cloudera
Ask Bigger Questions

Building Applications on Hadoop

Mark Grover

Software Engineer, Cloudera

@mark_grover

Jfokus 2014 (February 4th, 2014)

Agenda

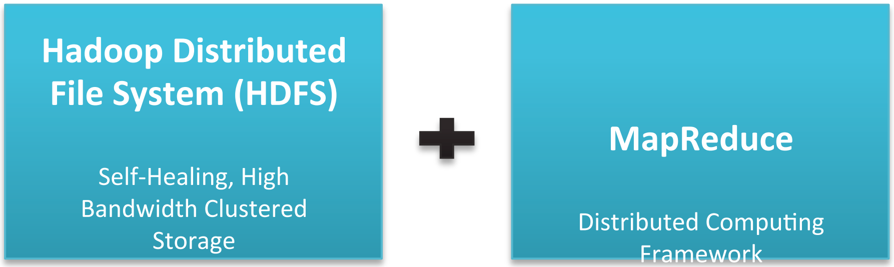
- Brief intro to Hadoop and the ecosystem
- Developing apps on Hadoop
    - What's the current problem?
    - How are we fixing it?

What is Apache Hadoop?

Apache Hadoop is an open source platform for data storage and processing that is…

- ✓ Scalable
- ✓ Fault tolerant
- ✓ Distributed

CORE HADOOP SYSTEM COMPONENTS



Hadoop Distributed File System (HDFS)

Self-Healing, High Bandwidth Clustered Storage

+

MapReduce

Distributed Computing Framework

Has the Flexibility to Store and Mine Any Type of Data

- Ask questions across structured and unstructured data that were previously impossible to ask or solve
- Not bound by a single schema

Excels at Processing Complex Data

- Scale-out architecture divides workloads across multiple nodes
- Flexible file system eliminates ETL bottlenecks

