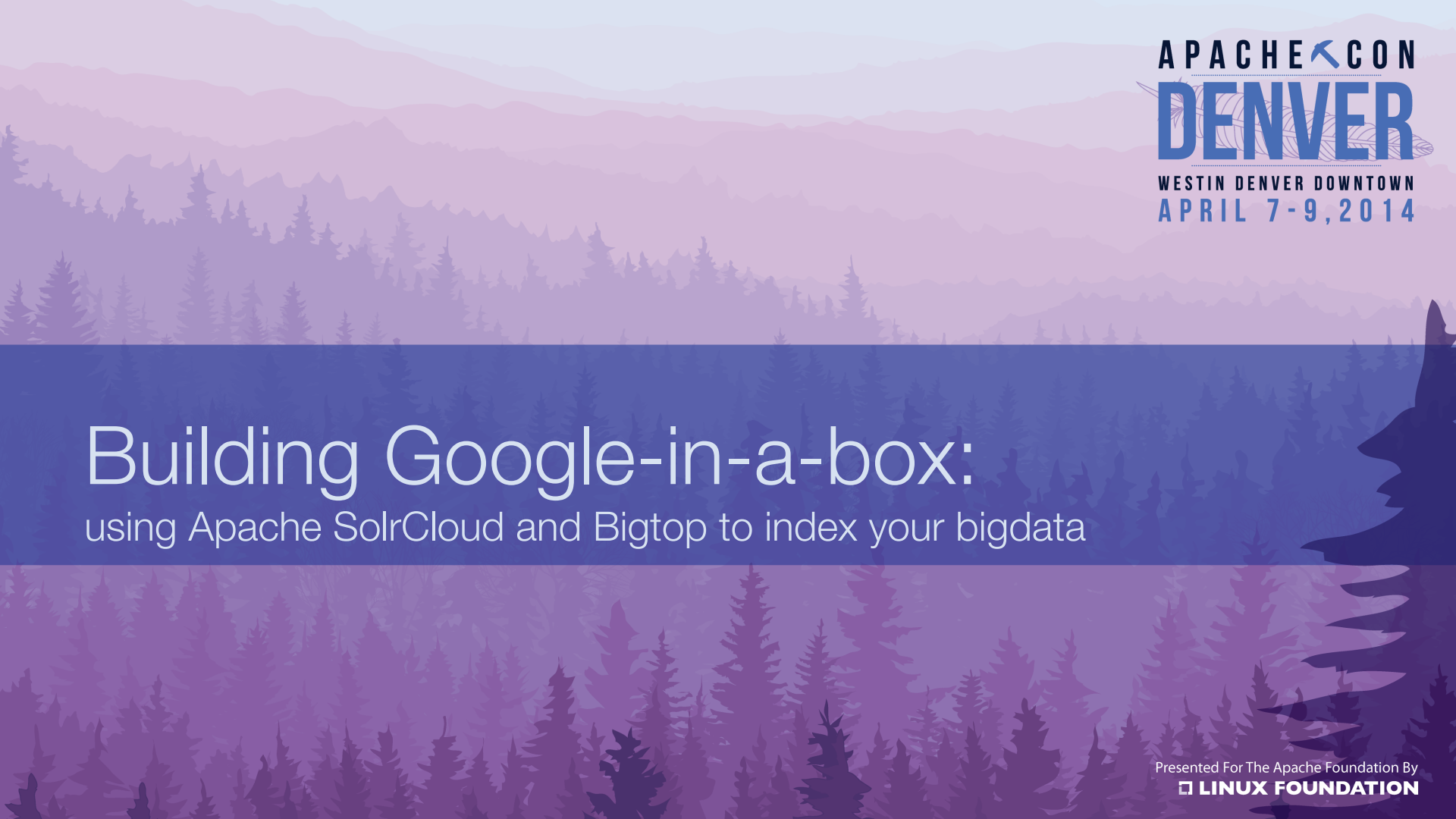


Building Google-in-a-box:

using Apache SolrCloud and Bigtop to index your bigdata

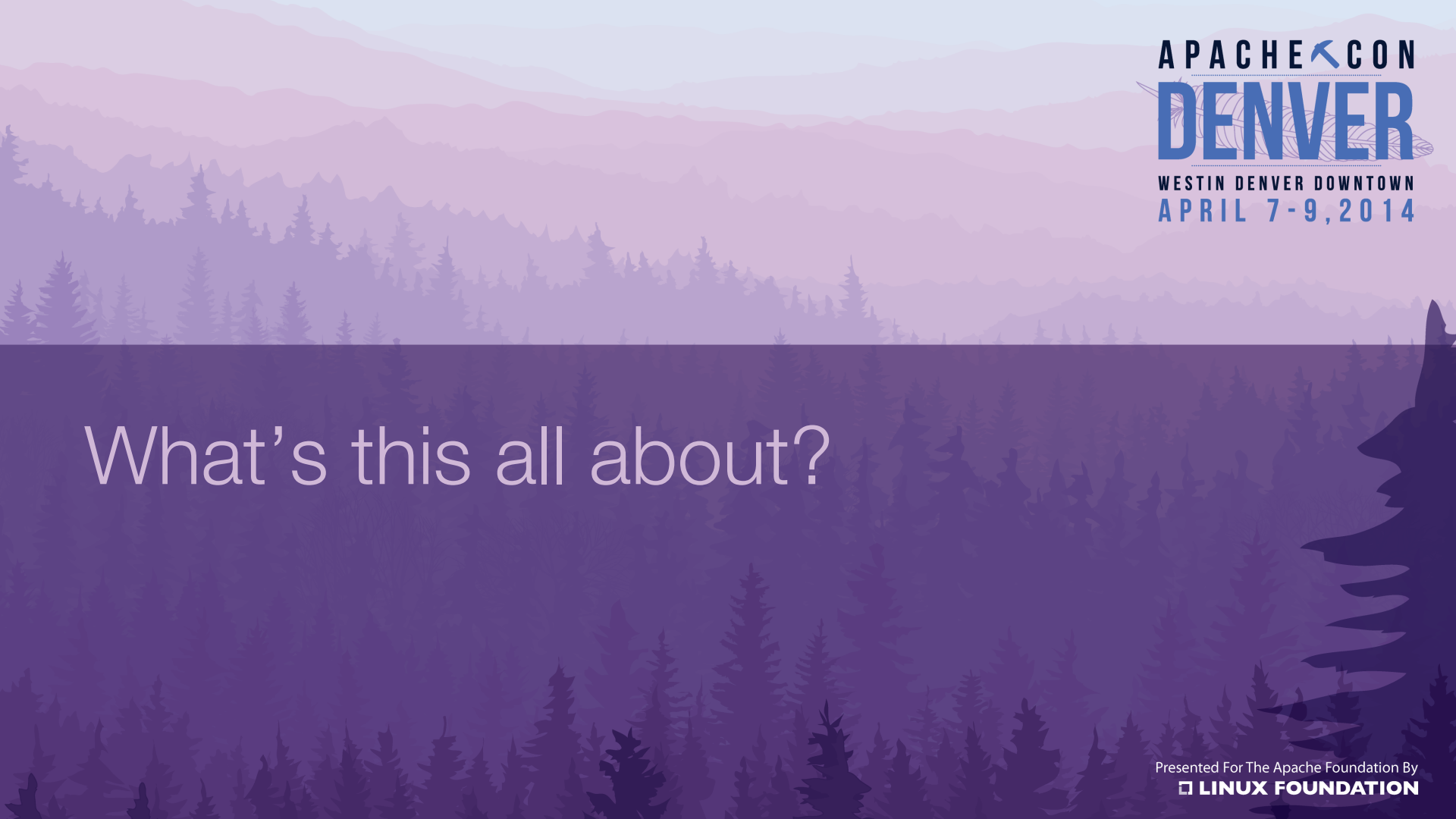
Presented For The Apache Foundation By
LINUX FOUNDATION


