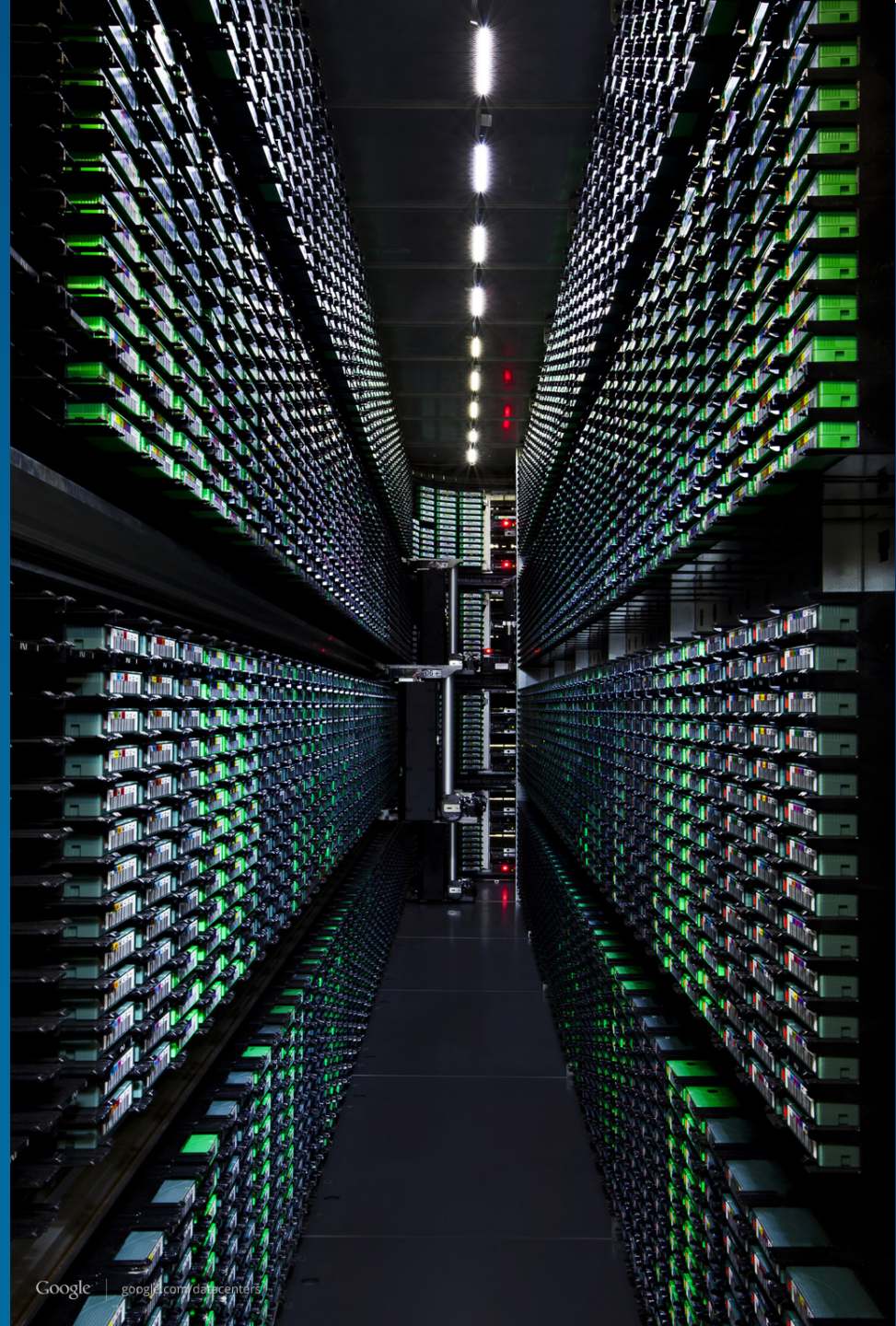


Distributed system architecture



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

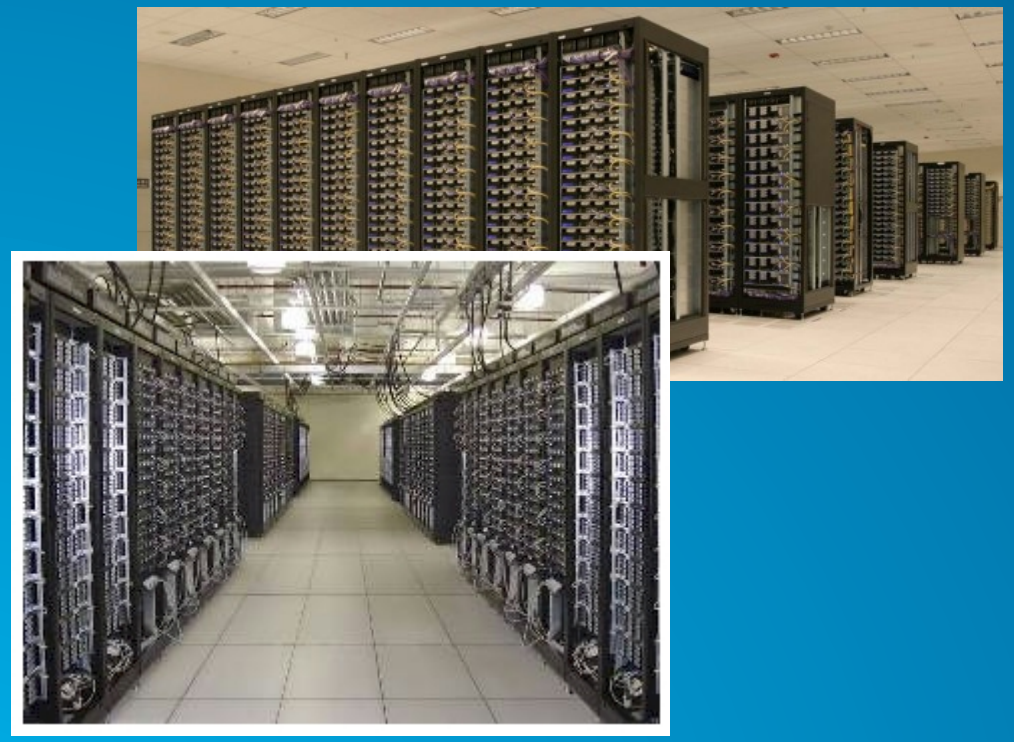
Overview

- 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

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

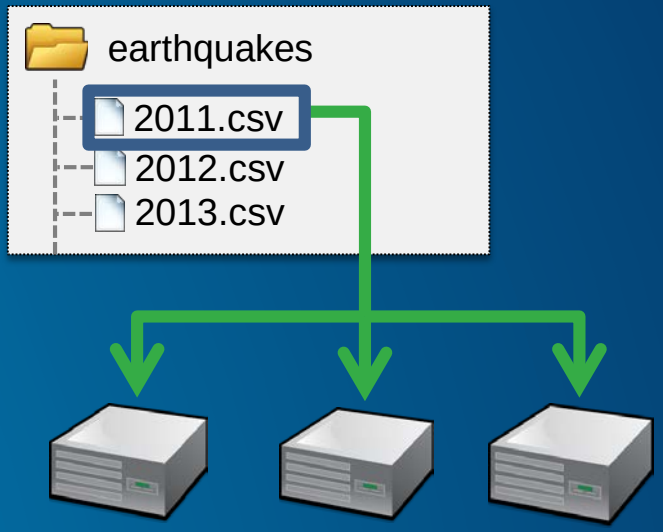
The Dredd Cluster

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

MapReduce Job

High Level MapReduce Walk-Through

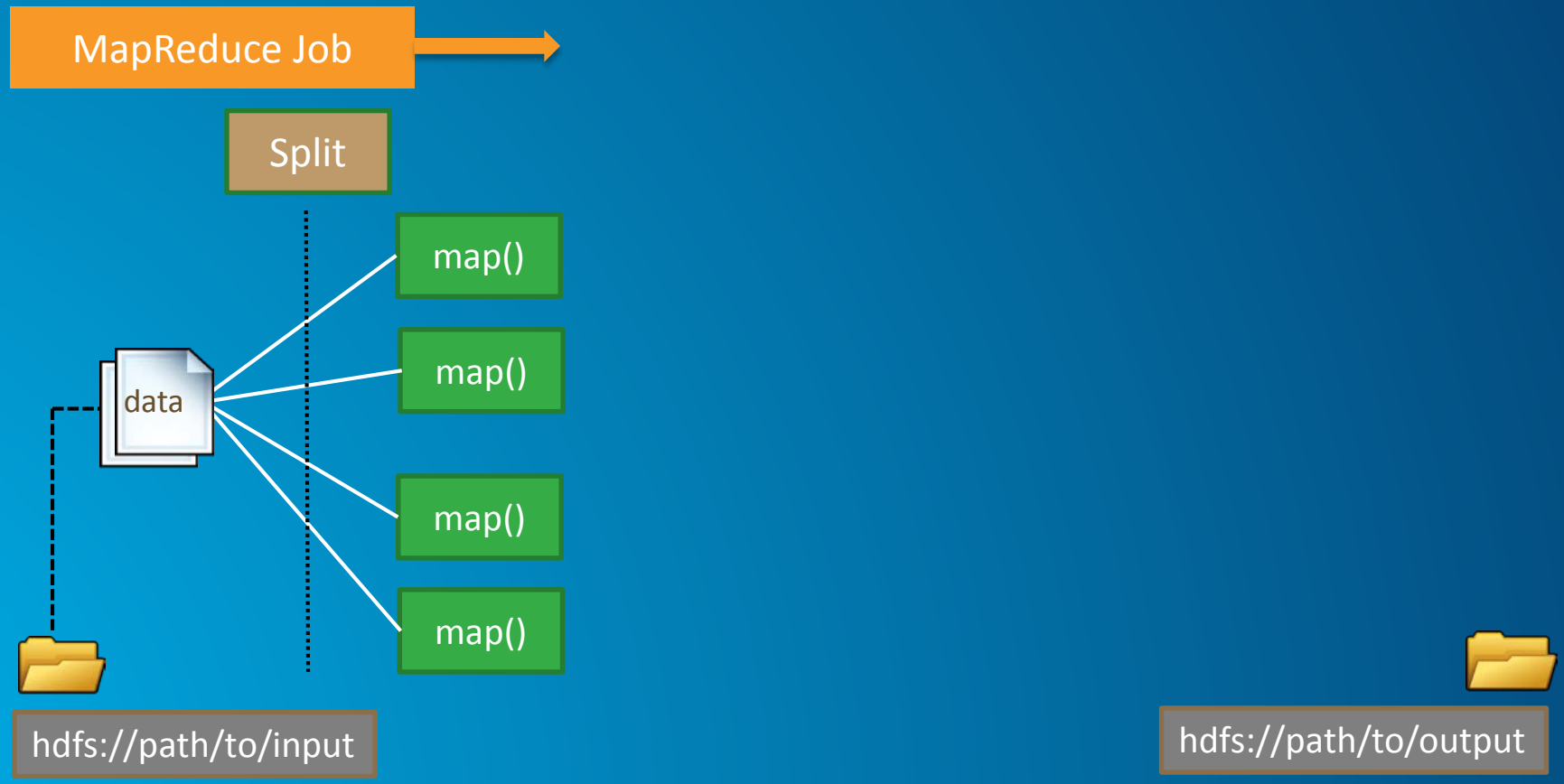
- The job accepts arguments for data input and outputs