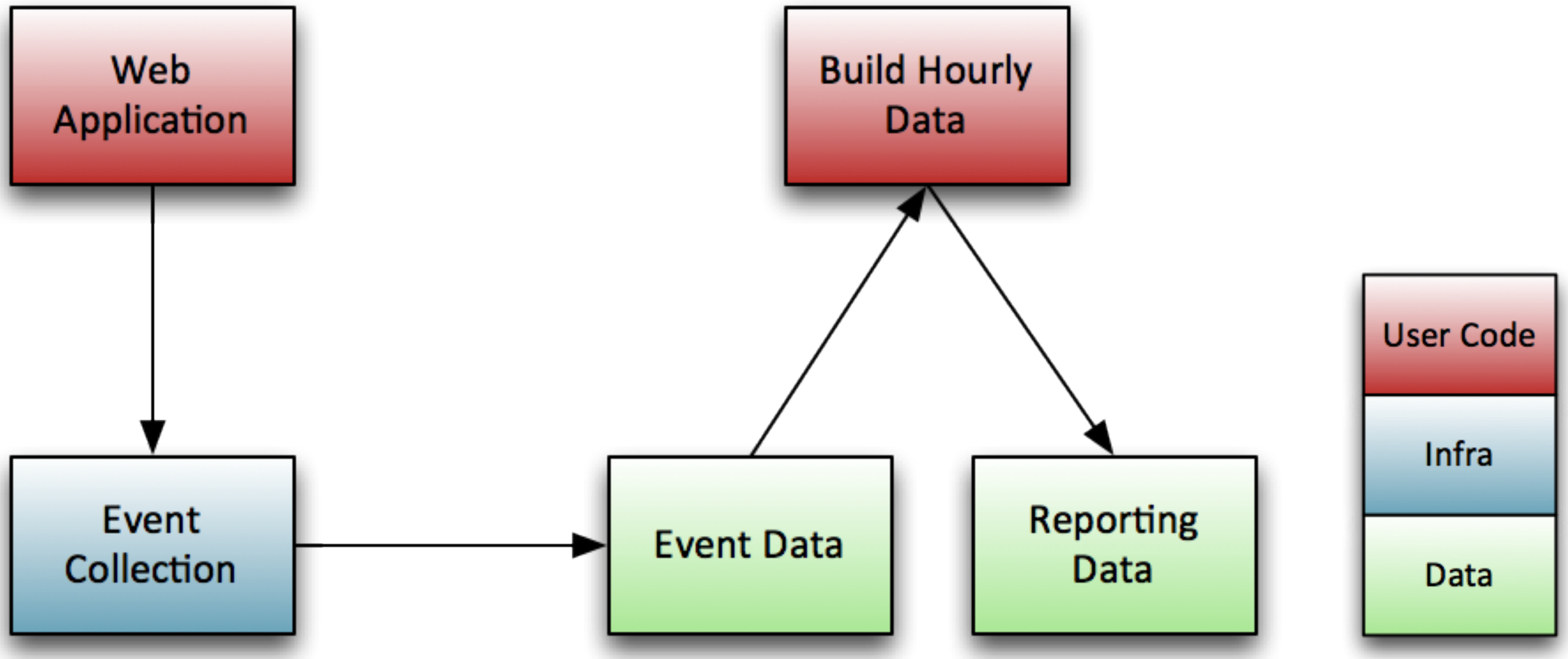
Scales Economically

- Can be deployed on commodity hardware
- Open source platform guards against vendor lock

Developing apps on Hadoop

Kite SDK

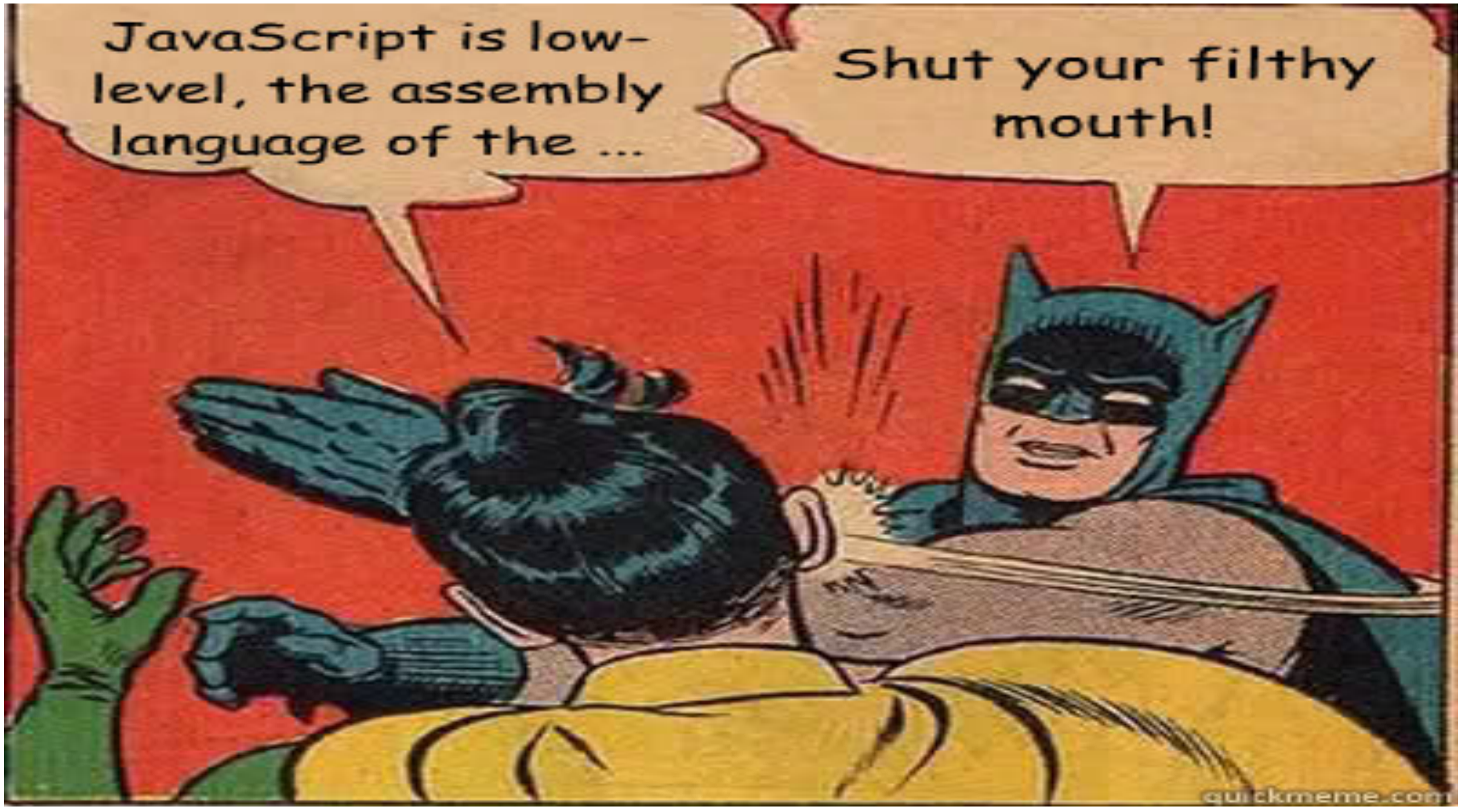
A typical system (zoom 100:1)