APACHECON
DENVER
WESTIN DENVER DOWNTOWN
APRIL 7-9, 2014
Who's this guy?
Presented For The Apache Foundation By
LINUX FOUNDATION

Roman Shaposhnik

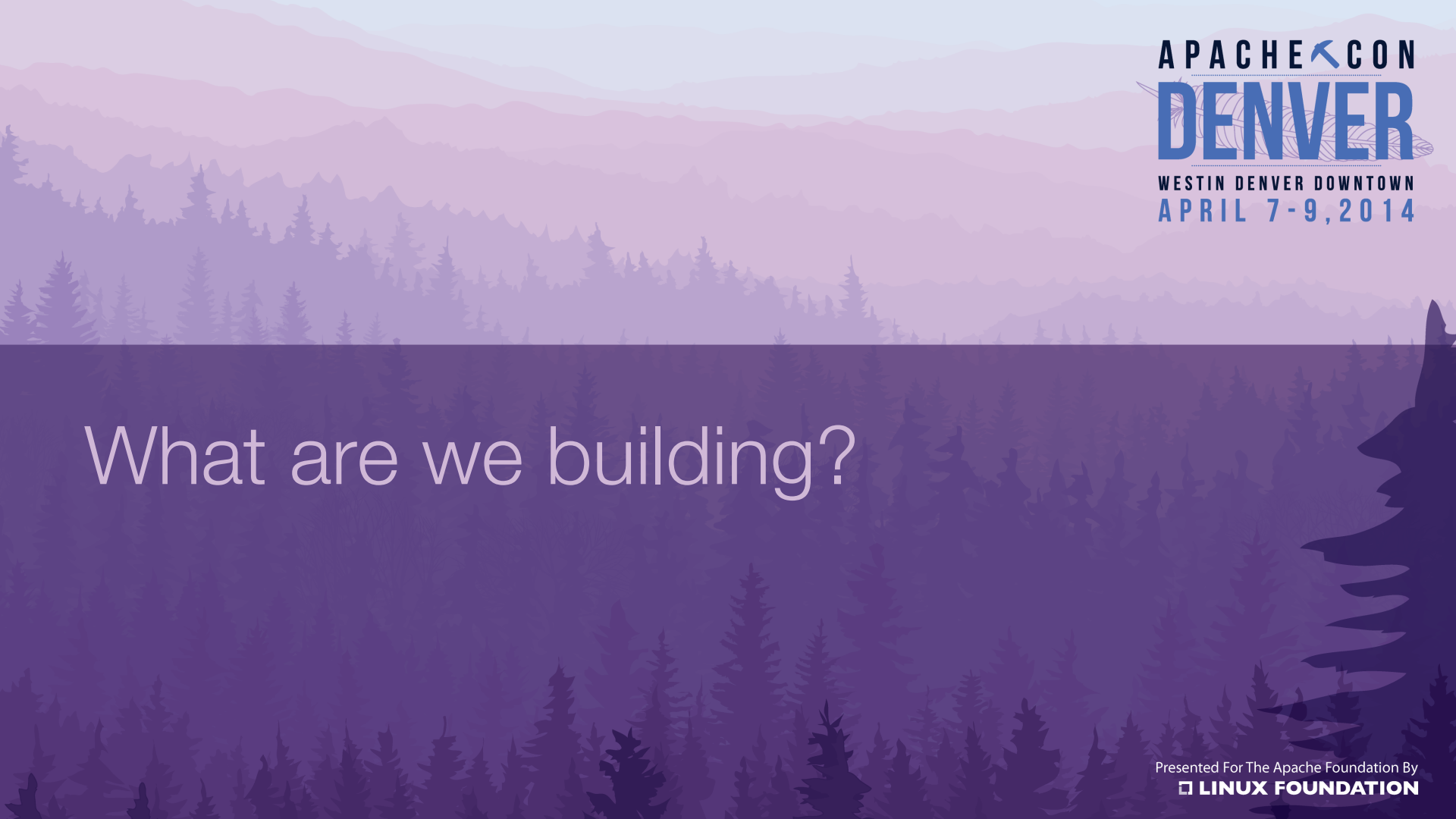
@rhatr or rvs@apache.org

- Sr. Manager at Pivotal Inc. building a team of ASF contributors
- ASF junkie
  - VP of Apache Incubator, former VP of Apache Bigtop
  - Hadoop/Sqoop/Giraph committer
  - contributor across the Hadoop ecosystem)
- Used to be root@Cloudera
- Used to be a PHB at Yahoo!
- Used to be a UNIX hacker at Sun microsystems
- First time author: “Giraph in action”