High Level MapReduce Walk-Through

- The MapReduce framework logically splits the data space

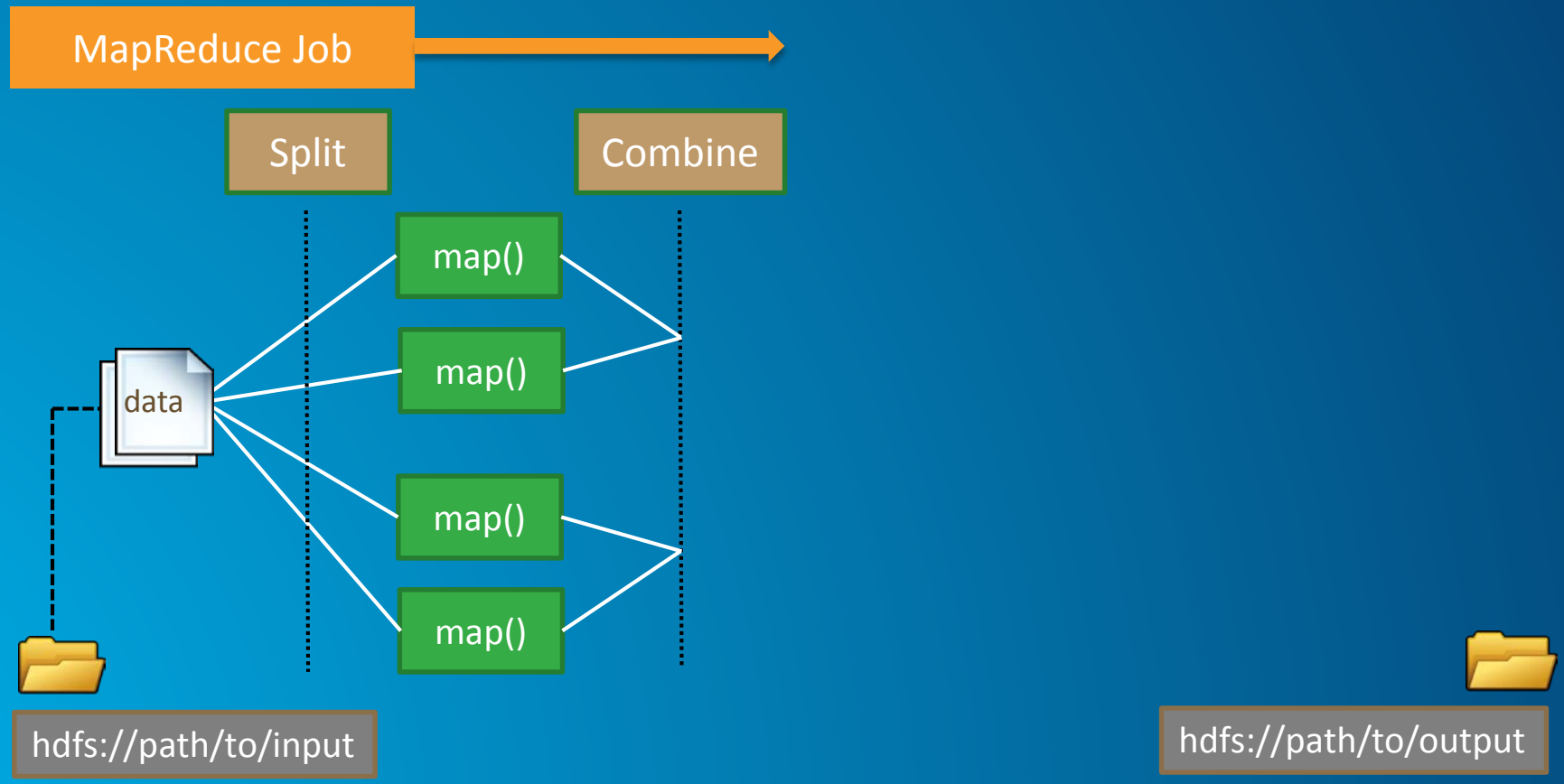
High Level MapReduce Walk-Through

- Many map() functions are run in parallel



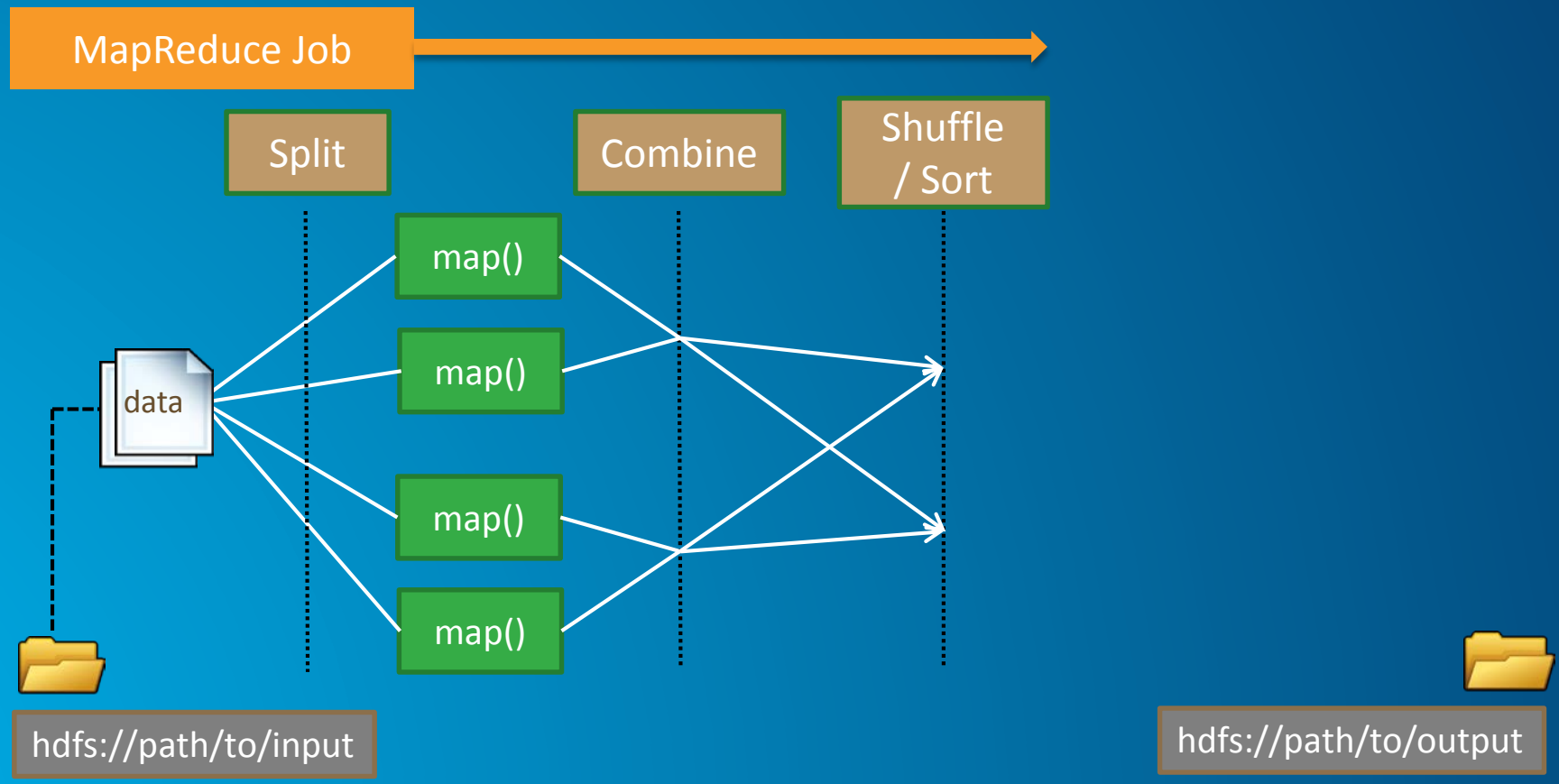
High Level MapReduce Walk-Through

- The framework runs combine to join map() outputs



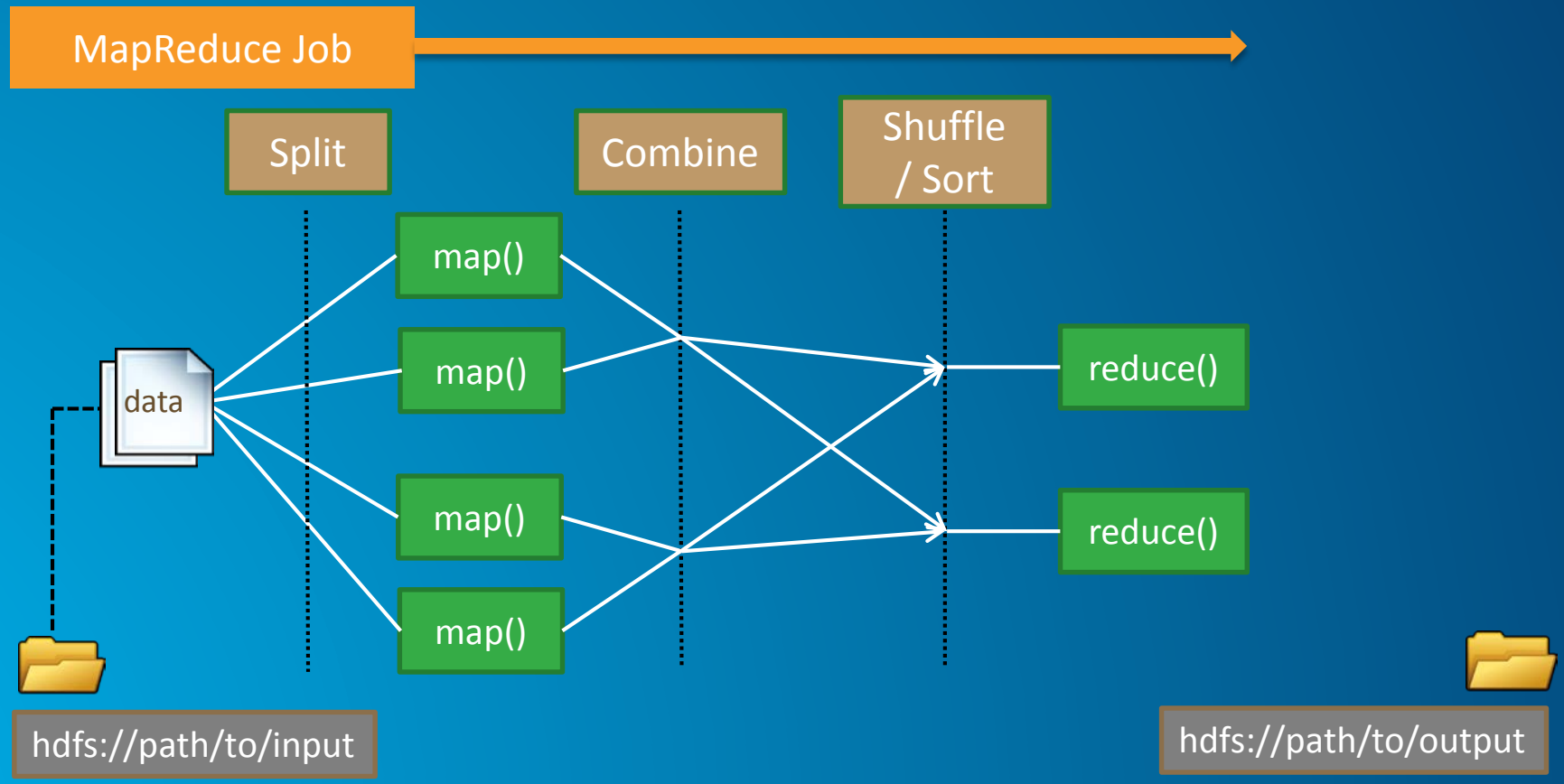
