Geospatial Processing on Hadoop

Hadoop World 2013

Erich Hochmuth, R&D IT Platform Engineering Lead
erich.hochmuth@monsanto.com

Eric Turcotte, Big Data Engineer
eric.d.turcotte@monsanto.com / @ericturcotte

MONSANTO

Agenda

- About Monsanto
- Use Case Overview
- Intro to Geospatial Data Types
- Geospatial on Hadoop
- Q&A

Our Vision: Sustainable Agriculture

Providing Choices for Farmers, Meeting Society's Needs

- **Producing more**
  - We are committed to increasing yields to meet the growing demand for food, fiber & fuel
- **Conserving more**
  - We are committed to reducing the amount of land, water and energy needed to grow our crops
- **Improving lives**
  - We are committed to improving lives around the world

This is sustainable agriculture and it's what we do

Our Approach to Driving Yield

A System of Agriculture Working Together to Boost Productivity

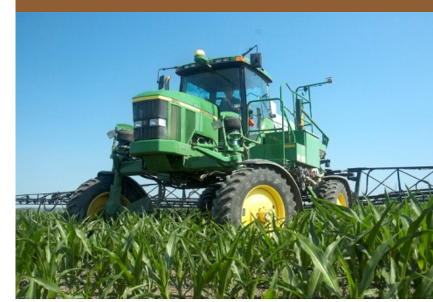
BREEDING

The art and science of combining genetic material to produce a new seed

BIOTECHNOLOGY

The science of improving plants by inserting genes into their DNA

AGRONOMICS



The farm management practices involved in growing plants

Doubling Yields by 2030 - Farming in the Future Will Be Increasingly Information-Driven

- ❑ ADVANCED EQUIPMENT
- ❑ AVERAGE CORN YIELD
  - -300 BU/AC
- ❑ AUTOMATED WEATHER STATIONS
- ❑ FIELD SENSORS PROVIDING INFORMATION
- ❑ ADVANCED IMAGERY TECHNOLOGY

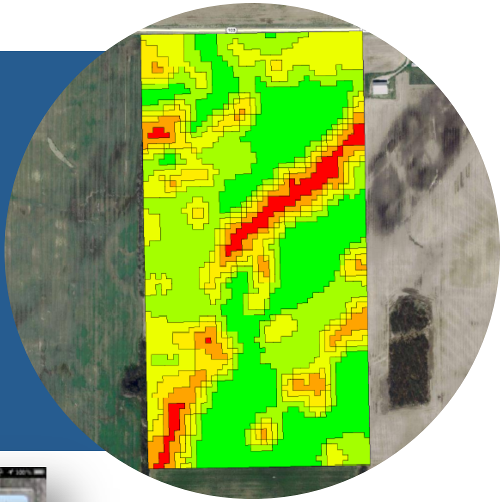
Integrated Farming Systems – FieldScripts℠ for 2014

- FieldScripts℠ will deliver, by field, a corn hybrid recommendation utilizing variable rate seeding by FieldScripts management zones to increase yield potential and reduce risk
- The science of FieldScripts is based on proprietary algorithms that combine data from the FieldScripts Testing Network and Monsanto generated hybrid response to plant population research

Planting Prescription 2012
(DKC63-84 Brand)

Target Rate (Count)
(ksds/ac)

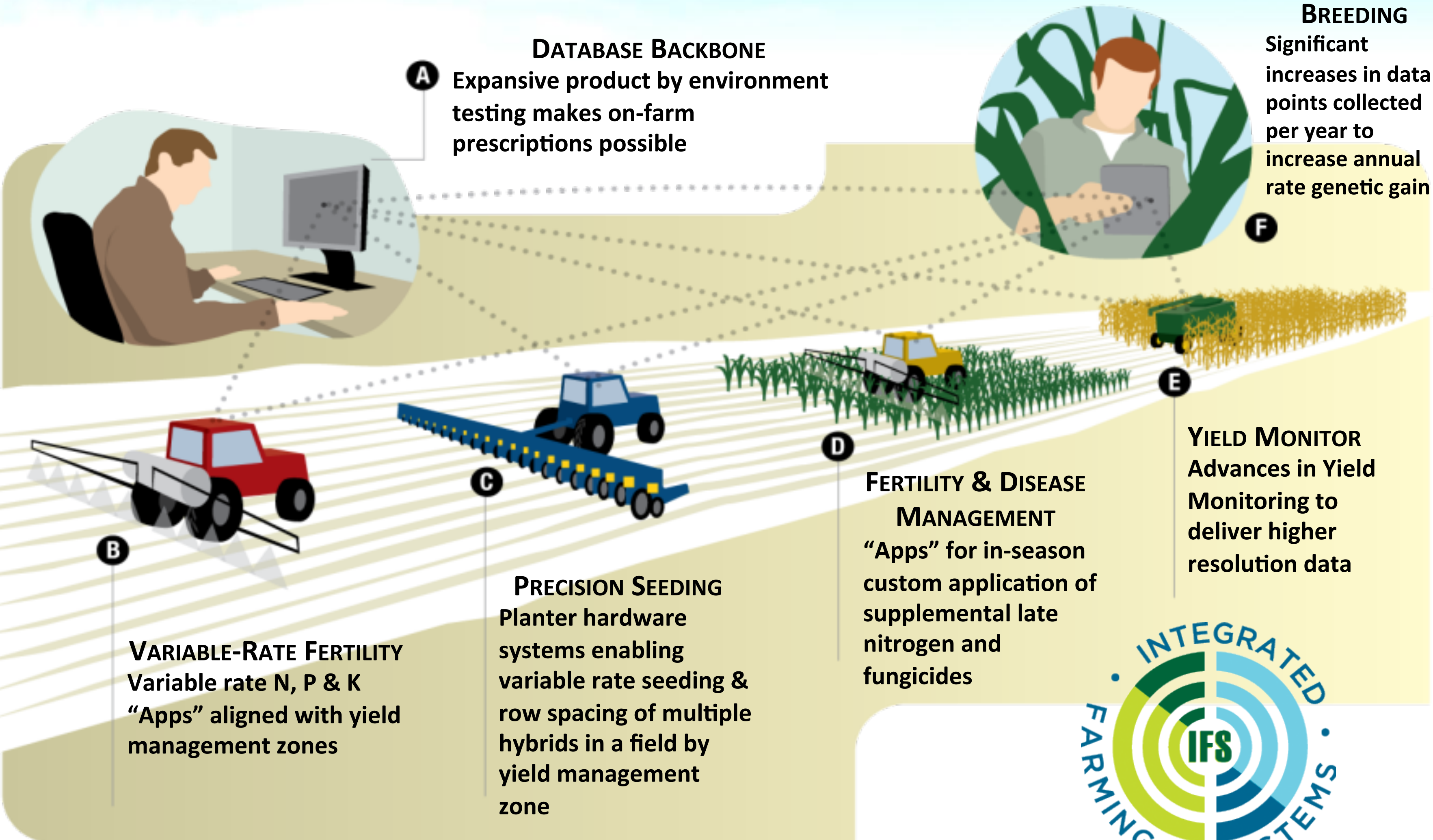
- 38.00 (24.75 ac)
- 37.00 (22.63 ac)
- 35.00 (16.60 ac)
- 34.00 (8.23 ac)
- 33.00 (6.00 ac)
- 32.00 (2.82 ac)