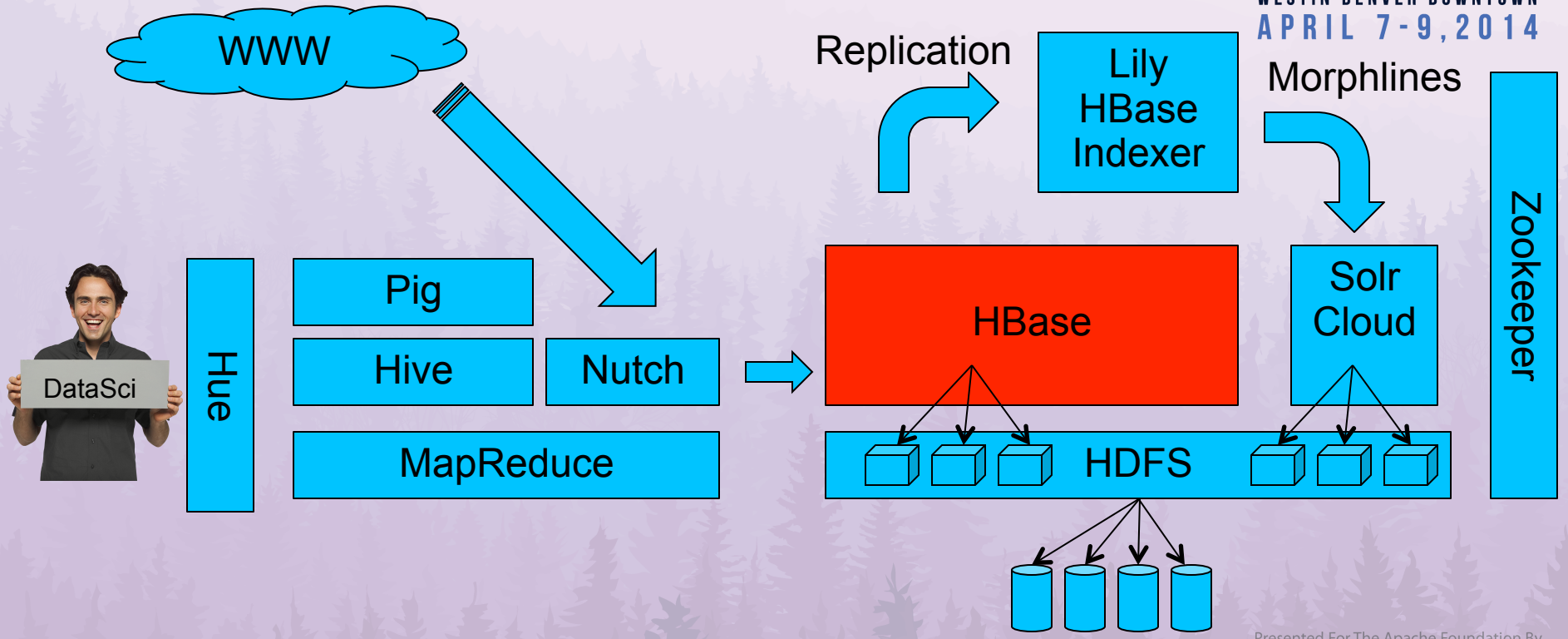

APACHECON
DENVER
WESTIN DENVER DOWNTOWN
APRIL 7-9, 2014
What's this all about?
Presented For The Apache Foundation By
LINUX FOUNDATION

This is NOT this kind of talk

This is *this* kind of a talk:


APACHECON
DENVER
WESTIN DENVER DOWNTOWN
APRIL 7-9, 2014
What are we building?
Presented For The Apache Foundation By
LINUX FOUNDATION

WWW analytics platform



Google papers

- GFS (Google FS) == HDFS
- MapReduce == MapReduce
- Bigtable == HBase
- Sawzall == Pig/Hive
- F1 == HAWQ/Impala