High Level MapReduce Walk-Through

- The framework performs a shuffle and sort on all data



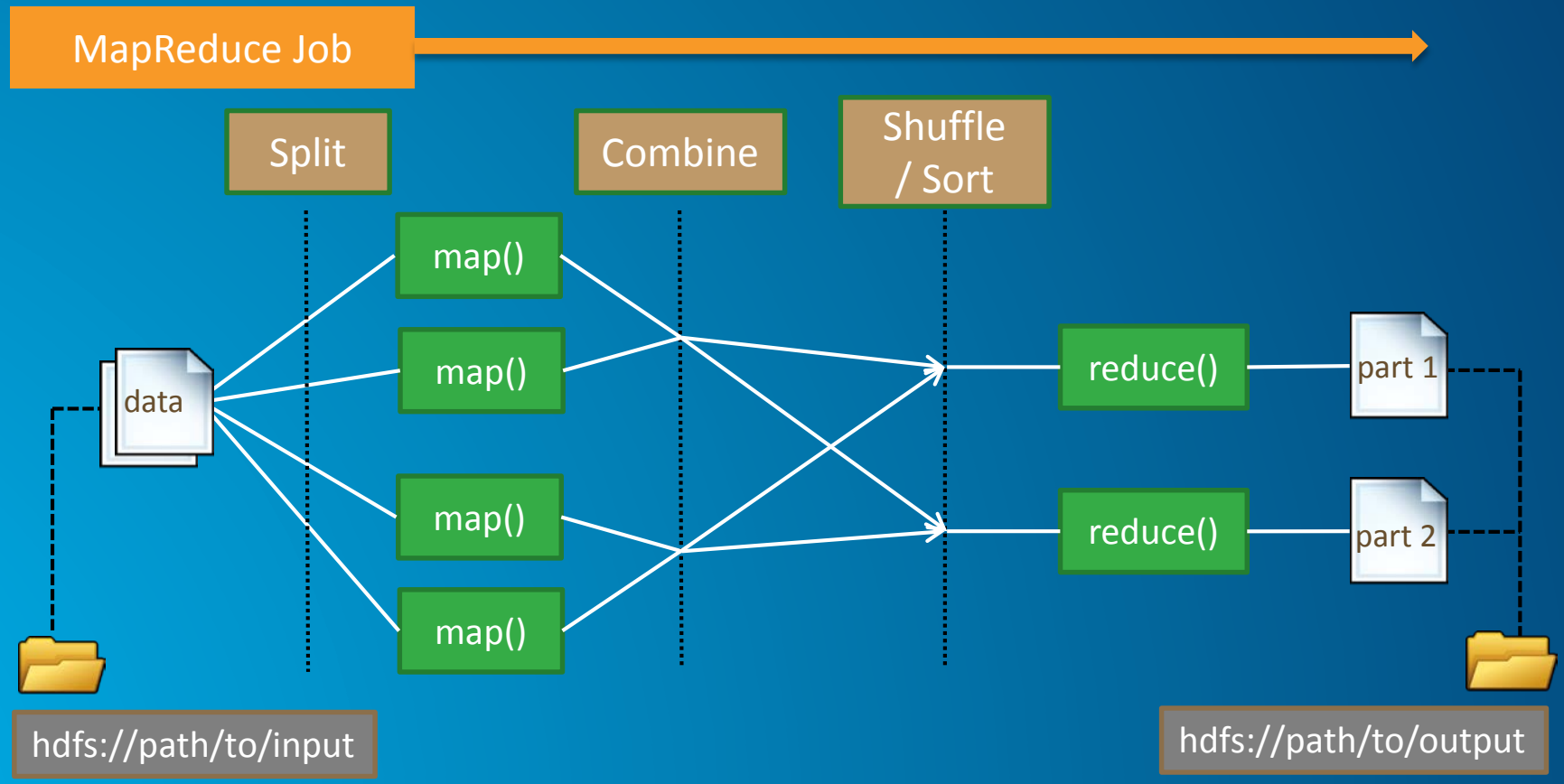
High Level MapReduce Walk-Through

- The reduce() functions work against the sorted data



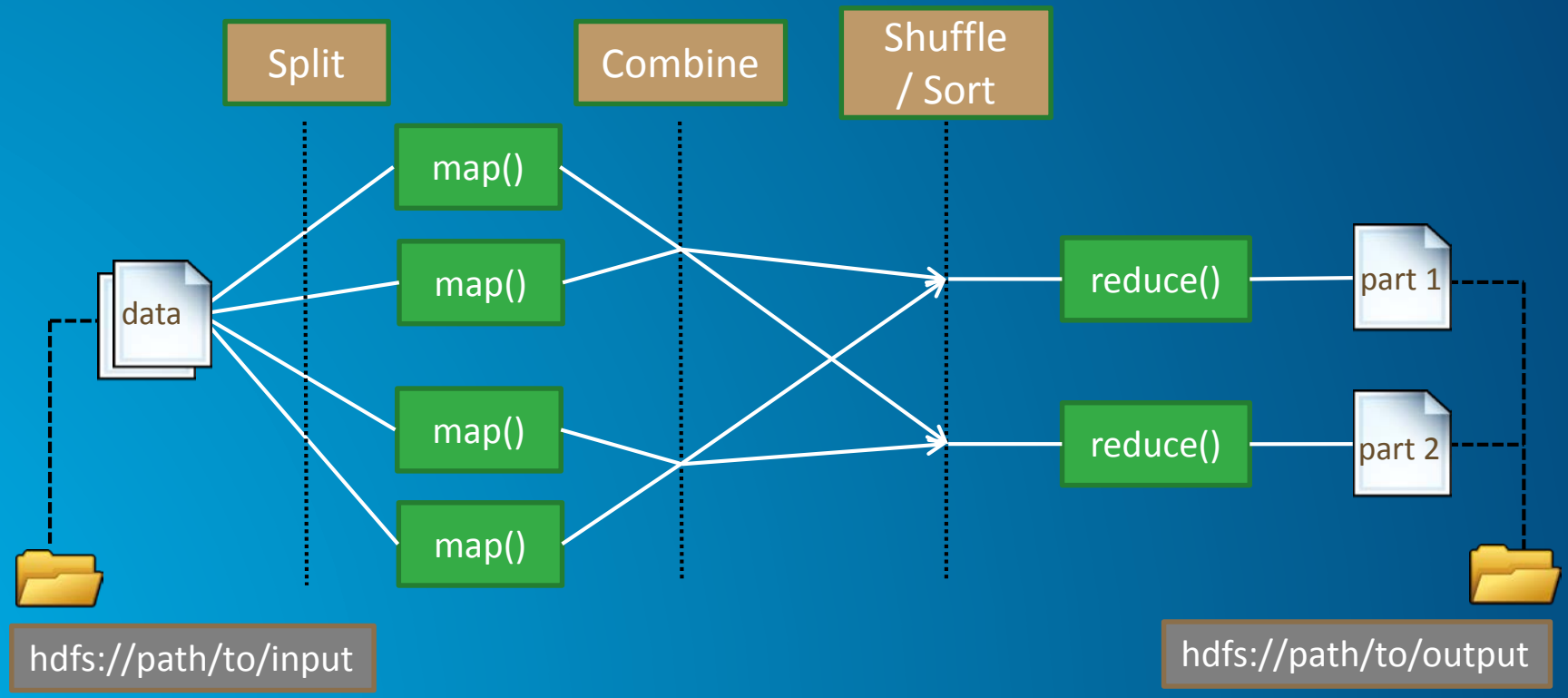
High Level MapReduce Walk-Through

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

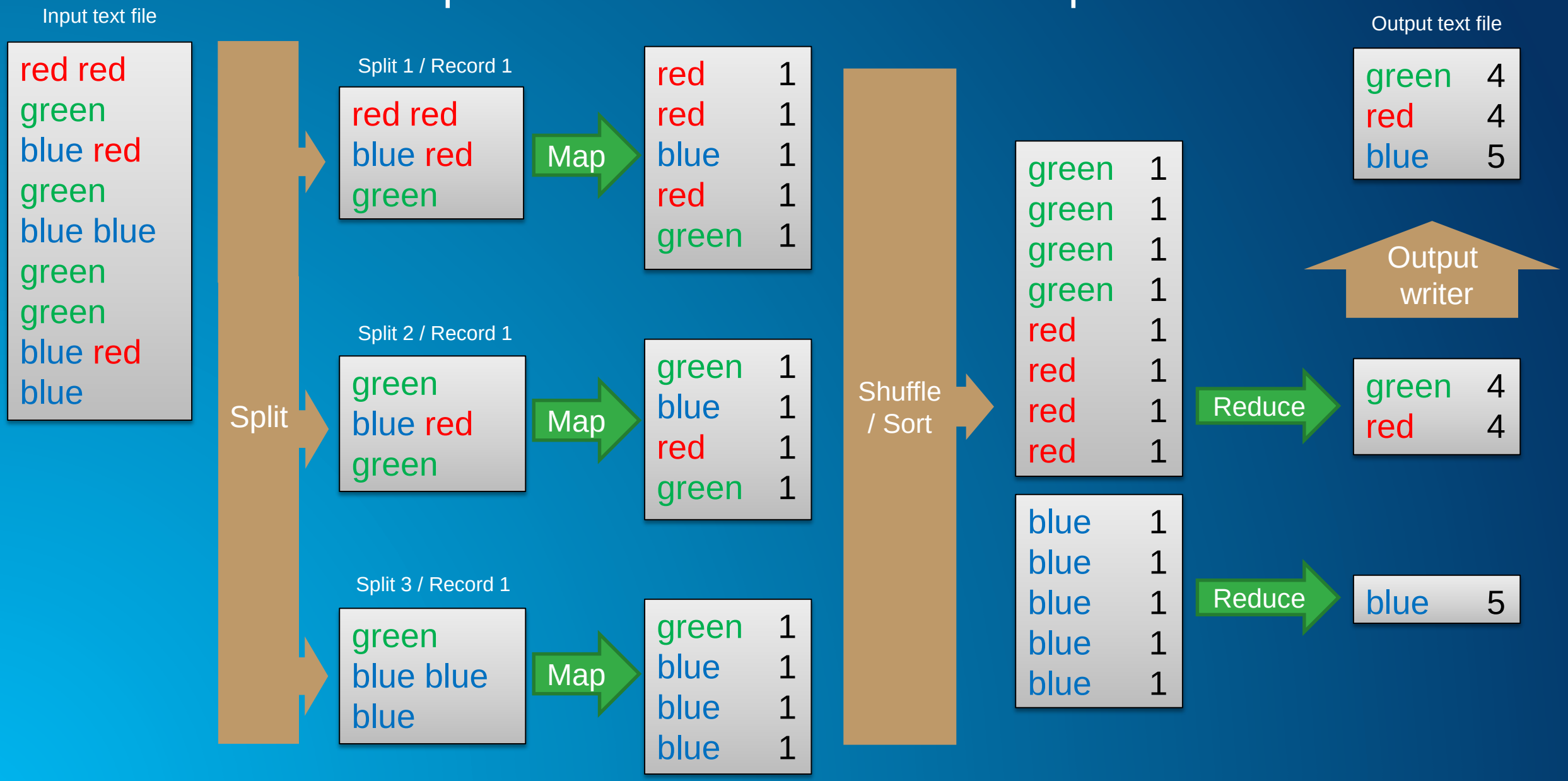