Precision Planting

FieldScripts℠

Integrated Farming Systems℠ Combine Advanced Seed Genetics, On-farm Agronomic Practices, Software and Hardware Innovations to Drive Yield

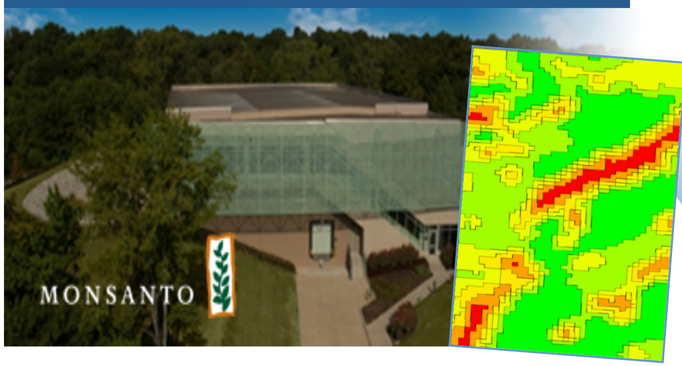


INTEGRATED FARMING SYSTEMS℠ IFS

GETTING MORE OUT OF EVERY ACRE

Harvesting & Returning FieldScripts Data

IFS Analytics Platform



- Analyze Data
- Generate FieldScripts

Wireless

Farmer – Combine Cab

- Return Harvest Data to Dealer/MON

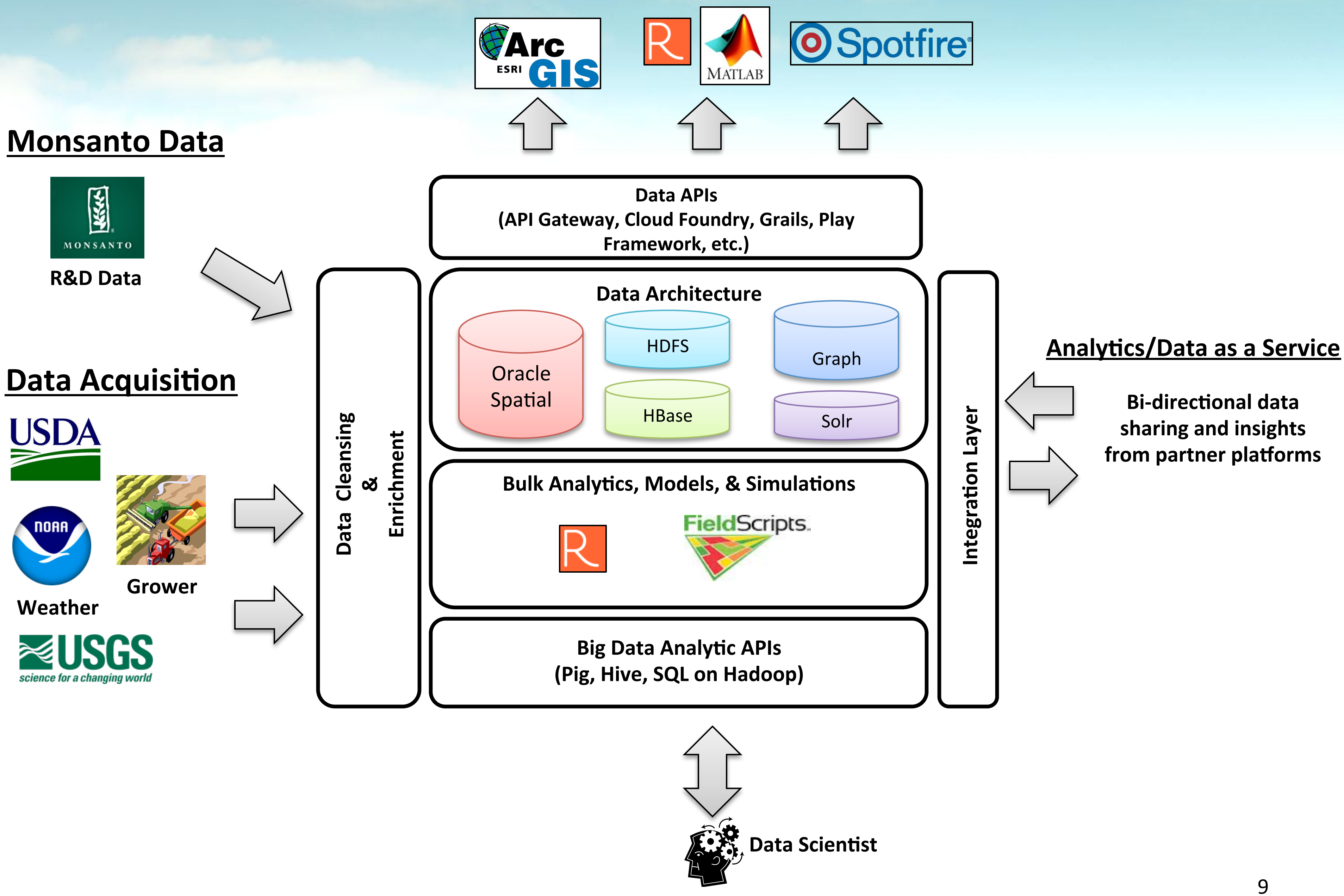