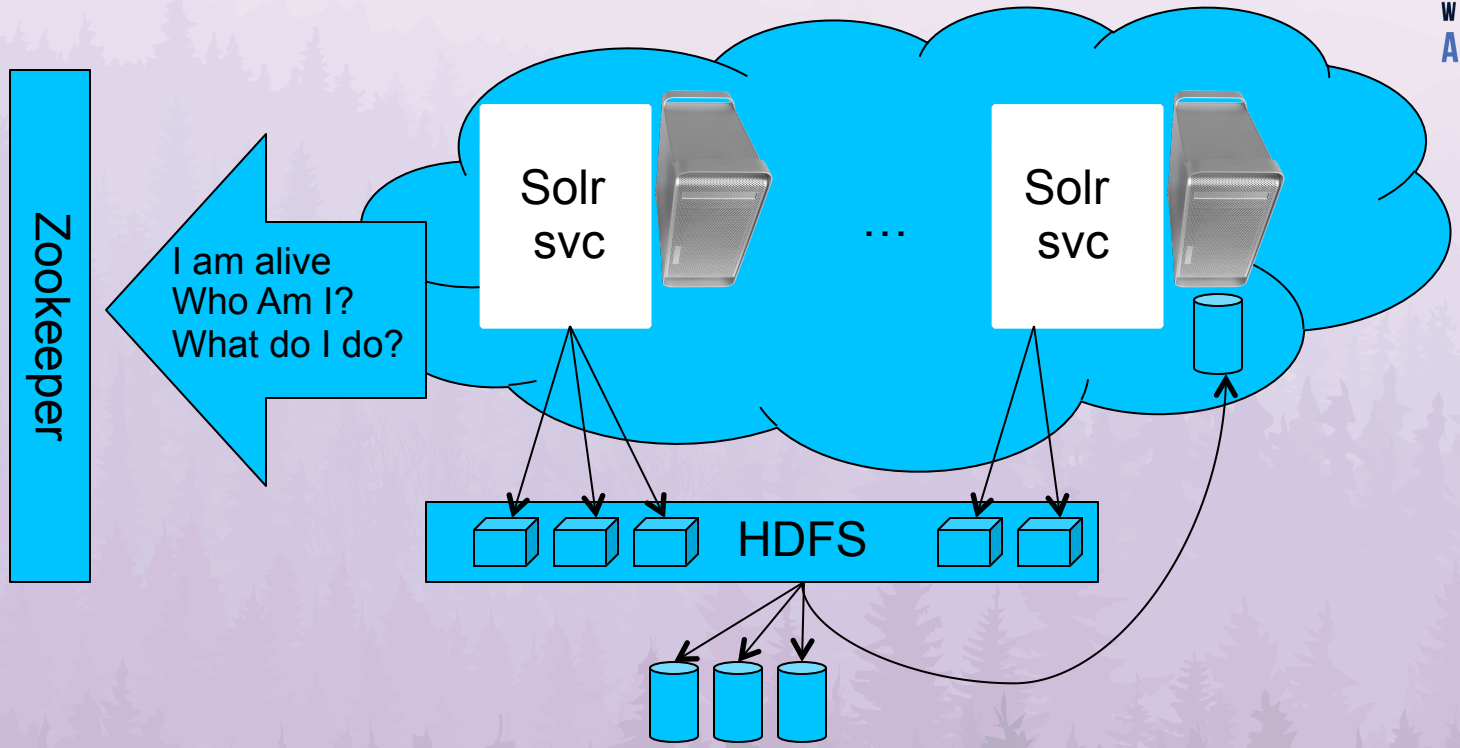
Storage design requirements

- Low-level storage layer: KISS
  - commodity hardware
  - massively scalable
  - highly available
  - minimalistic set of APIs (non-POSIX)
- Application specific storage layer
  - leverages LLSL
  - Fast r/w random access (vs. immutable streaming)
  - Scan operations

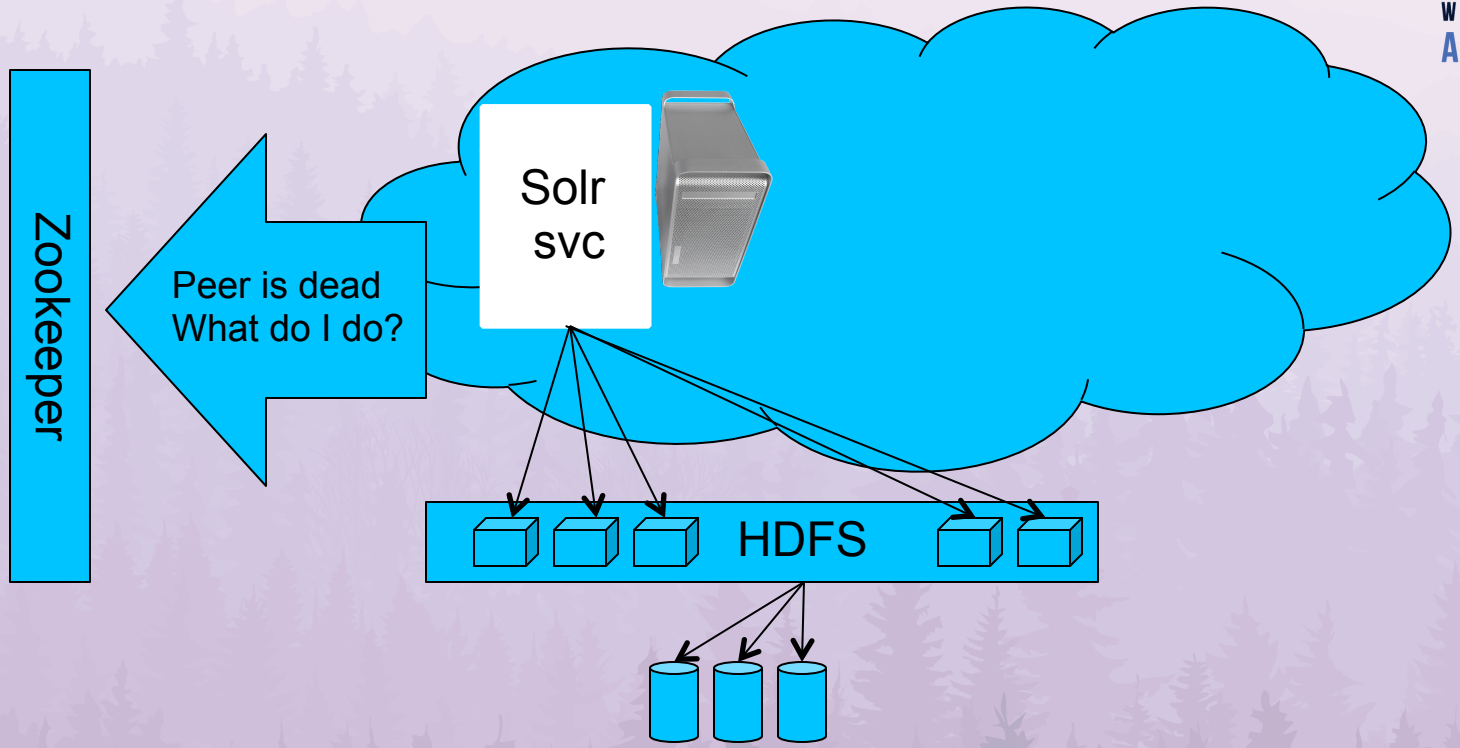
Design patterns

- HDFS is the “data lake”
  - Great simplification of storage administration (no-SANAS)
  - “Stateless” distributed applications persistence layer
- Applications are “stateless” compositions of various services
  - Can be instantiated anywhere (think YARN)
  - Can restart serving up the state from HDFS
  - Are coordinated via Zookeeper