Hadoop is incredibly powerful

Hadoop is incredibly flexible

Hadoop is incredibly low-level

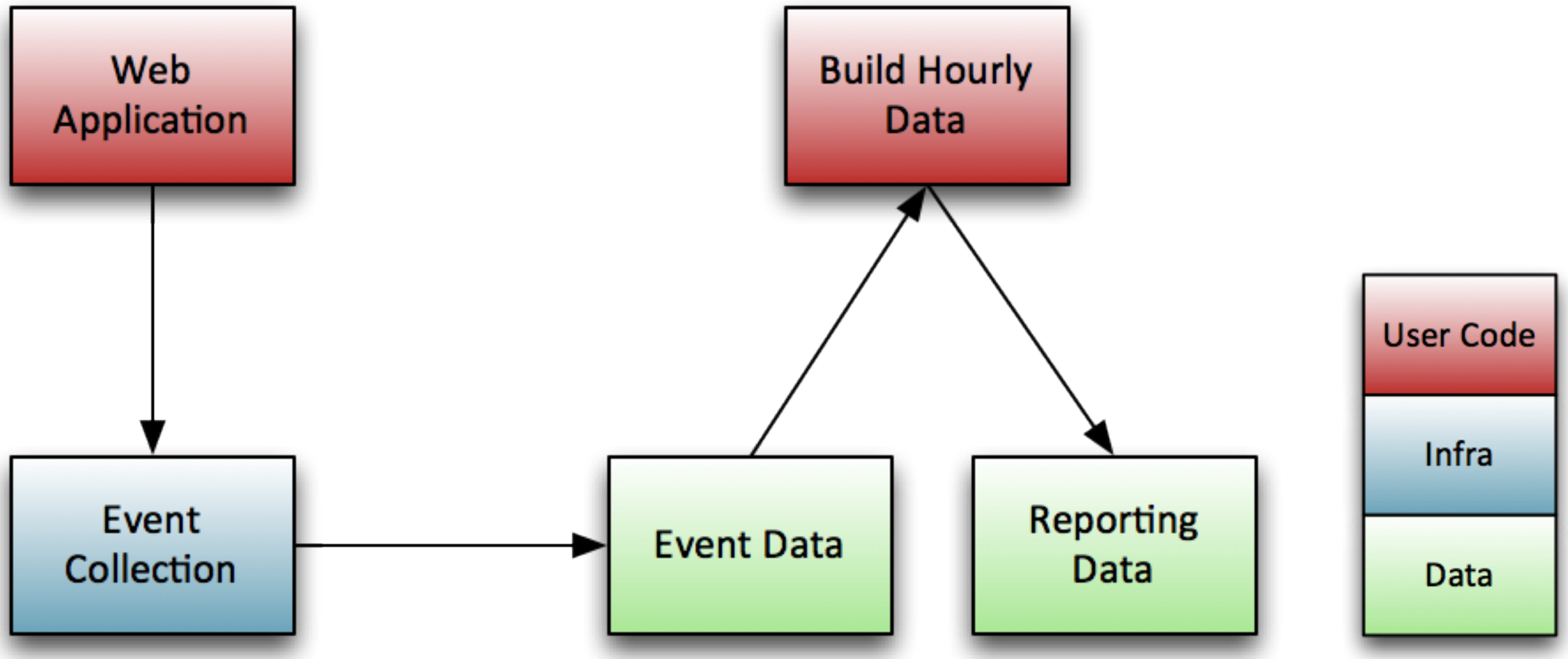


Hadoop is incredibly complex

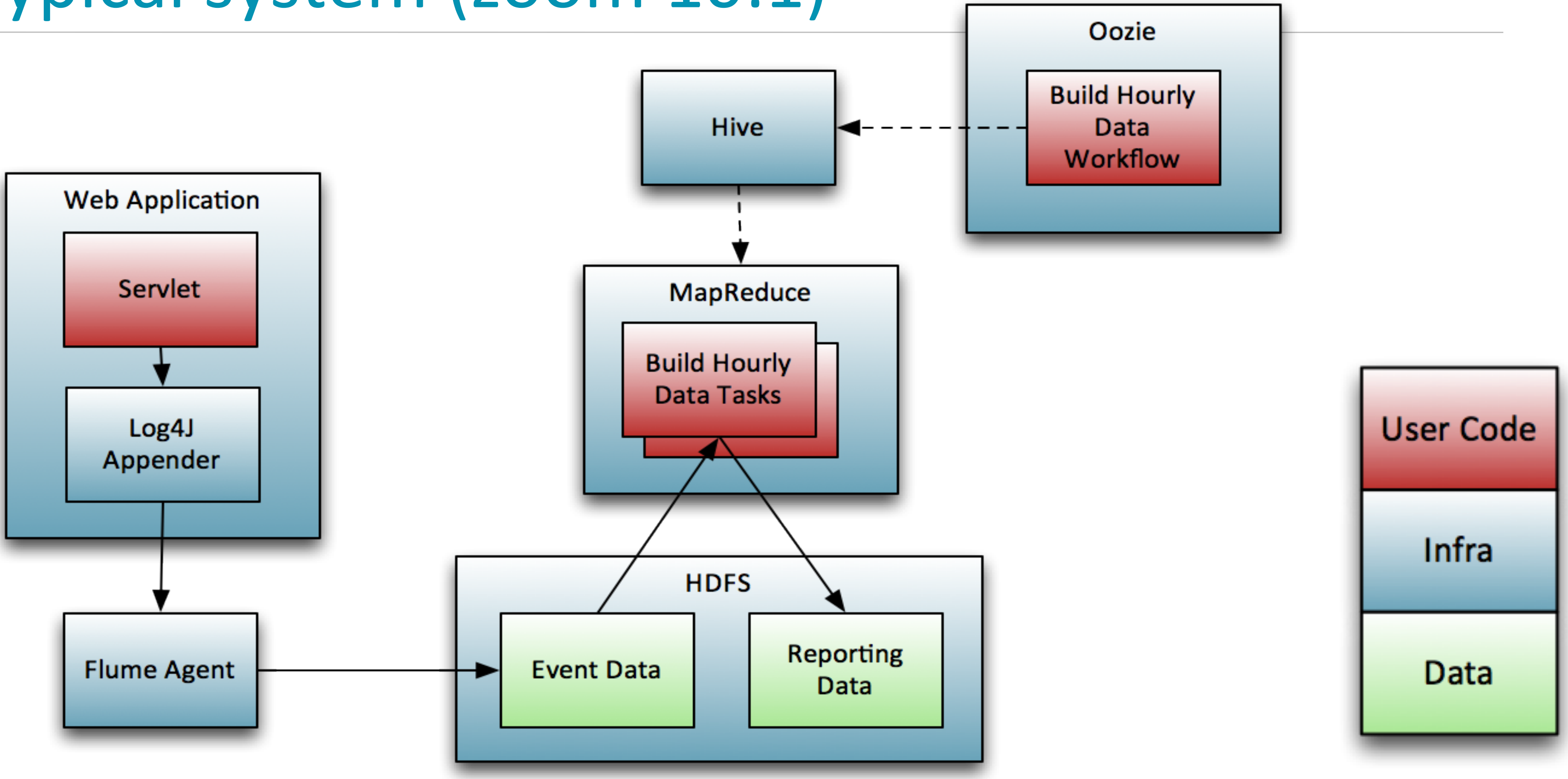
“[I]t’s not enough to just build a scalable and stable system; the system also has to be easy enough for thousands of internal developers of all types and all skill levels to use.”