Data

Seed Dealers

Data

- Harvest Data Collection
- Review Yield Results with Farmer

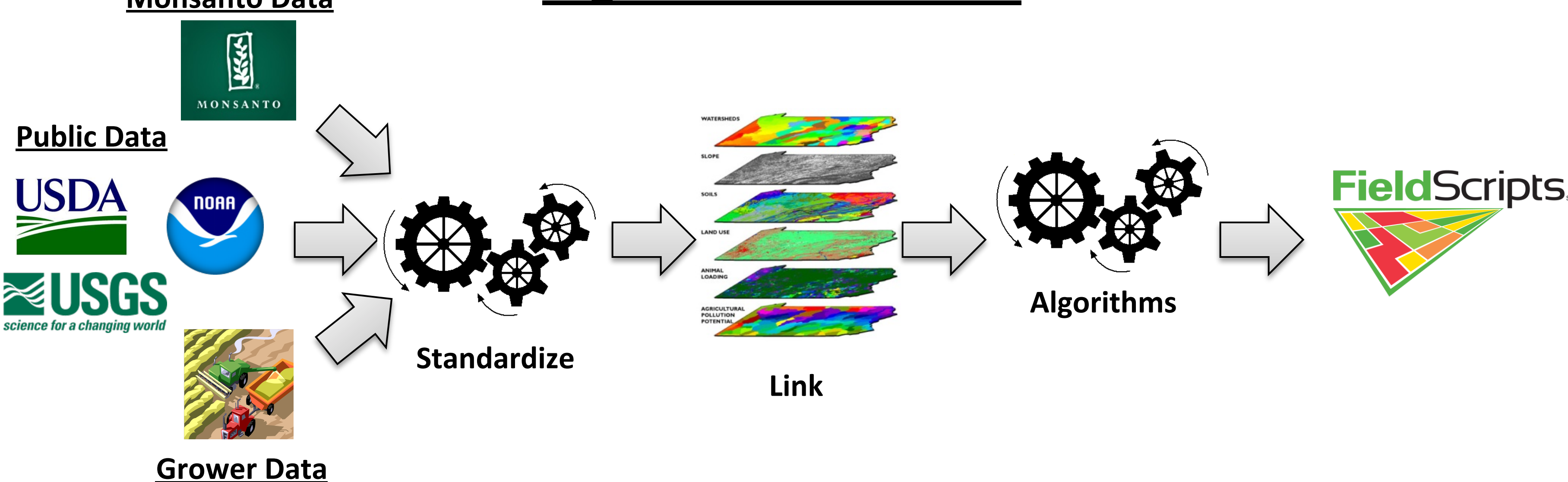
IFS Analytics Ecosystem Platform



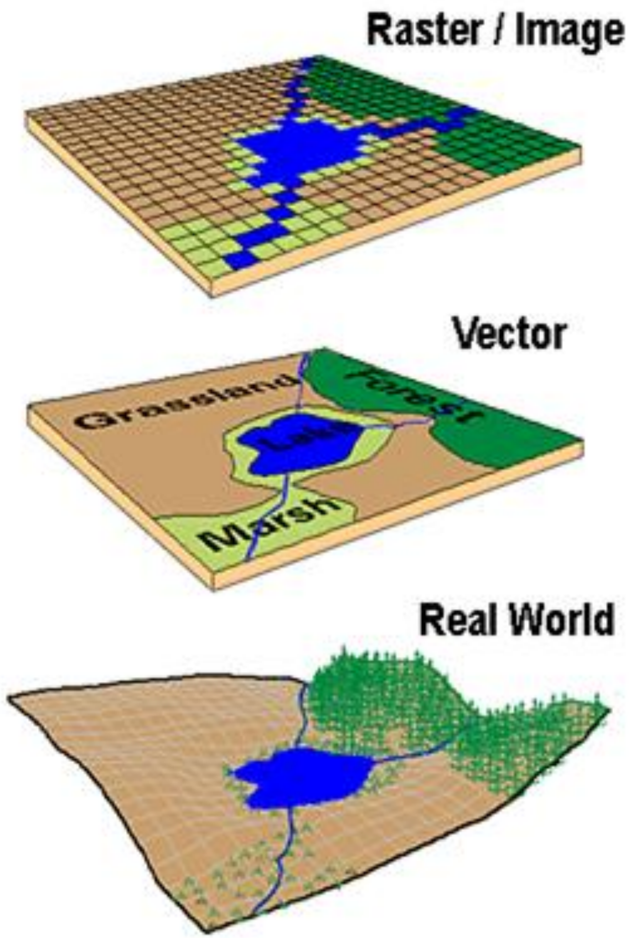
Platform Needs

- Load thousands of files containing spatial data
  - 10s TBs of data
- Support diverse range of data types
  - Tabular, xml, vector, raster
- Join & link data spatially
- Make data available for data products such as Field Scripts
- Make data available for data scientists