High Level MapReduce Walk-Through

- When the job has finished, ArcGIS can retrieve the results



MapReduce – Word Count Example

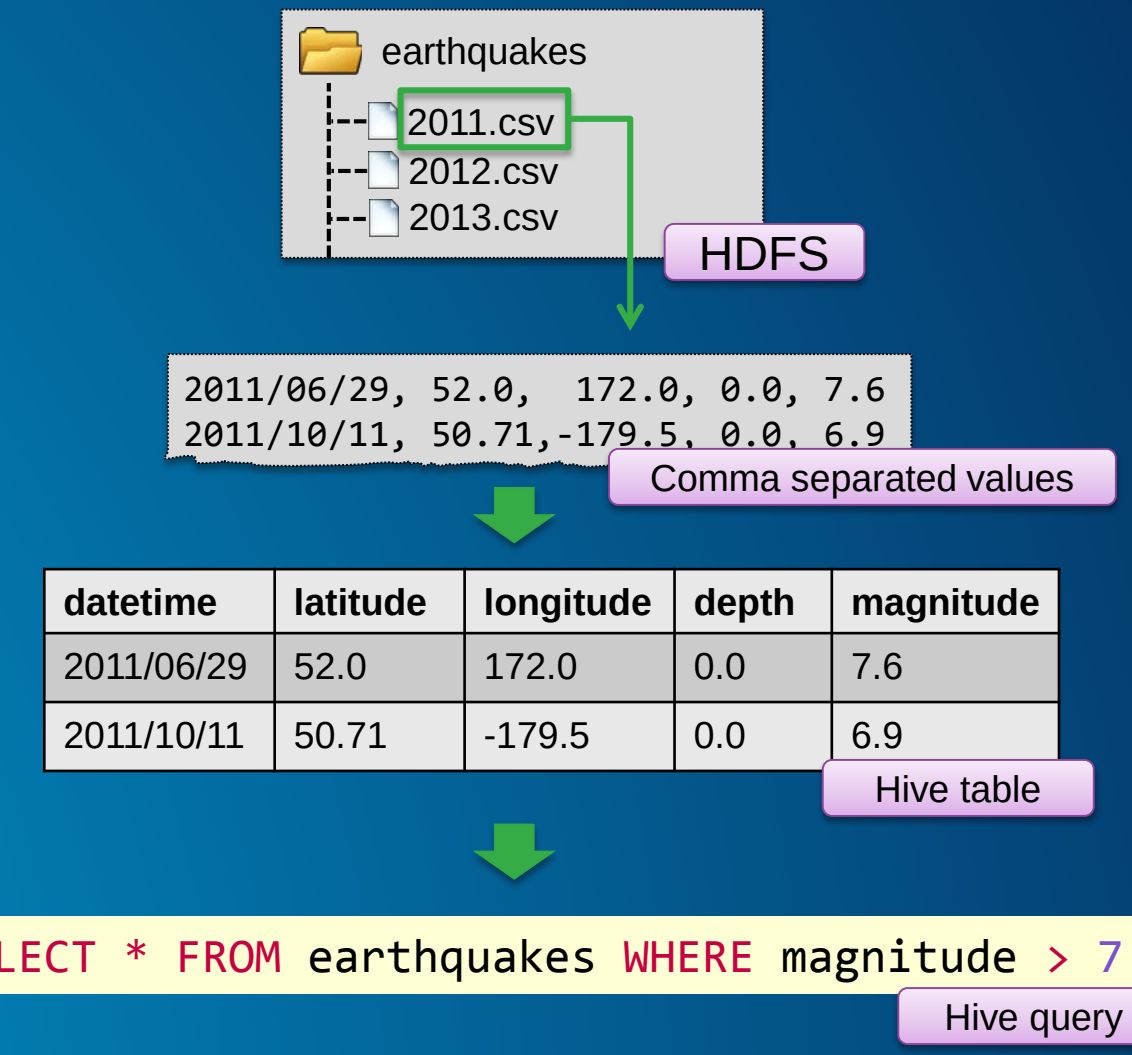


Apache Hive

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**

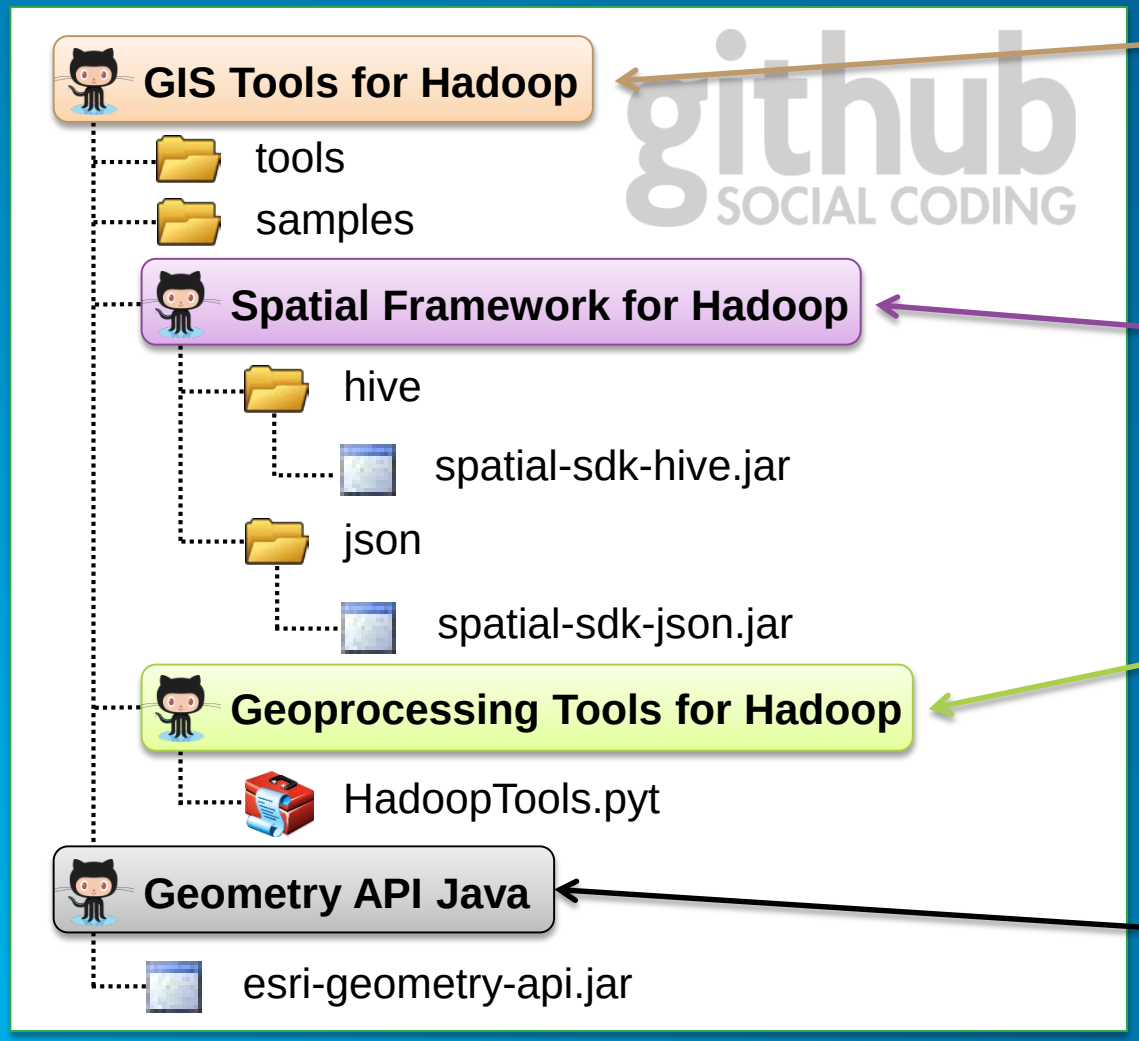
Note: The source files are never modified by Hive. Tables are merely a view of the data.