http://gigaom.com/data/how-disney-built-a-big-data-platform-on-a-startup-budget/

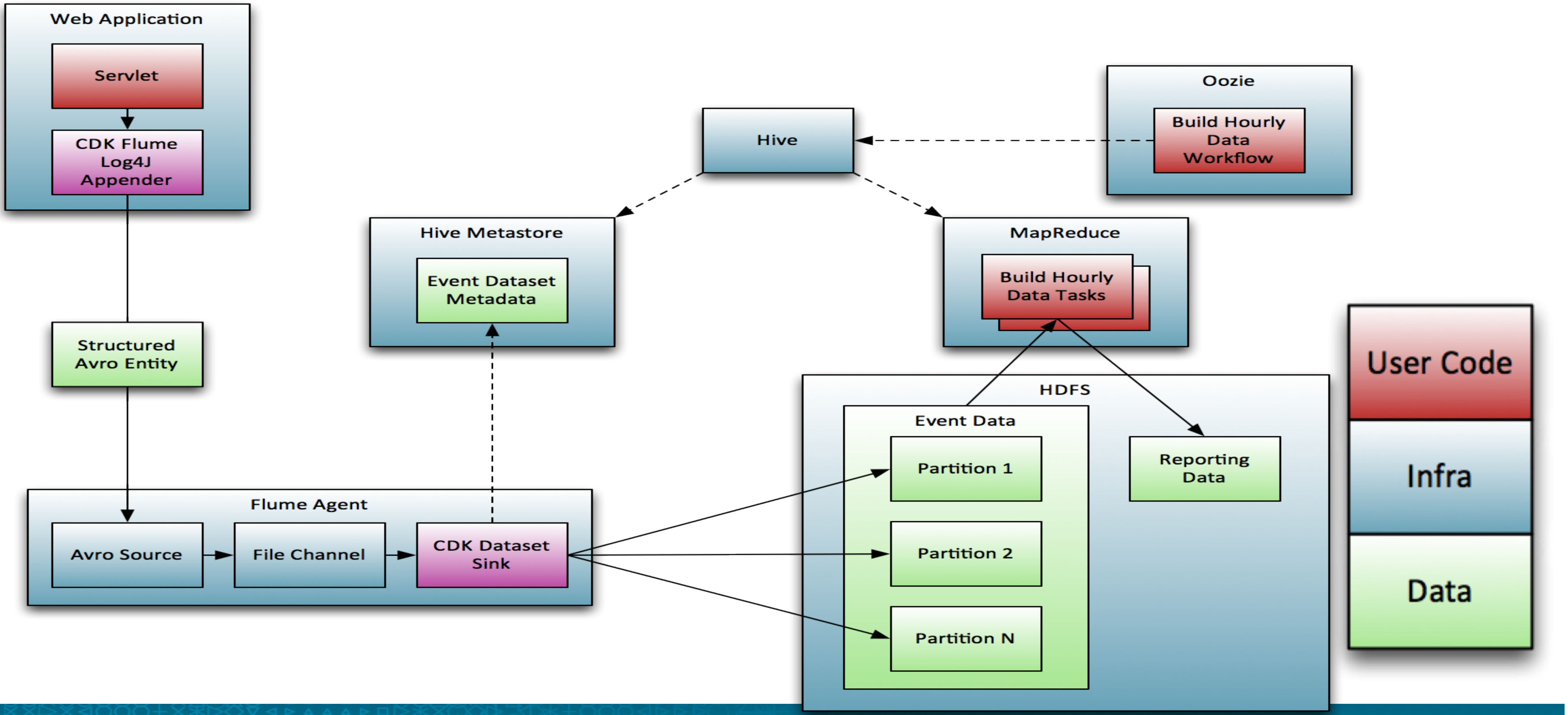
A typical system (zoom 100:1)