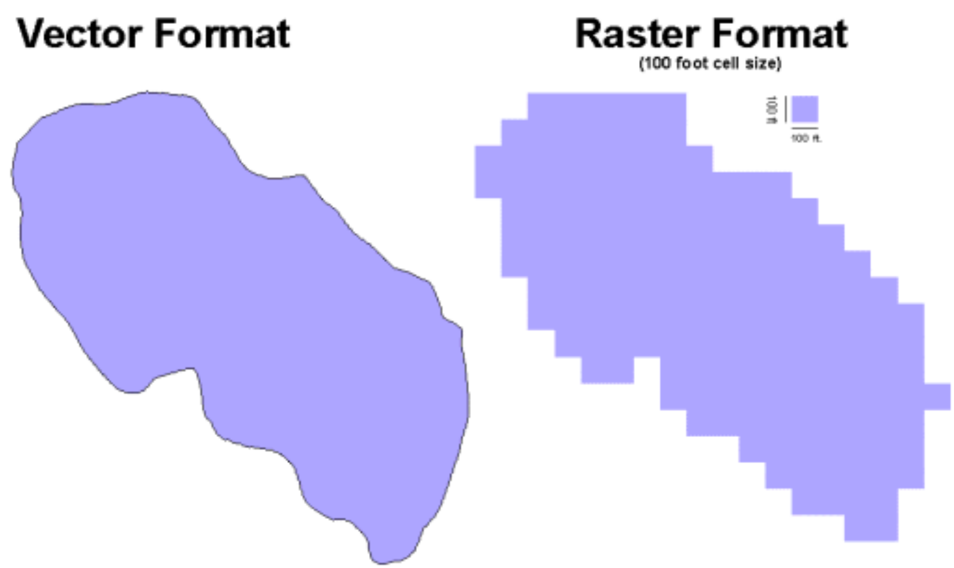
High Level Data Flow



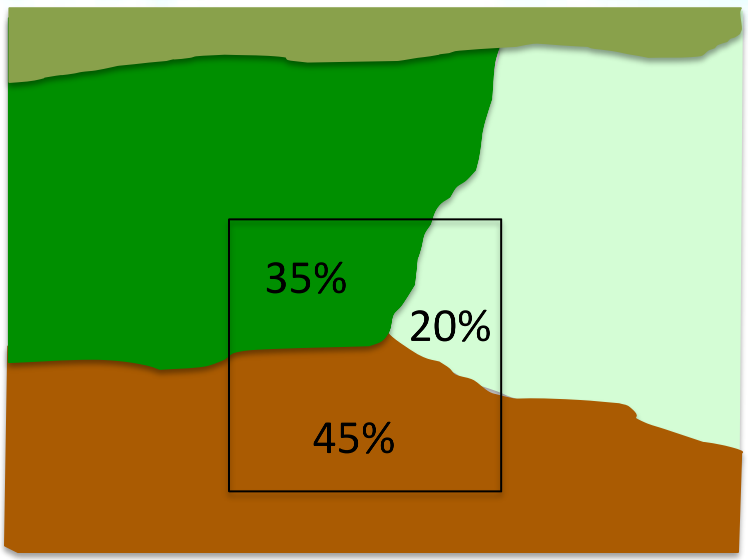
Spatial Data Types

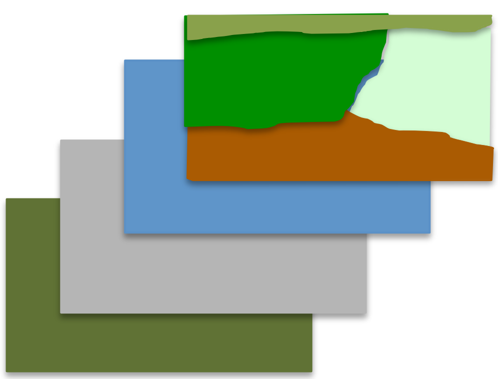


- Vector - geometries, with data
  - Point, polygon, line, circle, etc.
  - Compact even for large geographic areas
  - Increases size with level of detail
  - Resolution independent
  - Formats: Shape file, WKT, CAD
- Raster - pixels, with data
  - Fixed level of detail
  - Resolution dependent
  - Formats: GeoTiff, BMP



Spatial Data Types





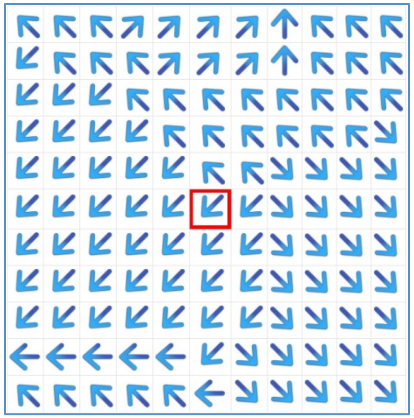
- Querying vector data
  - Load and find geometries near the target area
  - Do the geometries intersect with the target area?
  - How much do they cover the target area?
  - Is there overlap?
  - Spatially weigh values
- Considerations
  - Diverse range of data types
  - Multiple layers
  - Numerous features per layer
  - Overnumerous target areas
  - Spatial index maintenance

Data in Detail

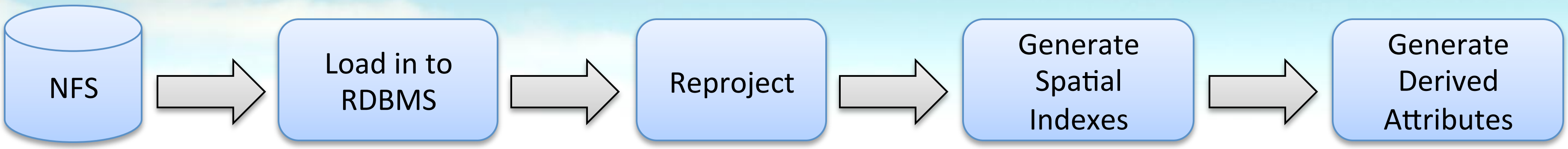
- Environmental Observations
  - Elevation, Soil, Grower Data
- Extremely Dense Structured Data
  - 10 m x 10 m grids
- Even More Dense Raw Data
  - Raw data captured at 5 Hz
- Complex Data Interactions
- Derived Data
  - Slope, Aspect, etc.
- Access Patterns
  - Bulk analytics & random reads

Example Yield Map

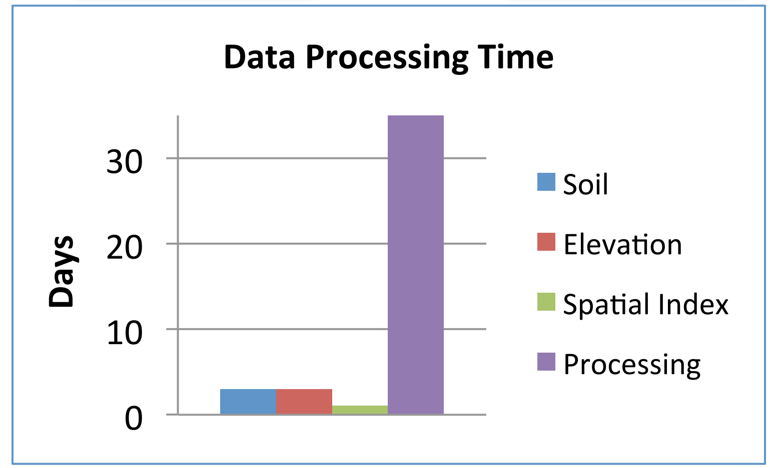
Computed Flow Direction

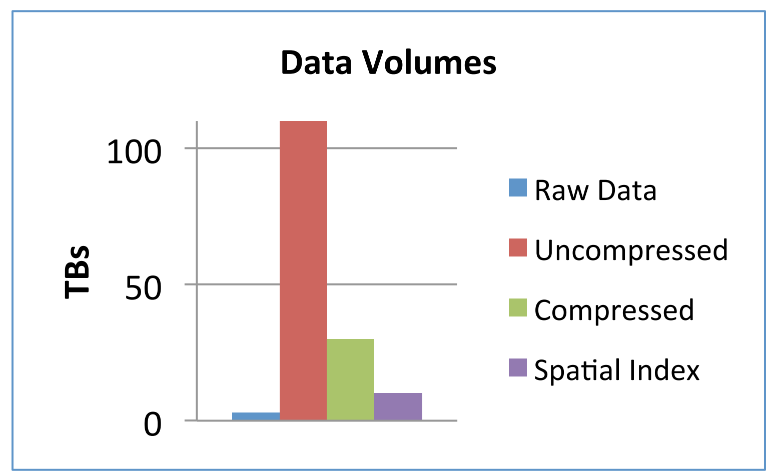


Take 1: RDBMS - Data Ingestion & Data Processing



- In RDBMS spatial
- PL/SQL
- Just 8% of the data!!
  - 35+ days to load
- TBs in indexes
- Multiple Patches to DB
- Tradeoffs
  - Compressed vs. Uncompressed
  - Performance vs. Storage
  - Read vs. Write performance
- Options/recommendations
  - Limit use of in DB spatial functionality
  - Buy larger RDBMS