GIS Tools for Hadoop

Esri on GitHub

Show list of samples and tools



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

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

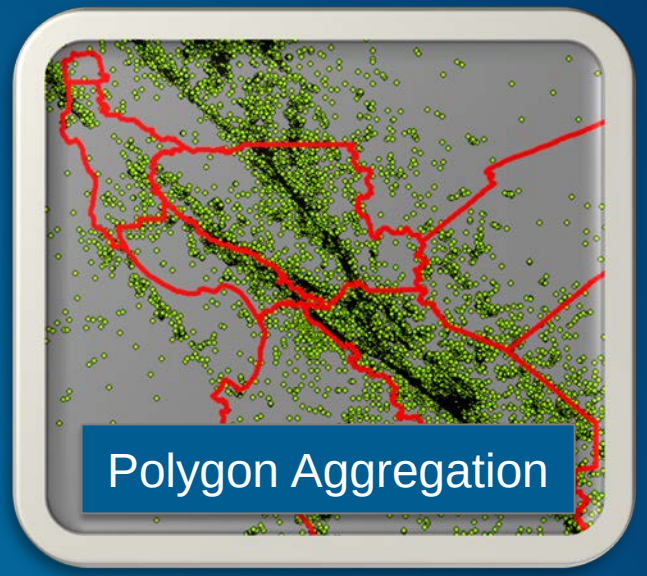
```
1 SELECT count(*) FROM faa
2 WHERE ST_Contains('POLYGON ((-124 32, -124 42, -114 42, -114 32))',
3                   ST_Point(faa.longitude, faa.latitude))
```

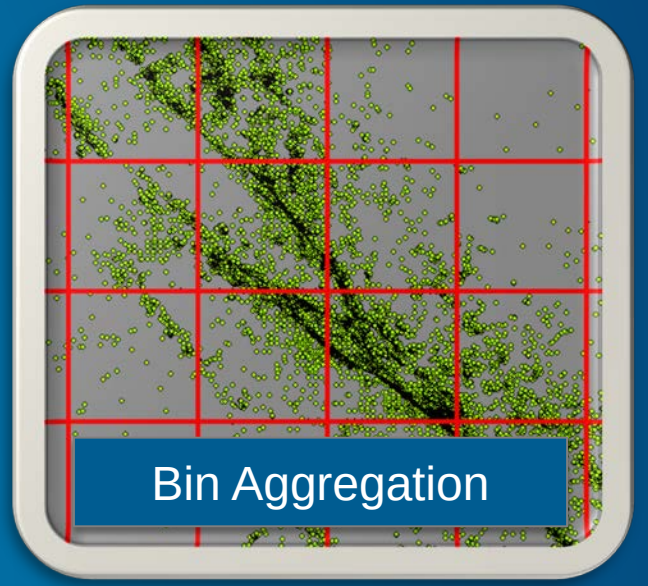
https://github.com/Esri/spatial-framework-for-hadoop/wiki/Hive-Spatial

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

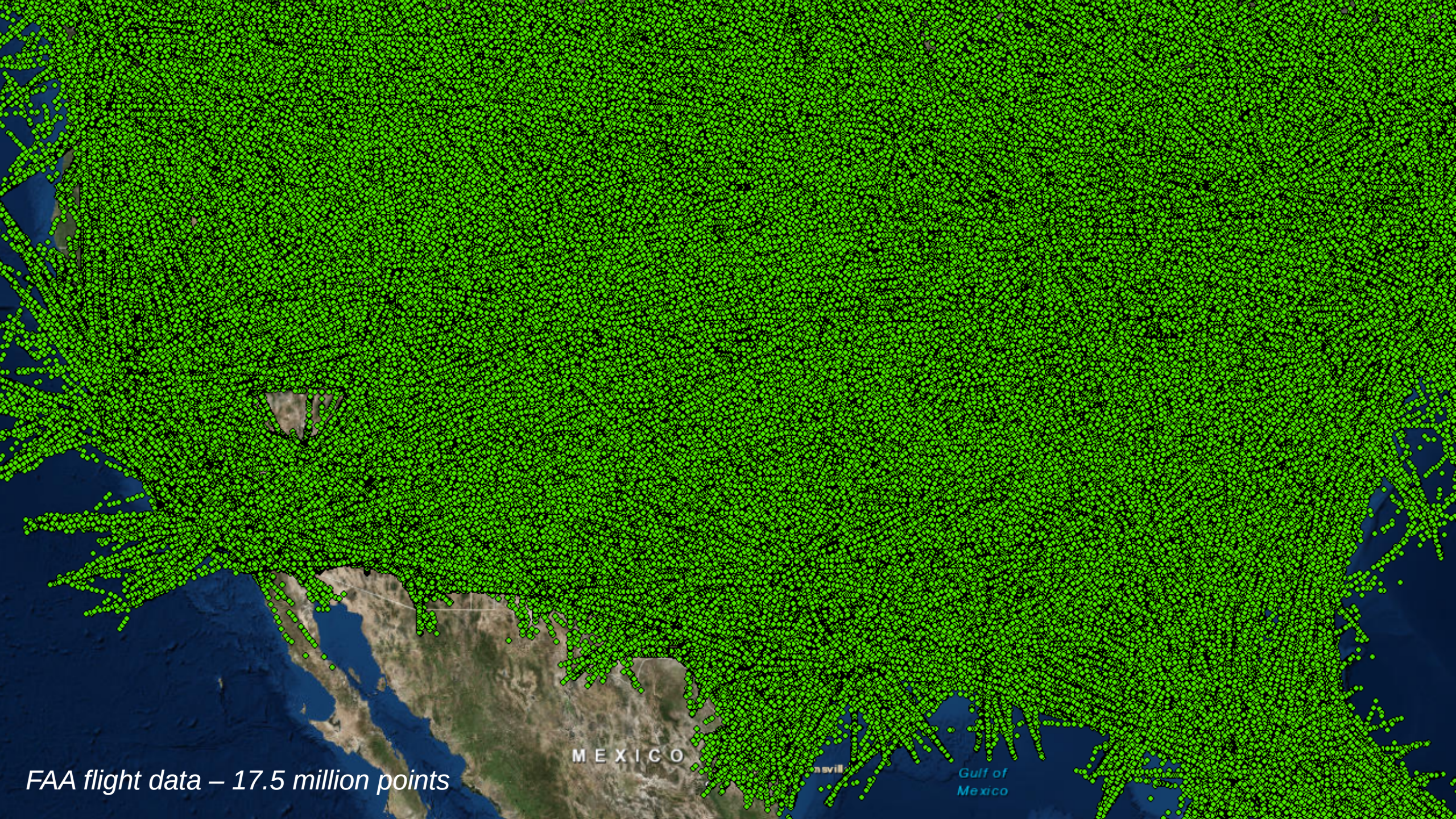




Spatial Aggregation

Who needs it?

I don’t need it. I’ll just draw everything.