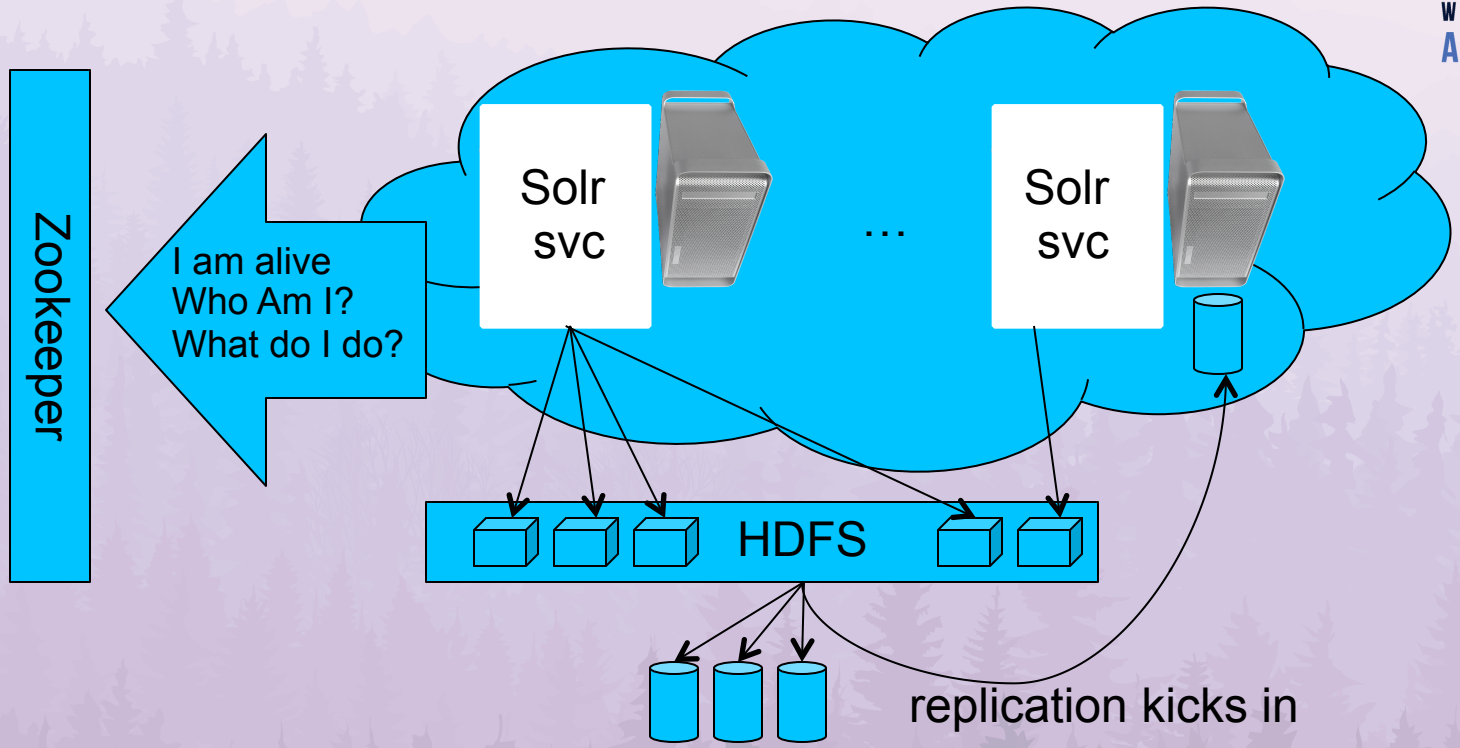
Application design: SolrCloud



Application design: SolrCloud



Application design: SolrCloud



How do we build something like this?

The bill of materials

- HDFS
- Zookeeper
- HBase
- Nutch
- Lily HBase indexer
- SolrCloud
- Morphlines (part of Project Kite)
- Hue
- Hive/Pig/…

How about?

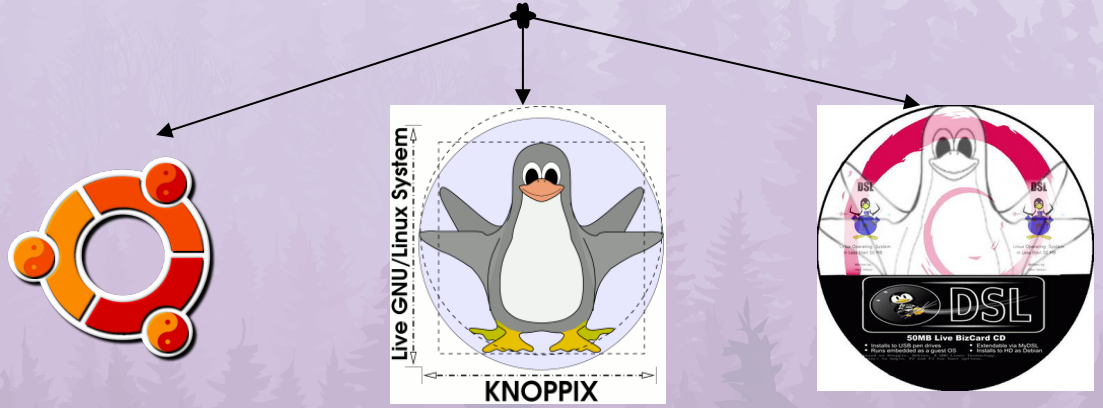
```
$ for comp in hadoop hbase zookeeper … ; do
    wget http://dist.apache.org/$comp
    tar xzvf $comp.tar.gz
    cd $comp ; mvn/ant/make install
    scp …
    ssh …
  done
```

How about?

```
$ for comp in hadoop hbase zookeeper … ; do
    wget http://d[illegible]pache.org/$com
    tar xzvf $comp[illegible]
    cd $comp ; mvn/a[illegible]
    scp …
    ssh …
  done
```

We've seen this before!

GNU Software → debian ← Linux kernel



Apache Bigtop!

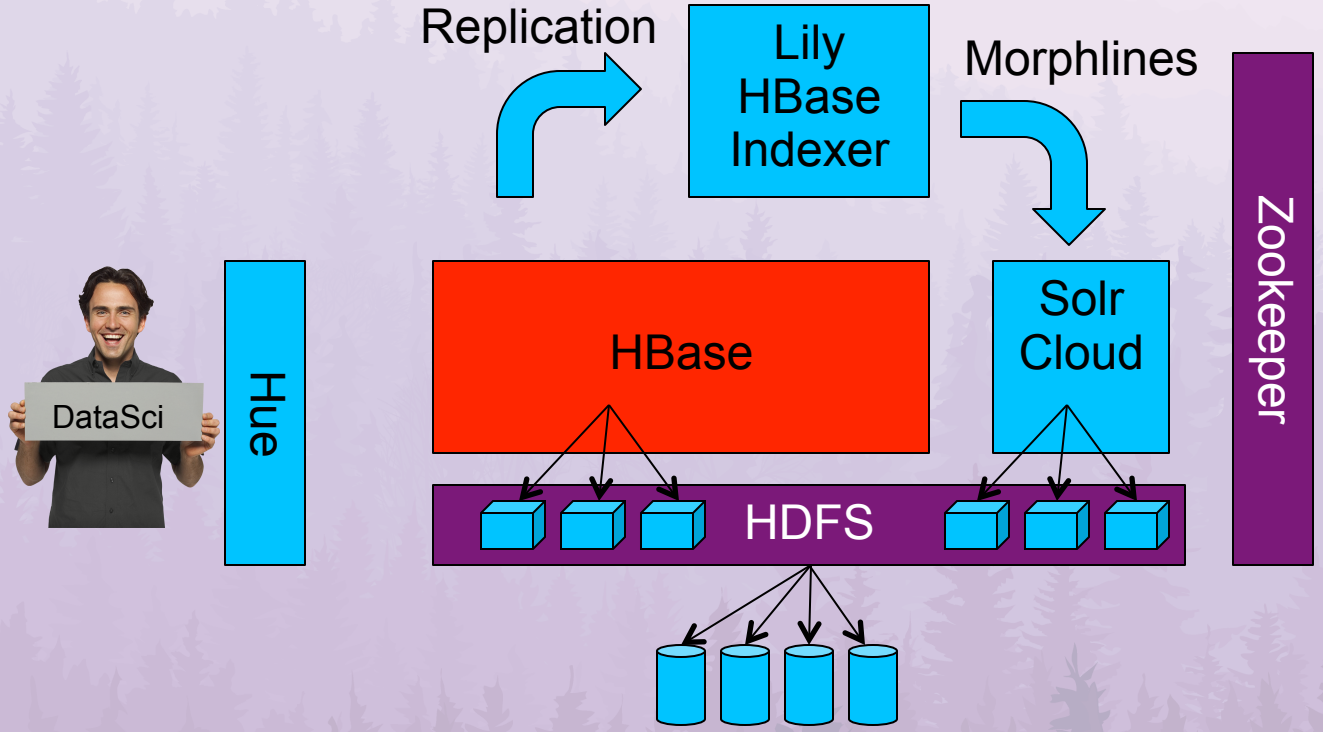
HBase, Solr..