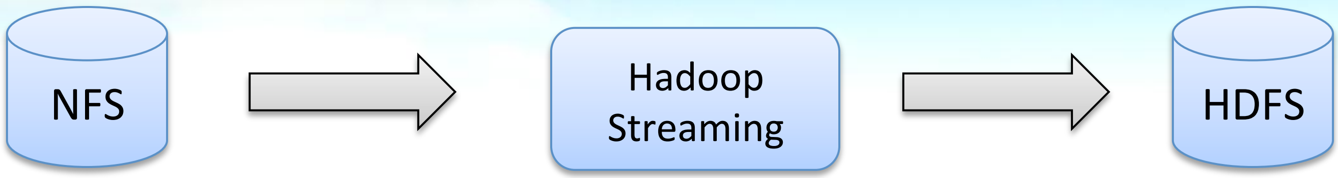
A Different Approach

Requirements

- Scalable
- Complex geospatial data types
- Push analytics/data process to the storage
- Commodity/cost effective storage and compute
- Vendor support

Alternative Considered

Data Ingestion Revisited



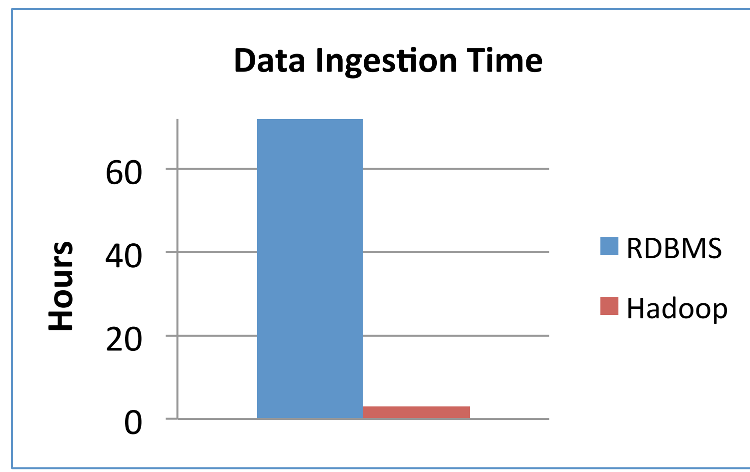
- Raster Images
- Vector Shape Files
- Zip Files
- Text Data

- Unzip
- Convert to Raster
- Re-project

- GeoTiff

- Bulk load 1,000s of files into HDFS
- Standardize data
  - Common usable format
    - Storage vs. Compute
    - Raster format is easily splitable
- Hadoop Streaming integrated with GDAL
- Streaming API Lessons Learned
  - Lack of documentation
  - Counters to track task progress
  - Jobs run as mapred user
  - HDFS access outside of MR

<u>Results</u>

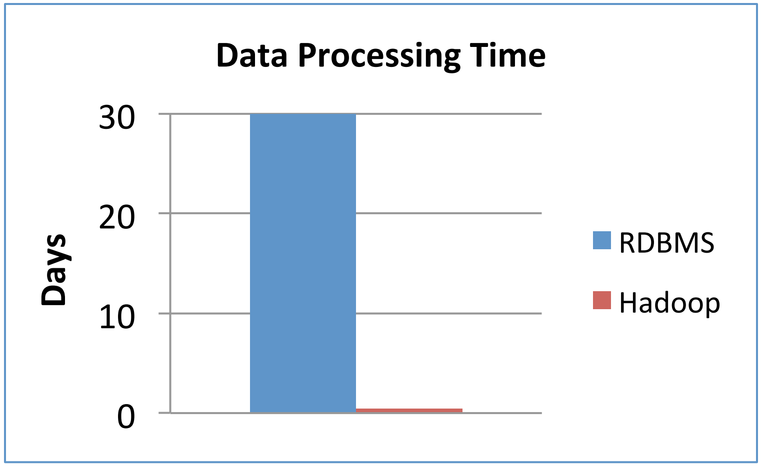


Data Processing Revisited

- Raster Files

- Process raster data
  - Dense matrix
- Generic InputFormat & RecordReader for raster data
- HFiles easily transportable between clusters
- Challenges tuning Jobs
  - I/O Sort Factor
  - Split/Task Size

Results

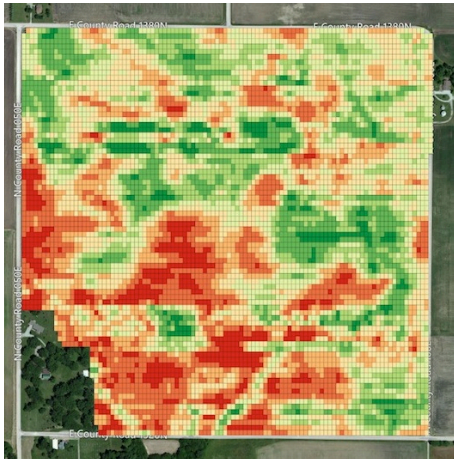


Geospatial in HBase

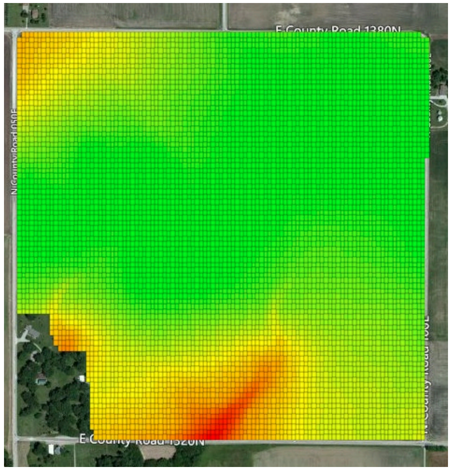
- Needs
  - Spatially enabled HBase key
  - Reduce/eliminate need for index tables
  - Scalable & cost efficient
  - Support targeted & bulk random reads
  - Optimize I/O for reads