A typical system (zoom 10:1)



A typical system (zoom 5:1)



What you actually care about

- Getting data from A to B
- Using it later

Infrastructure details

- Serialization, file formats, and compression
- Metadata capture and maintenance
- Dataset organization and partitioning
- Durability and delivery guarantees
- Well-defined failure semantics
- Performance and health instrumentation

Wouldn’t it be nice…?

- Make Hadoop accessible to the enterprise developer
- Address the most common cases
- Codify expert patterns and practices for building data-oriented systems and applications.
- Let developers focus on business logic, not plumbing or infrastructure.
- Provide smart defaults for platform choices.
- Support piecemeal adoption via loosely-coupled modules

Kite SDK

- An open source set of libraries, guides, and examples for building data-oriented systems and applications
- Provides higher level APIs atop existing components of CDH
- Supports piecemeal adoption via loosely coupled modules

Kite SDK Data Module

- Logical abstractions of records, datasets and repositories with implementations for HDFS and HBase (upcoming)
- APIs to drastically simplify working with datasets in Hadoop filesystems. The Data module:
  - Handles automatic serialization and deserialization of Java POJOs as well as Avro Records.
  - Automatic compression.
  - File and directory layout and management.
  - Automatic partitioning based on configurable functions.
  - A metadata provider plugin interface to integrate with centralized metadata management systems.

Code

```
DatasetRepository repo = new FileSystemDatasetRepository.Builder()
  .fileSystem(FileSystem.get(new Configuration()))
  .directory(new Path(“/data”))
  .get();

Dataset events = repo.create(“events”,
  new DatasetDescriptor.Builder()
    .schema(new File(“event.avsc”))
    .partitionStrategy(
      new PartitionStrategy.Builder().hash(“userId”, 53).get()
    ).get()
);

DatasetWriter<GenericRecord> writer = events.getWriter();
writer.open();
writer.write(
  new GenericRecordBuilder(schema)
    .set(“userId”, 1)
    .set(“timeStamp”, System.currentTimeMillis())
    .build()
);
writer.close();
```

Data

```
/data
  /events
    /.metadata
      /schema.avsc
      /descriptor.properties
    /userId=0
      /10000000.avro
      /10000001.avro
    /userId=1
      /20000000.avro
    /userId=2
      /30000000.avro
```

Kite SDK Morphlines Module

- Pluggable, configuration-driven data transform library
  - Born out of Cloudera Search, but general purpose
  - Configure record transform stages in a container library
  - Use the library in Flume, MapReduce jobs, Storm, and other Java applications

Other Modules

- Maven plugin
  - Package, deploy, and execute “apps”
  - Execute dataset operations
- Examples
  - POJO, generic, and generated entity ingest
  - Dataset administrative operations
  - Crunch and MR integration
  - …

Future

- HBase
  - Extending data APIs to support random access
  - Same automatic serialization, schema management, etc.
- Higher-order data management
  - Common tasks
  - Think background compaction, conversion, etc.
- Integration with existing middleware frameworks
- Give us all your good ideas (and code)!

Kite SDK Resources

- Docs
    - http://kitesdk.org/docs/current/
- Examples
    - https://github.com/kite-sdk/kite-examples
- Source code
    - https://github.com/kite-sdk/

Binary artifacts available from Cloudera's Maven repository

- Twitter: @mark_grover
- Slides at http://www.slideshare.net/markgrover/applications-on-hadoop
- LinkedIn: linkedin.com/in/grovermark

Co-authoring O'Reilly book

- Titled 'Hadoop Application Architectures'
- How to build end-to-end solutions using Apache Hadoop and related tools
- Updates on Twitter: @hadooparchbook
- http://www.hadooparchitecturebook.com/