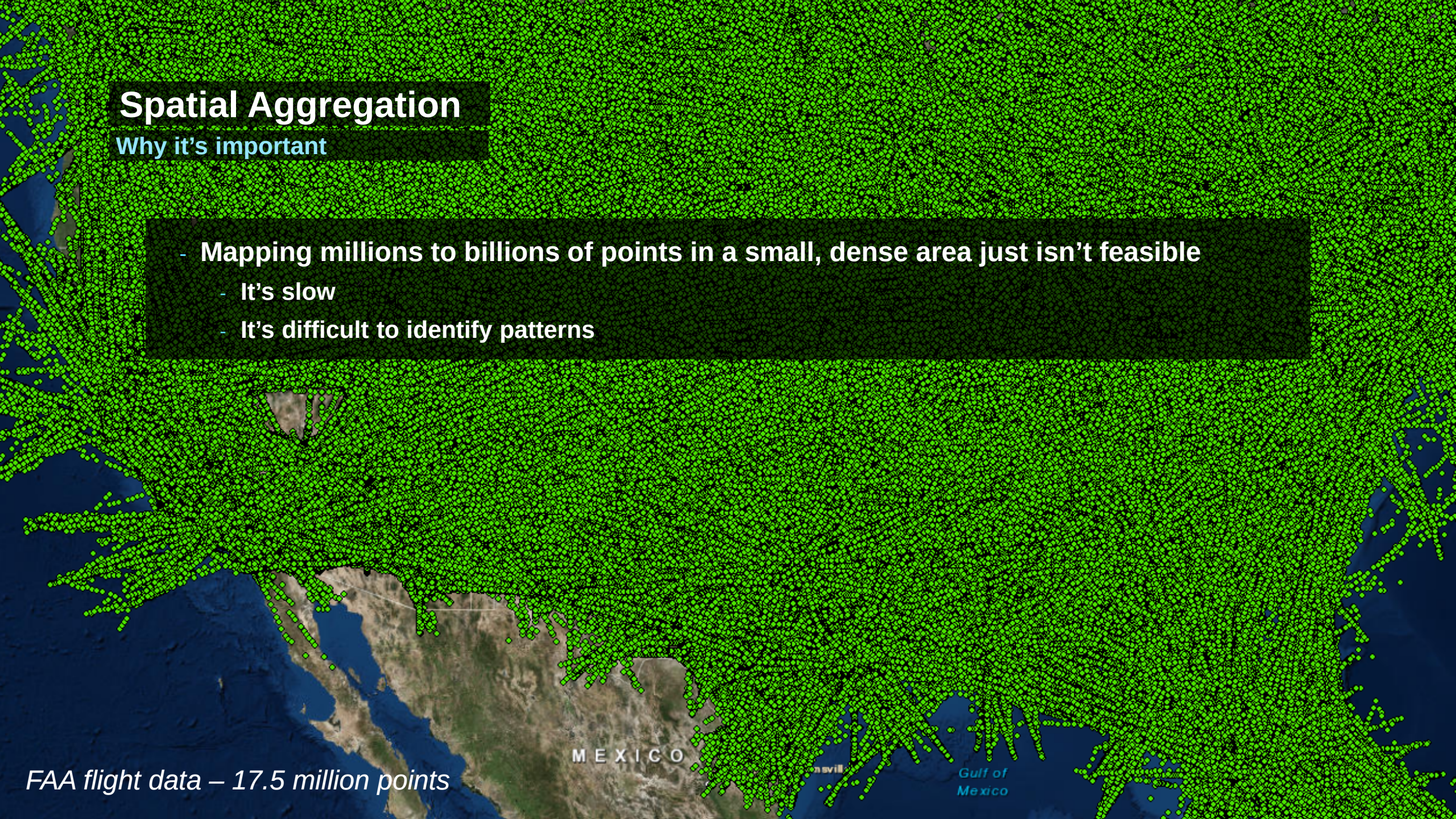


FAA flight data – 17.5 million points

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

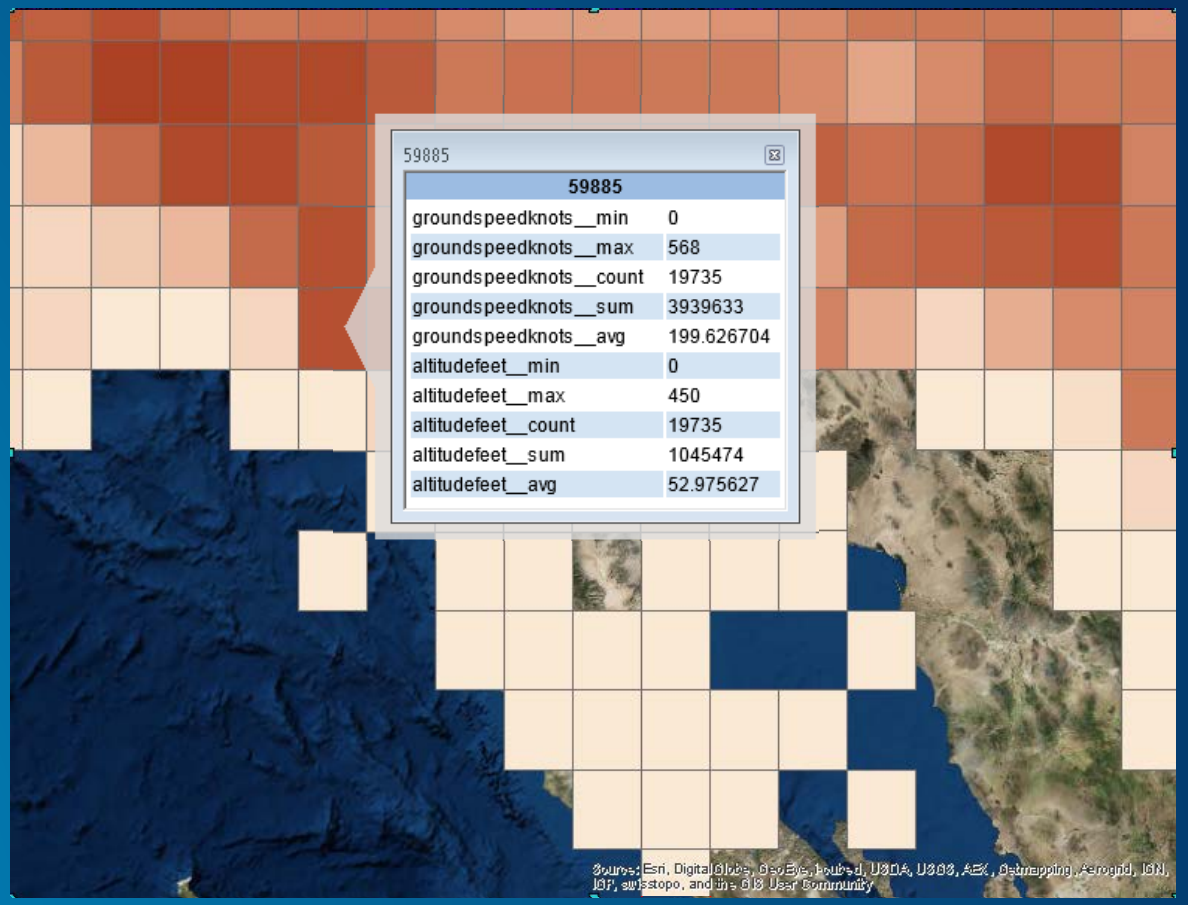
- Mapping millions to billions of points in a small, dense area just isn't feasible
  - It's slow
  - It's difficult to identify patterns

FAA flight data – 17.5 million points (spatially binned)

Spatial Aggregation

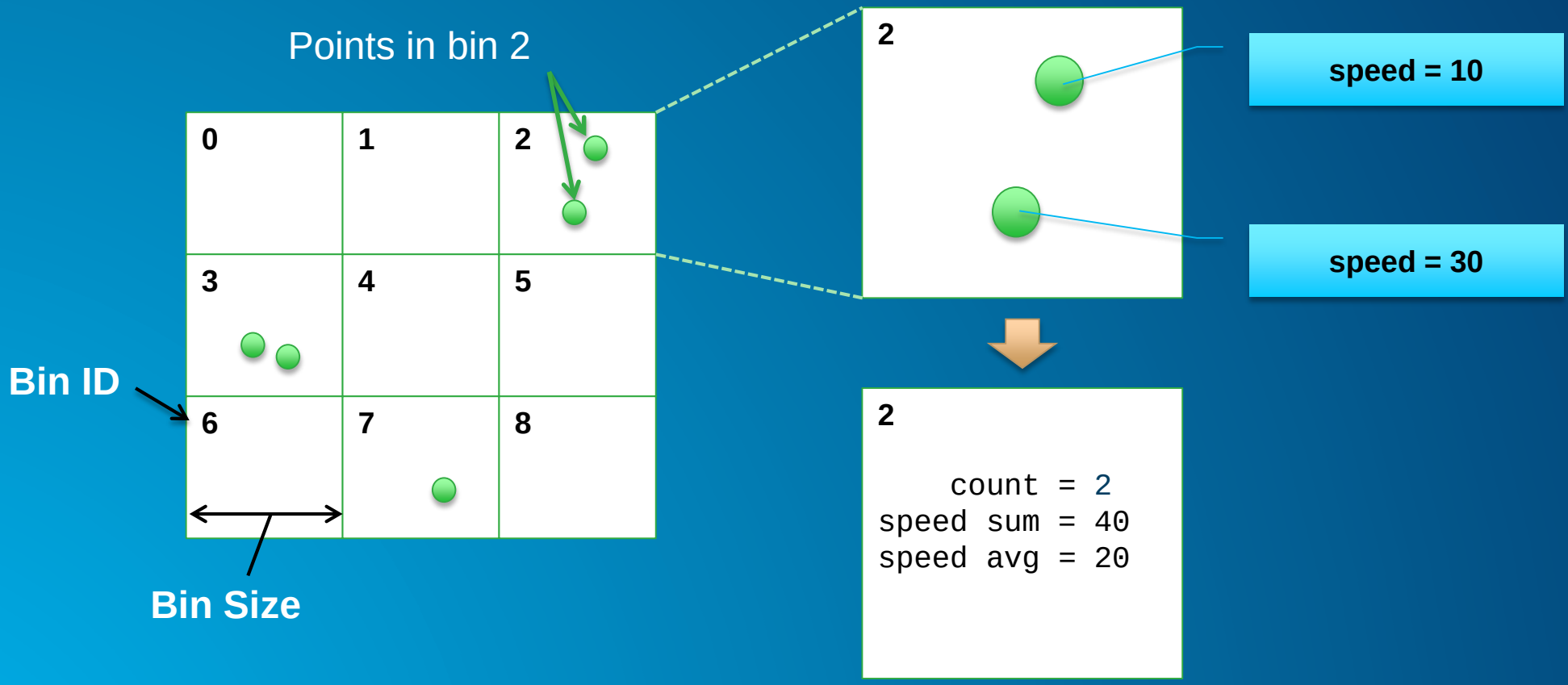
With Bins

- 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



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**

```
1 FROM (SELECT ST_Bin(0.5, ST_Point(longitude, latitude)) bin_id, * FROM faa) bins
2 SELECT ST_BinEnvelope(0.5, bin_id)   shape,
3        COUNT(*)                      count
4 GROUP BY bin_id
```

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

```
1 FROM (SE                                    oint(longitude,                    bins
2 SELECT S                                    n_id)    shape,
3         C                                              t
4 GROUP BY
```

...	latitude	longitude
...	41.733	-90.65
...	34.966	-119.05
...	42.016	-76.75
...	40.966	-120.73
...	37.616	-122.03
...	48.466	-122.51
...	32.766	-116.58
...	33.383	-97.633
...	34.75	-119.41
...	39.466	-104.93

FAA

...	latitude	longitude	bin_id
...	41.733	-90.65	46116857633
...	34.966	-119.05	46116858028
...	42.016	-76.75	46116857603
...	40.966	-120.73	46116857664
...	37.616	-122.03	46116857876
...	48.466	-122.51	46116857208
...	32.766	-116.58	46116858150
...	33.383	-97.633	46116858119
...	34.75	-119.41	46116858059
...	39.466	-104.93	46116857755

FAA with bin ID

Spatial Aggregation

Spatial Query Example

One record per bin

```
1 FROM (SELECT ST_
2 SELECT ST_BinEnv
3        COUNT(*)
4 GROUP BY bin_id
```

```
FROM faa) bins
```