Example Data Layers

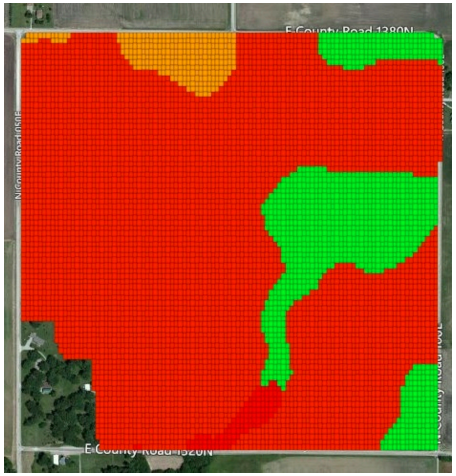
Yield



Elevation



Soil



Geospatial in HBase

- Needs
  - Spatially enabled HBase key
  - Reduce/eliminate need for index tables
  - Scalable & cost efficient
  - Support targeted & bulk random reads
  - Optimize I/O for reads
- Considerations
  - Key overhead
  - Scan vs. Get performance
  - Reduce reading unnecessary data
  - Data within a field only!

Filter data to what overlaps with the Field Boundary

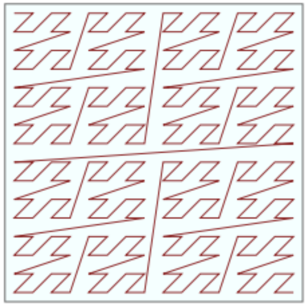
Data Stored with Each Value

	Key Length	Value Length	Row Length	Row	Column Family Length	Column Family	Column Qualifier	Time stamp	Key Type	Value
Data Type	Integer	Integer	Short	byte[]	Byte	byte[]	byte[]	long	Byte	byte[]
Size (Bytes)	4	4	2	Row Bytes	1	CF Bytes	CQ Bytes	8	1	Value Bytes

Geospatial in HBase

- Needs
  - Spatially enabled HBase key
  - Reduce/eliminate need for index tables
  - Scalable & cost efficient
  - Support targeted & bulk random reads
  - Optimize I/O for reads
- Considerations
  - Key overhead
  - Scan vs. Get performance
  - Reduce reading unnecessary data
- Options
  - GeoHash most notable example
    - Best suited for sparse data
  - MGRS (Military grid reference system)
    - Boundary edge cases
  - Quad tree & R-trees

Geohash Z-order-curve



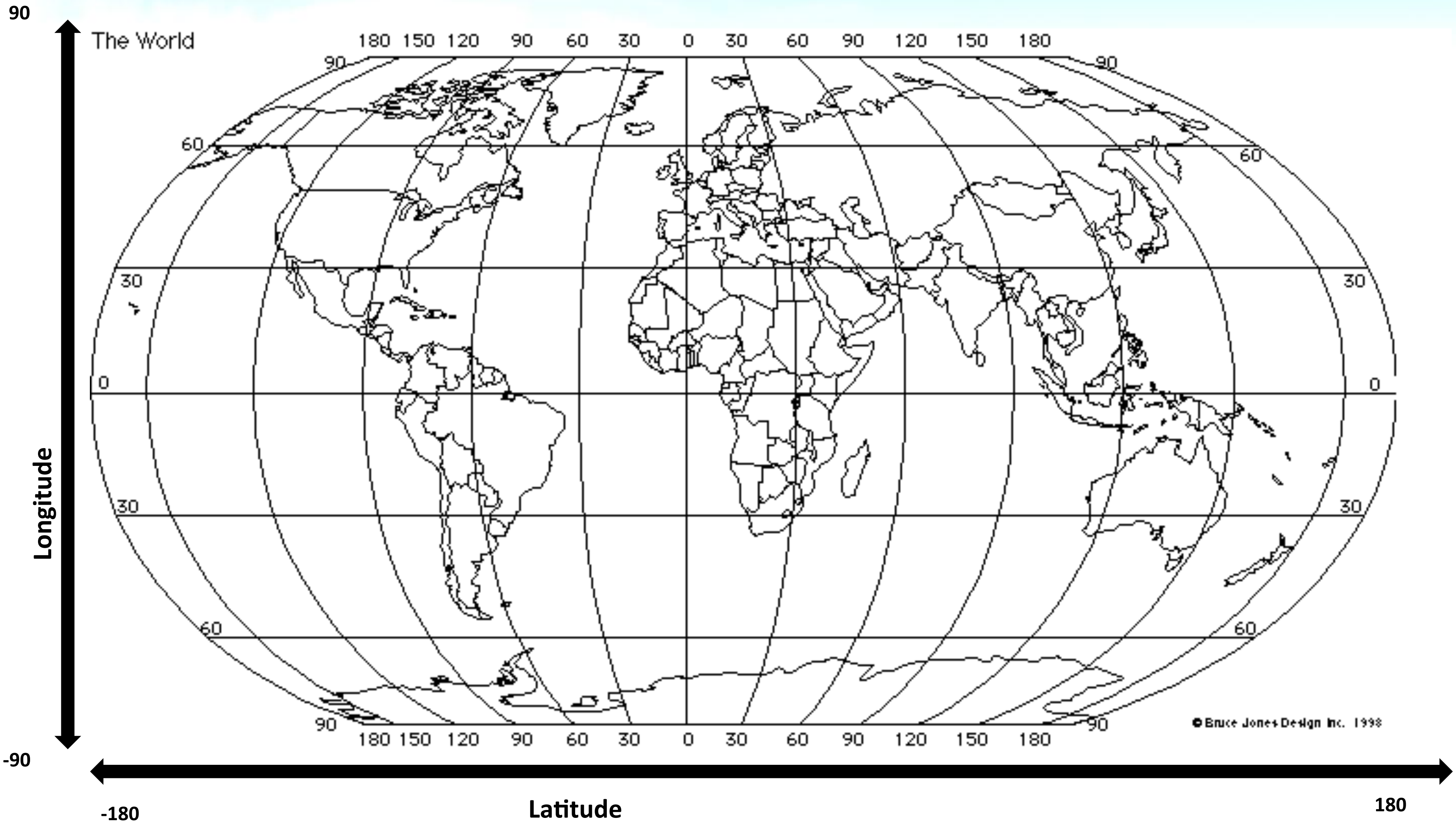
Alphanumeric Keys

4QFJ123678 ....precision level 100 m
4QFJ12346789 ....precision level 10 m
4QFJ1234567890 ....precision level 1 m