Hadoop

cloudera Pivotal™ hortonworks


APACHECON
DENVER
WESTIN DENVER DOWNTOWN
APRIL 7-9, 2014
Lets get down to business
Presented For The Apache Foundation By
LINUX FOUNDATION

Still remember this?



HBase: row-key design

Indexing: schema design

- Bad news: no more “schema on write”
- Good news: you can change it on the fly
- Lets start with the simplest one:

```
<field name=”id" type=”string" indexed="true" stored="true” required=“true”/>
<field name=”text" type="text_general" indexed="true" stored="true"/>
<field name=“url” type=”string" indexed="true" stored="true”/>
```

Deployment

- Single node pseudo distributed configuration
- Puppet-driven deployment
  - Bigtop comes with modules
  - You provide your own cluster topology in cluster.pp

Deploying the ‘data lake’

- Zookeeper
  - 3-5 members of the ensemble

    # vi /etc/zookeeper/conf/zoo.cfg

    # service zookeeper-server init

    # service zookeeper-server start

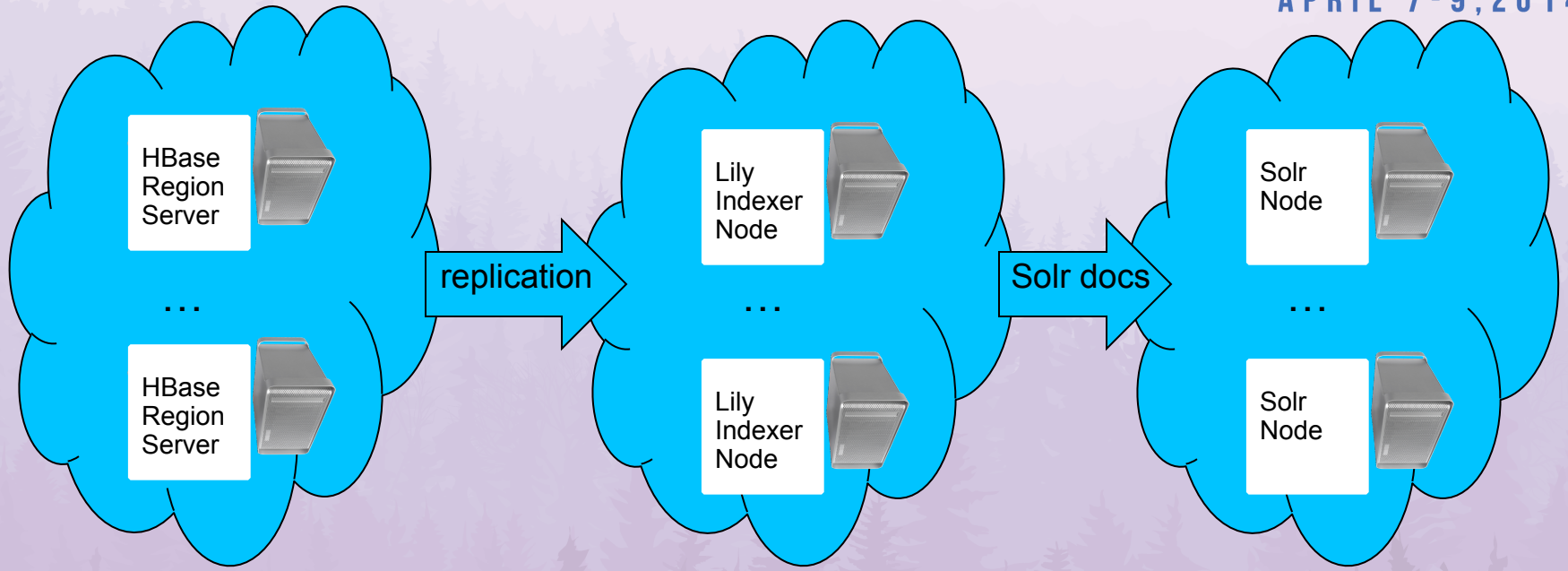
- HDFS
  - tons of configurations to consider: HA, NFS, etc.
  - see above, plus: /usr/lib/hadoop/libexec/init-hdfs.sh

HBase asynchronous indexing

- leveraging WAL for indexing
- can achieve infinite scalability of the indexer
- doesn't slow down HBase (unlike co-processors)
- /etc/hbase/conf/hbase-site.xml:

```
<property>
    <name>hbase.replication</name>
    <value>true</value>
</property>
```

Different clouds



HBase “cloud”

Lily Indexer “cloud”
home of Morphline ETL

SolrCloud

Lily HBase indexer

- Pretends to be a region server on the receiving end
- Gets records
- Pipes them through the Morphline ETL
- Feeds the result to Solr
- All operations are managed via individual indexers