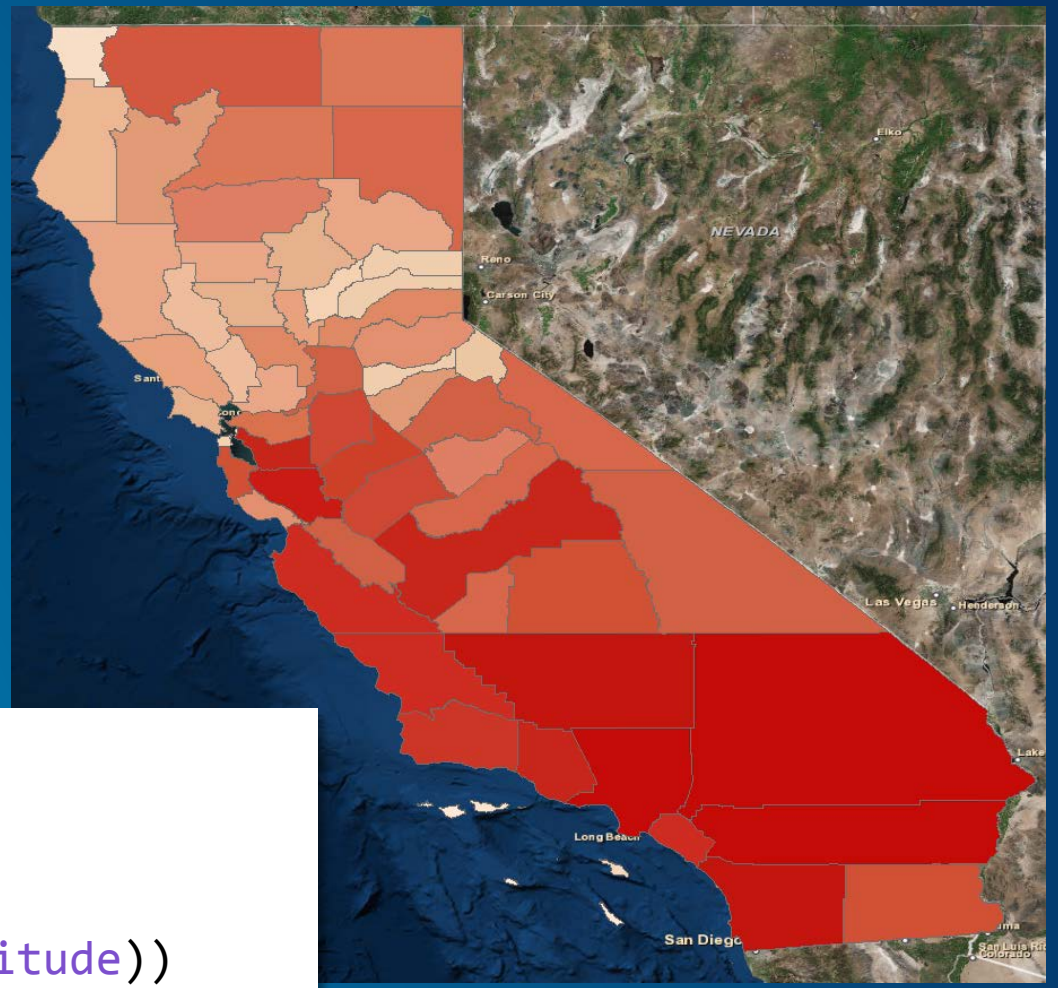
shape	count
POLYGON ((144.44 13.85, 144.54 13.85, 144.54 13.95, 144.44 13.95, 144.44 13.85))	33
POLYGON ((144.34 13.75, 144.44 13.75, 144.44 13.85, 144.34 13.85, 144.34 13.75))	27
POLYGON ((144.54 13.75, 144.64 13.75, 144.64 13.85, 144.54 13.85, 144.54 13.75))	49
POLYGON ((144.94 13.75, 145.04 13.75, 145.04 13.85, 144.94 13.85, 144.94 13.75))	39
POLYGON ((144.34 13.65, 144.44 13.65, 144.44 13.75, 144.34 13.75, 144.34 13.65))	29
POLYGON ((144.94 13.65, 145.04 13.65, 145.04 13.75, 144.94 13.75, 144.94 13.65))	75
POLYGON ((144.34 13.55, 144.44 13.55, 144.44 13.65, 144.34 13.65, 144.34 13.55))	25
POLYGON ((144.94 13.55, 145.04 13.55, 145.04 13.65, 144.94 13.65, 144.94 13.55))	71
POLYGON ((144.64 13.45, 144.74 13.45, 144.74 13.55, 144.64 13.55, 144.64 13.45))	189

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

Spatial Aggregation

With Polygons

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

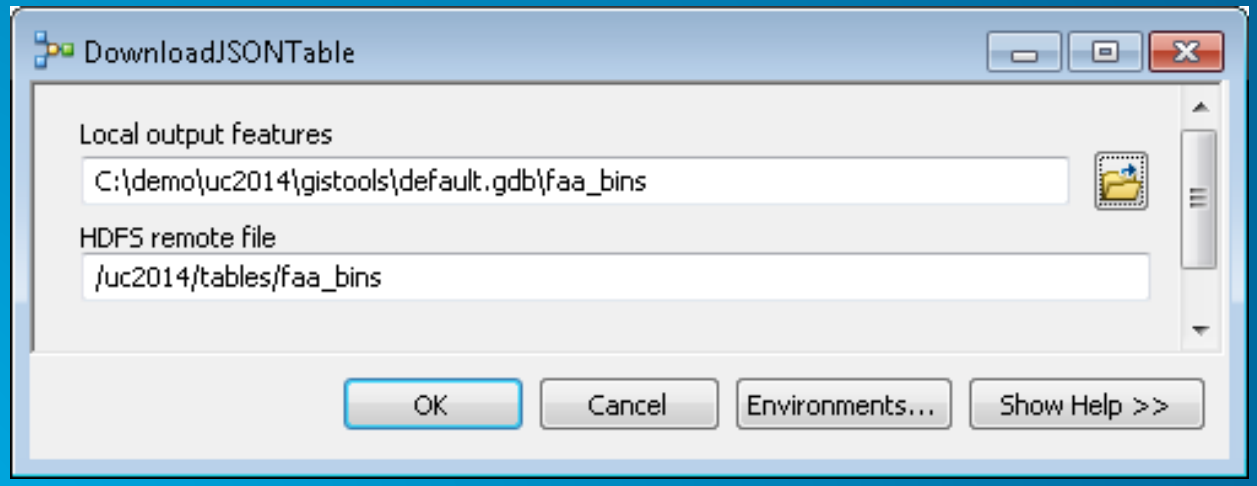


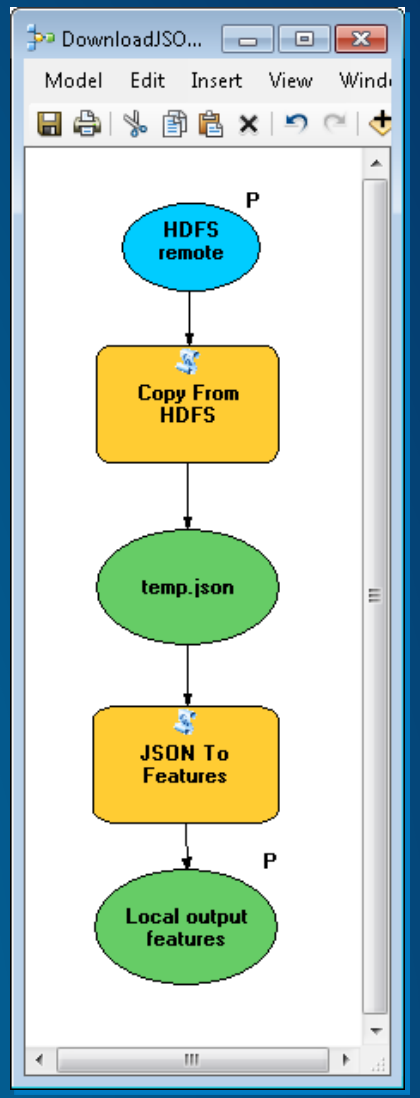

```
1 SELECT counties.shape, count(*)
2 FROM counties JOIN faa
3 WHERE ST_Contains(counties.shape,
4                   ST_Point(faa.longitude, faa.latitude))
5 GROUP BY counties.shape;
```

GIS Tools for Hadoop

Visualizing your data in ArcGIS

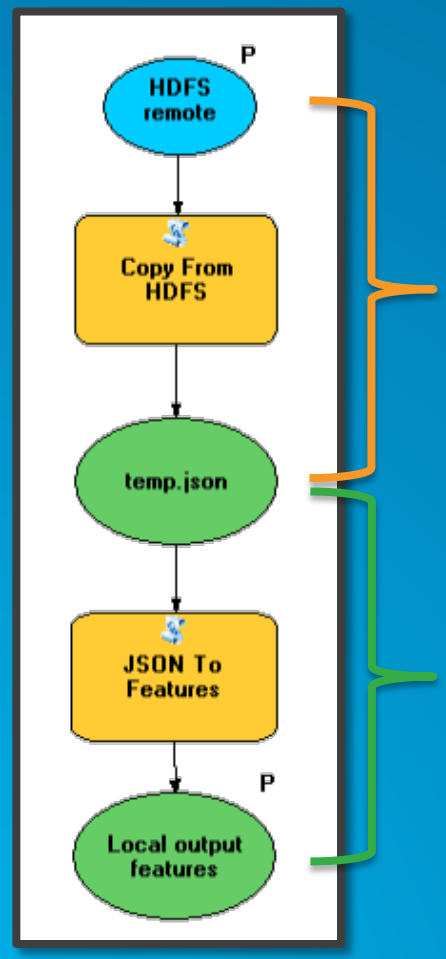
- A model was used to bring the output of our Hive queries back into ArcGIS as a local feature class
  - Models let you chain geoprocessing tools together
  - The output of one tool becomes the input to the next tool