Data Stored with Each Value

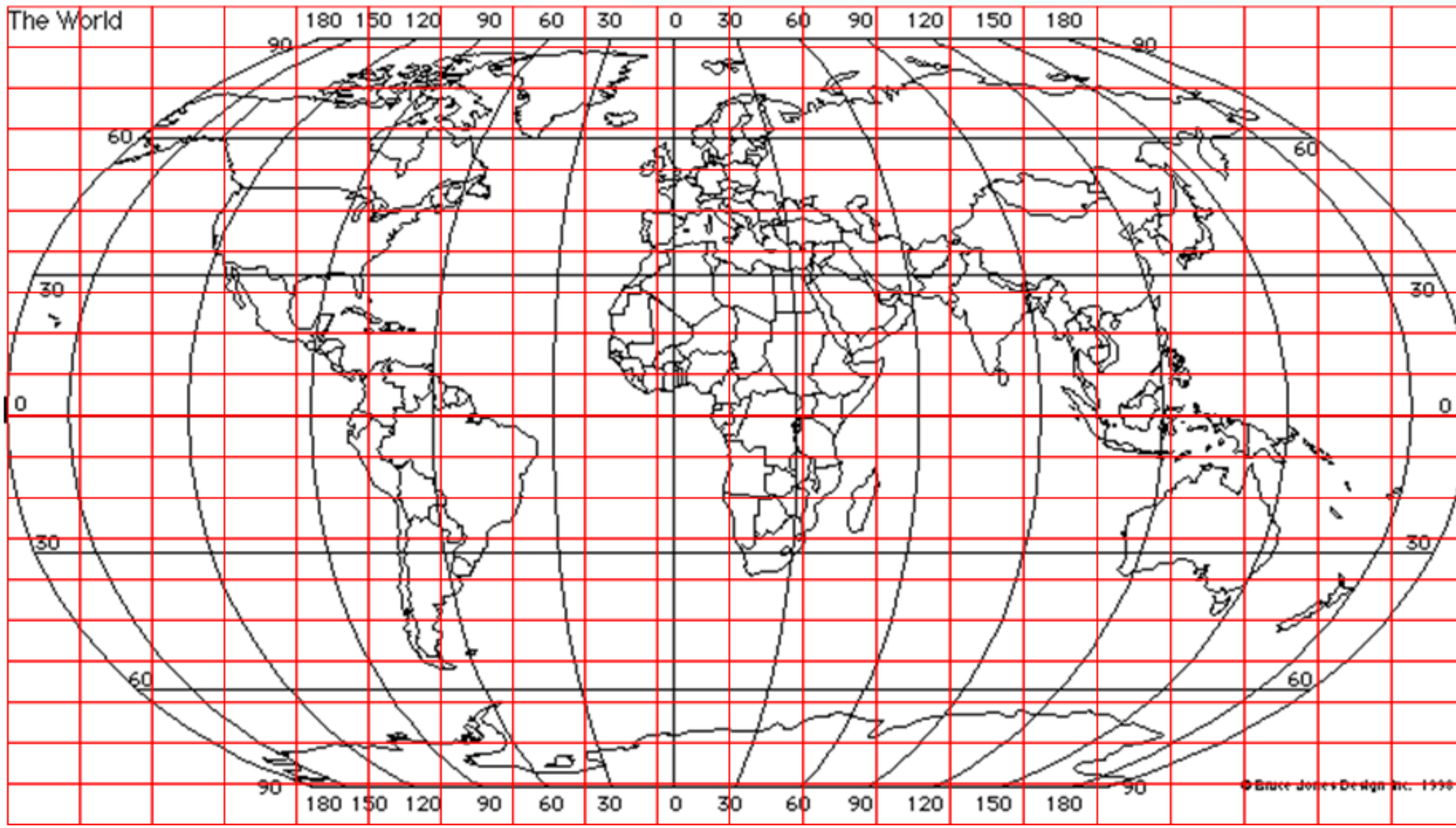
	Key Length	Value Length	Row Length	Row	Column Family Length	Column Family	Column Qualifier	Time stamp	Key Type	Value
Data Type	Integer	Integer	Short	byte[]	Byte	byte[]	byte[]	long	Byte	byte[]
Size (Bytes)	4	4	2	Row Bytes	1	CF Bytes	CQ Bytes	8	1	Value Bytes