Creating an indexer


```
$ hbase-indexer add-indexer
    --name web_crawl
    --indexer-conf ./indexer.xml
    --connection-param solr.zk=localhost/solr
    --connection-param solr.collection=web_crawl
    --zookeeper localhost:2181
```

indexer.xml

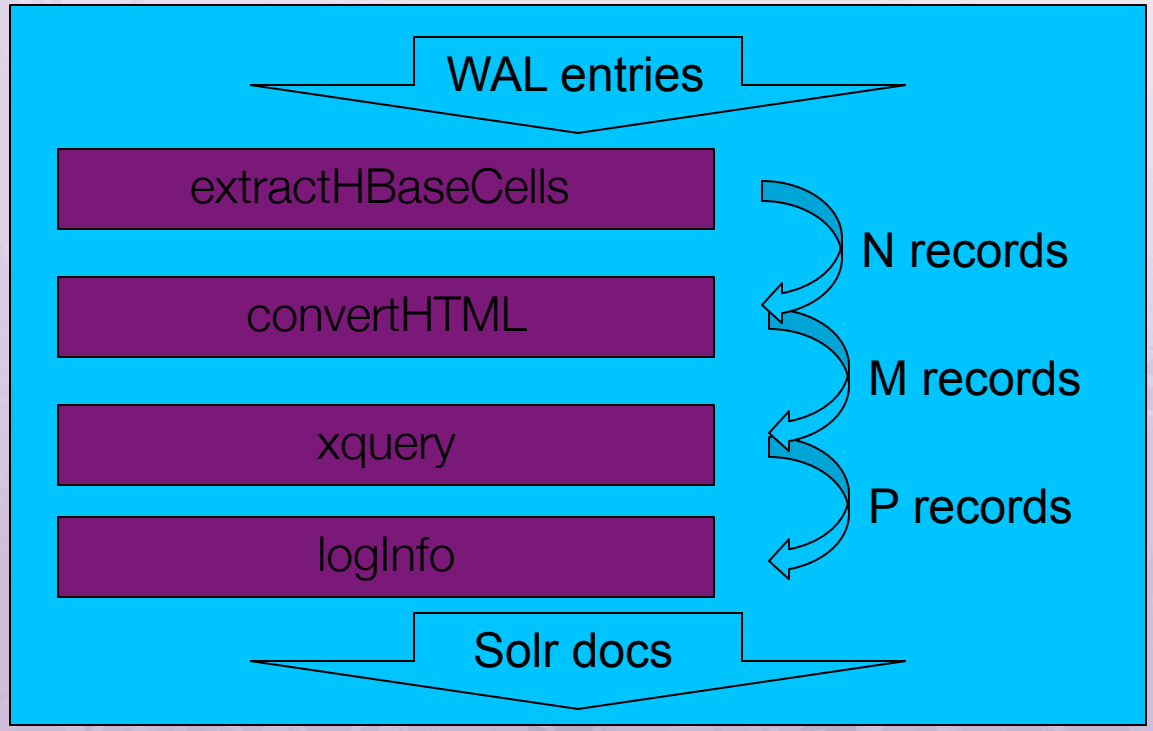

```
<indexer table="web_crawl"
mapper="com.ngdata.hbaseindexer.morphline.MorphlineResultToSolrMapper">

<param name="morphlineFile" value="/etc/hbase-solr/conf/morphlines.conf"/>
<!-- <param name="morphlineId" value="morphline1"/> →
</indexer>
```

Morphlines

- Part of Project Kite (look for it on GitHub)
- A very flexible ETL library (not just for HBase)
- “UNIX pipes” for bigdata
- Designed for NRT processing
- Record-oriented processing driven by HOCON definition
- Require a “pump” (most of the time)
- Have built-in syncs (e.g. loadSolr)
- Essentially a push-based data flow engine

Different clouds



Morphline spec

```
morphlines : [
  {  id : morphline1
     importCommands : ["org.kitesdk.morphline.**", "com.ngdata.**"]

     commands : [
       { extractHBaseCells {…} }
       { convertHTML {charset : UTF-8} }
       { xquery {…} }
       { logInfo { format : "output record: {}", args : ["@{}"] } }
     ]
  }
]
```

extractHBaseCells

```
{
  extractHBaseCells {
    mappings : [
      {
        inputColumn : "content:*"
        outputField : "_attachment_body"
        type : "byte[]"
        source : value
      }
    ]
  }
}
```

xquery

```
{ xquery {
      fragments : [ {
           fragmentPath : "/"
           queryString : """
   <fieldsToIndex>
      <webpage> {for $tk in //text() return concat($tk, ' ')} </webpage>
   </fieldsToIndex>
           """
        }
      ]
    }
}
```

SolrCloud

- Serves up lucene indices from HDFS
- A webapp running on bigtop-tomcat
  - gets configured via /etc/default/solr
    SOLR_PORT=8983
    SOLR_ADMIN_PORT=8984
    SOLR_LOG=/var/log/solr
    SOLR_ZK_ENSEMBLE=localhost:2181/solr
    SOLR_HDFS_HOME=hdfs://localhost:8020/solr
    SOLR_HDFS_CONFIG=/etc/hadoop/conf

Collections and instancedirs

- All of these objects reside in Zookeeper
  - An unfortunate trend we already saw with Lily indexers
- Collection
  - a distributed set of lucene indices
  - an object defined by Zookeeper configuration
- Collection require (and can share) configurations in instancedir
- Bigtop-provided tool: solrcrl