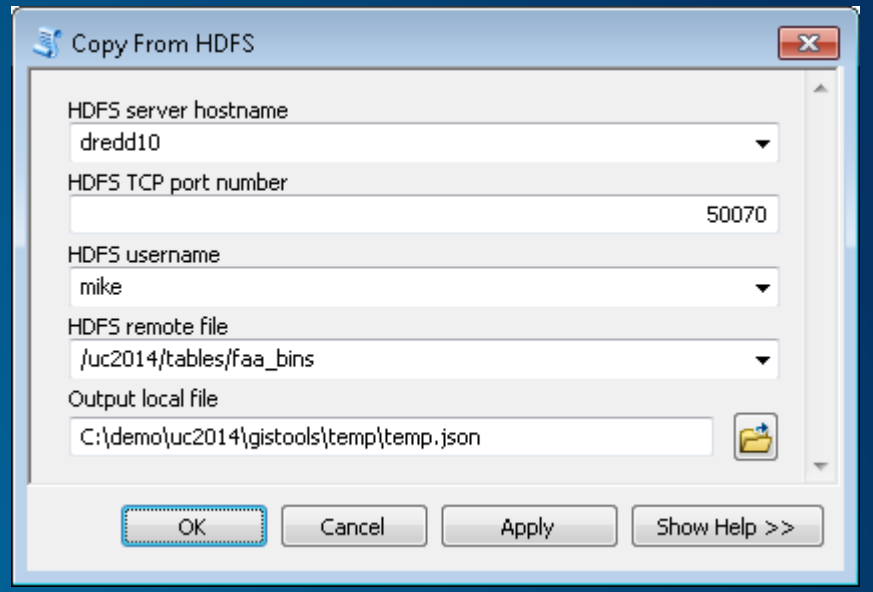
GIS Tools for Hadoop

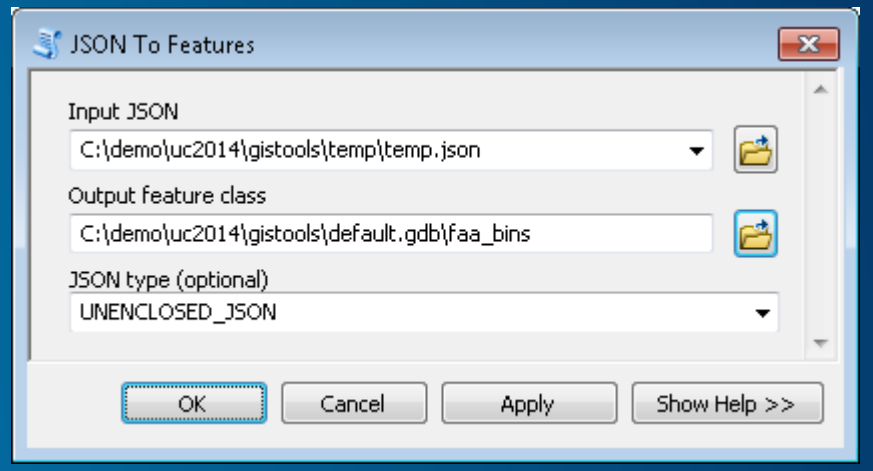
Visualizing your data in ArcGIS



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

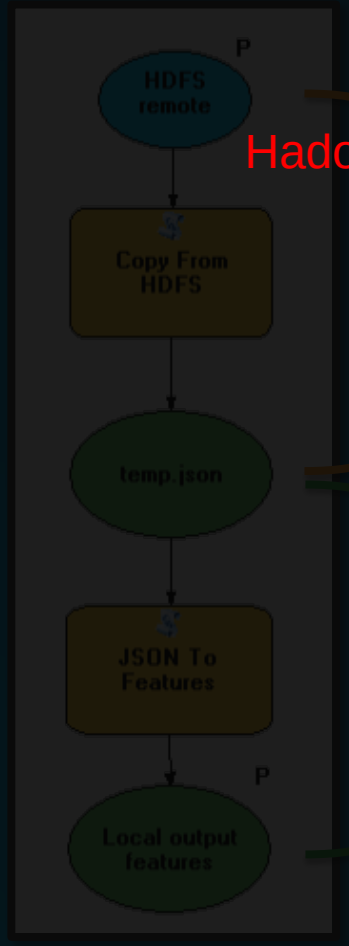
JSON To Features – Create a feature class from a JSON file





GIS Tools for Hadoop

Visualizing your data in ArcGIS



Hadoop service information

Hadoop path to the result of the Hive aggregation query

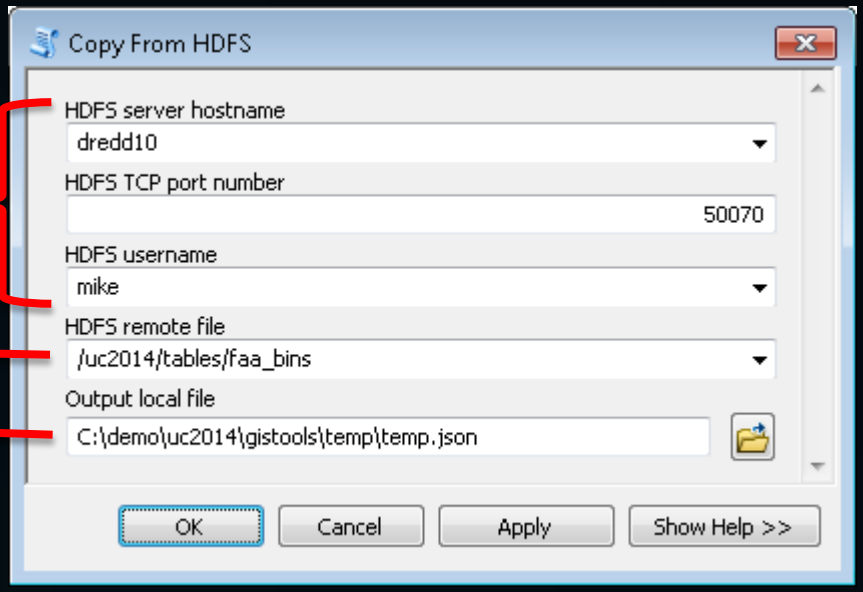
Copy From HDFS – Copy a file or directory from Hadoop to a local file

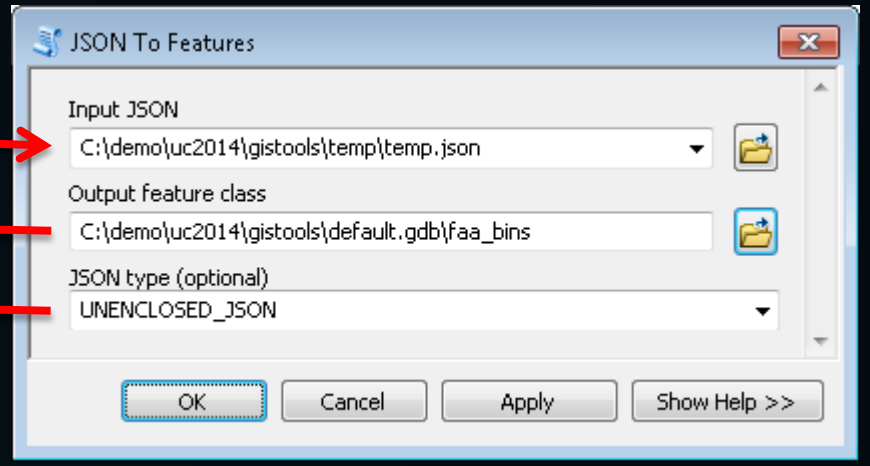
Path to temporary file on local hard disk

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

Output feature class to create

JSON file type (usually unenclosed)



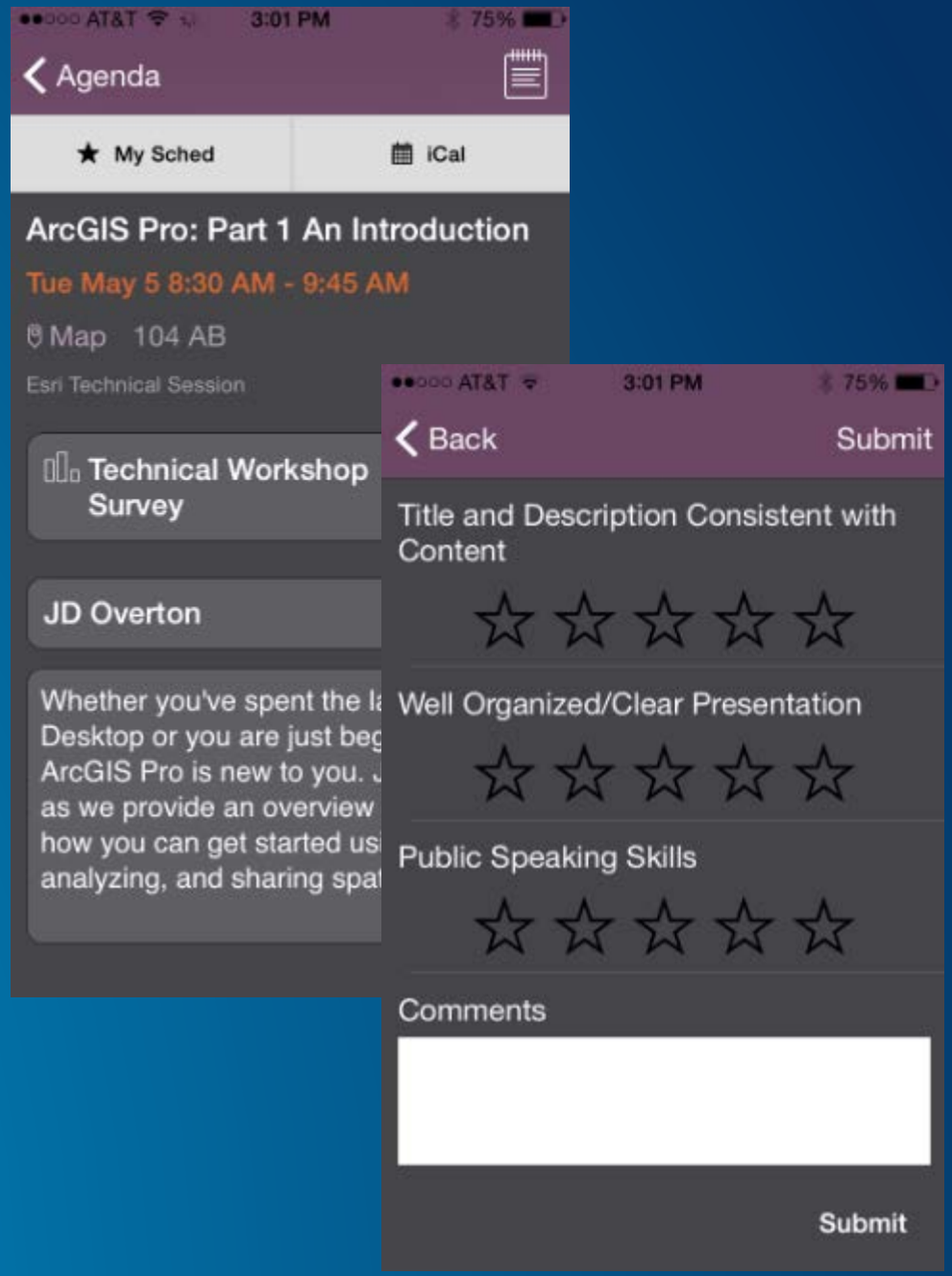


Other relevant sessions

- 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



Understanding our world.