Global Coordinate System



Reference System

90



-90

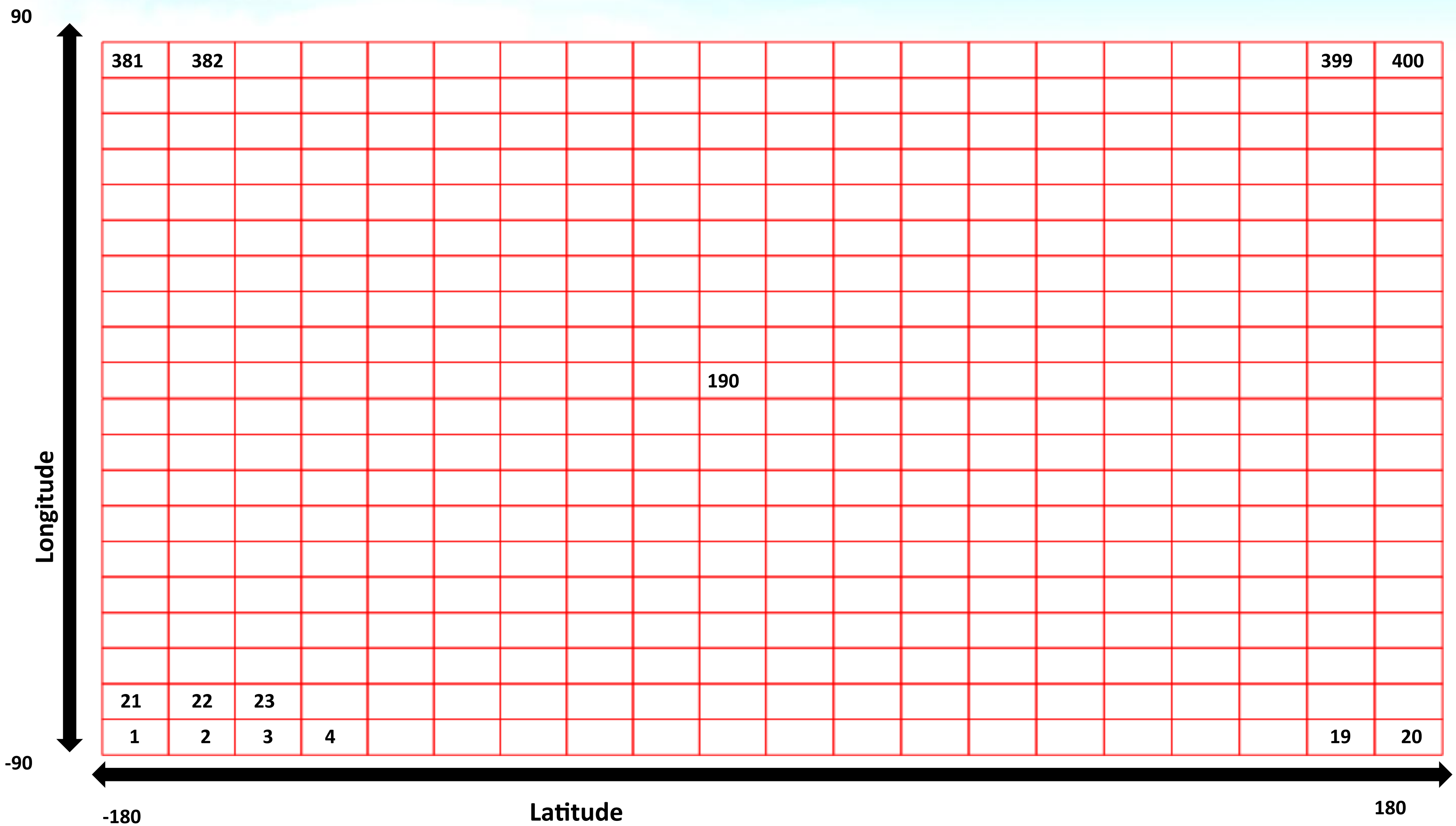
Longitude

Latitude

-180

180

Reference System Continued



HBase Schema Take 1

Spatial Table

- Key: cell_id long
- Column Family: A
  - Column: Data Holder
    - elevation: float
    - slope: float
- Each spatial dataset is a separate table
- All attributes for a layer that are read together are stored together
  - *Attributes packed into a single column as an Avro object*
- 1 row per record
- 120 billion rows total!
- 1,000s of Get requests per field
- TBs of key overhead – roughly 56% of the data

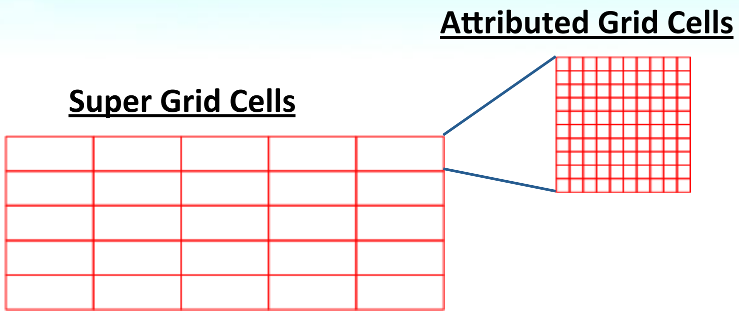
Data Stored with Each Value

	Key Length	Value Length	Row Length	Row	Column Family Length	Column Family	Column Qualifier	Time stamp	Key Type	Value
Data Type	Integer	Integer	Short	byte[]	Byte	byte[]	byte[]	long	Byte	byte[]
Size (Bytes)	4	4	2	Row Bytes	1	CF Bytes	CQ Bytes	8	1	Value Bytes

HBase Schema Take 2

Spatial Table

- Key: super_cell_id long
- Column Family: A
  - Column: Data Holder
    - elevation : array float [values]
    - slope: array float [values]
    - aspect: array float [values]



- Data grouped into 100 x 100 super cells
- A super cell of 100 x 100 cells is a single row in HBase
- At most 4 disk reads are required to read all data for one layer for a 150 acre field
- Given a bounding box the super cells and attributed grid cells containing the desired data can easily be computed
- A generic geospatial data service when given a set of layers will read each layer in parallel
- Overhead of key data reduced from 56% to below 0.1%

Results

- Significant cost savings in required hardware
- 120 billion unique polygons in total
- 1.5 trillion data points
- Dense grid of the entire U.S.
- Foundational architecture for other spatial data sets
- Fully unit tested implementation

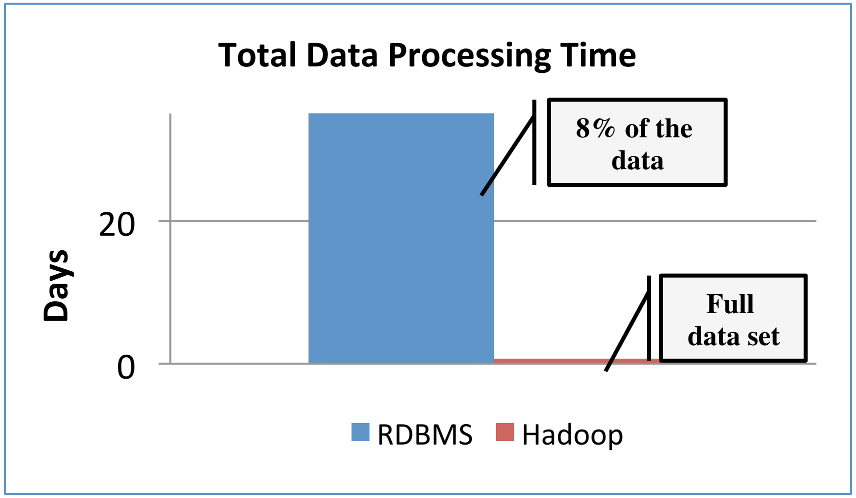
Hadoop

- Entire U.S.
- 18 hour load time
- 3 months of development
- 100% scalable

RDBMS

- 4 states only
- 30+ days to load
- 8 months of development

Total Run Time



Future Considerations

- HBase Filters
  - Push spatial functionality into RegionServer
- Pre-computed aggregates at different resolutions
- Distributed Vector Data Store
  - Solr/Lucene via Cloudera Search
- Spatial UDFs
  - Pig, Hive/Impala
- Metadata repository & data lineage

Thank You

Yes, we are hiring.

erich.hochmuth@monsanto.com

eric.d.turcotte@monsanto.com

Big Data Engineer - http://bit.ly/16Iuojt

Geospatial Analytics Scientist - http://bit.ly/16iZgtM

Discovery Engineer - http://bit.ly/1byFLNQ

Technical Architect - http://bit.ly/1gXQp7T

More…