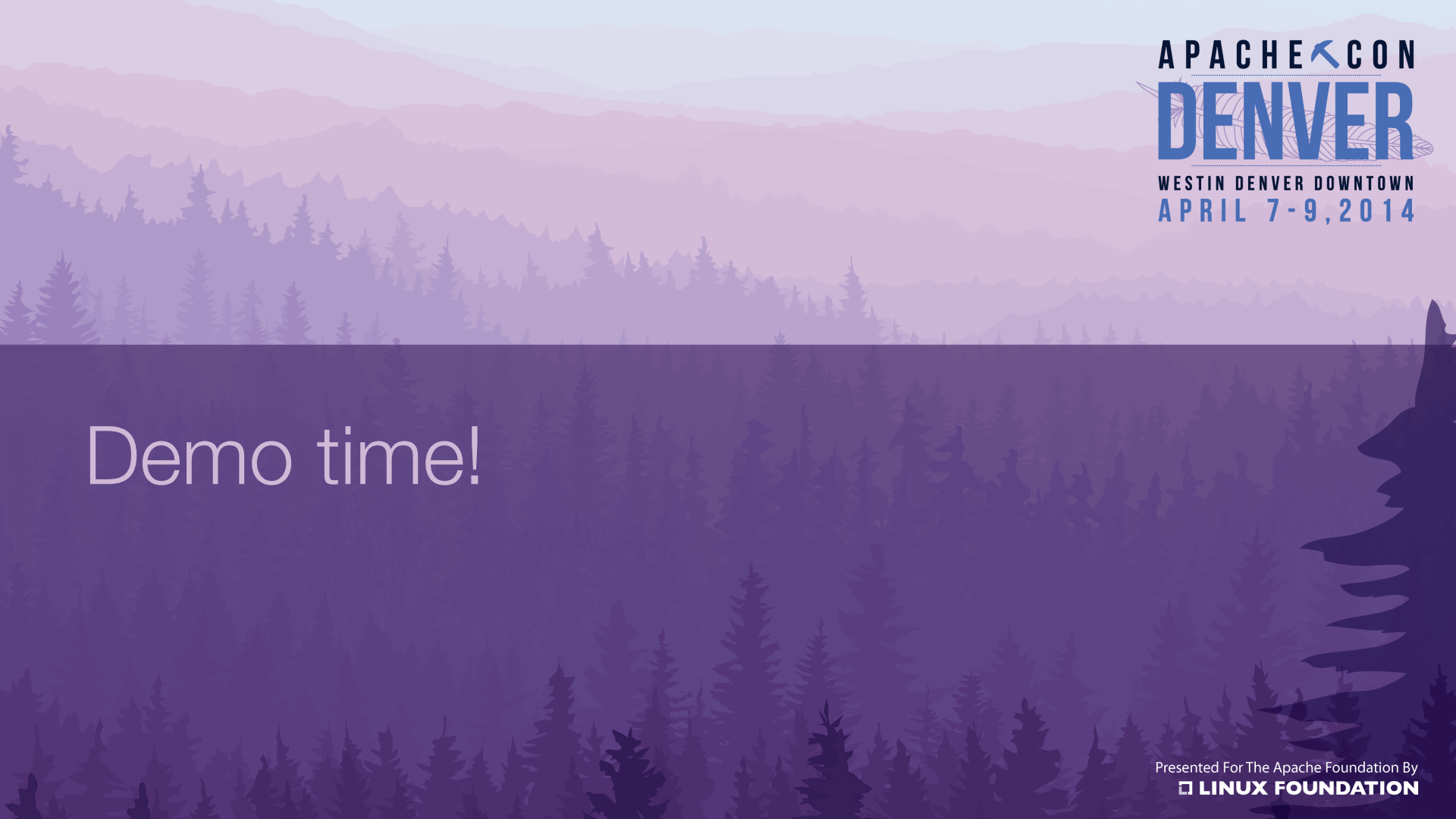
  $ solrctl [init|instacedir|collection|…]

Creating a collection

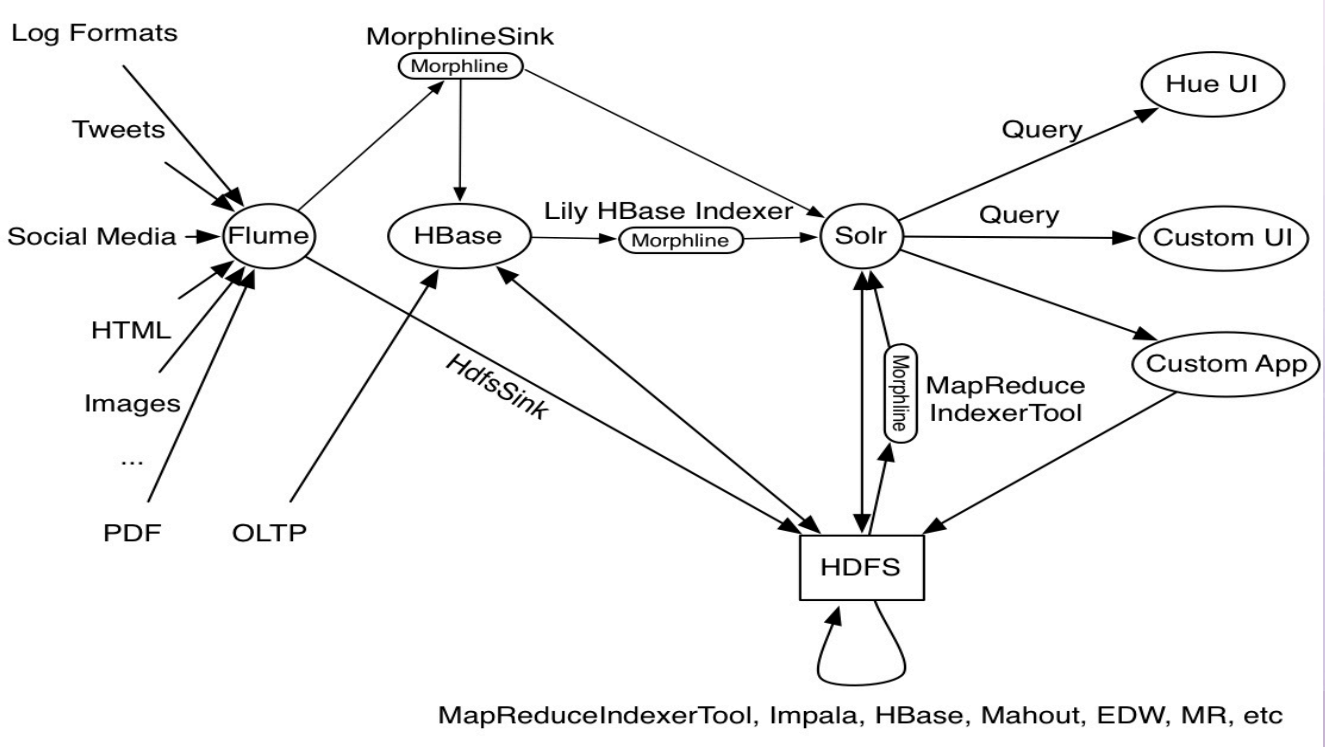
```
# solrctl init
$ solrctl instancedir --generate /tmp/web_crawl
$ vim /tmp/web_crawl/conf/schema.xml
$ vim /tmp/web_crawl/conf/solrconfig.xml
$ solrctl instancedir --create web_crawl /tmp/web_crawl
$ solrctl collection --create web_crawl -s 1
```

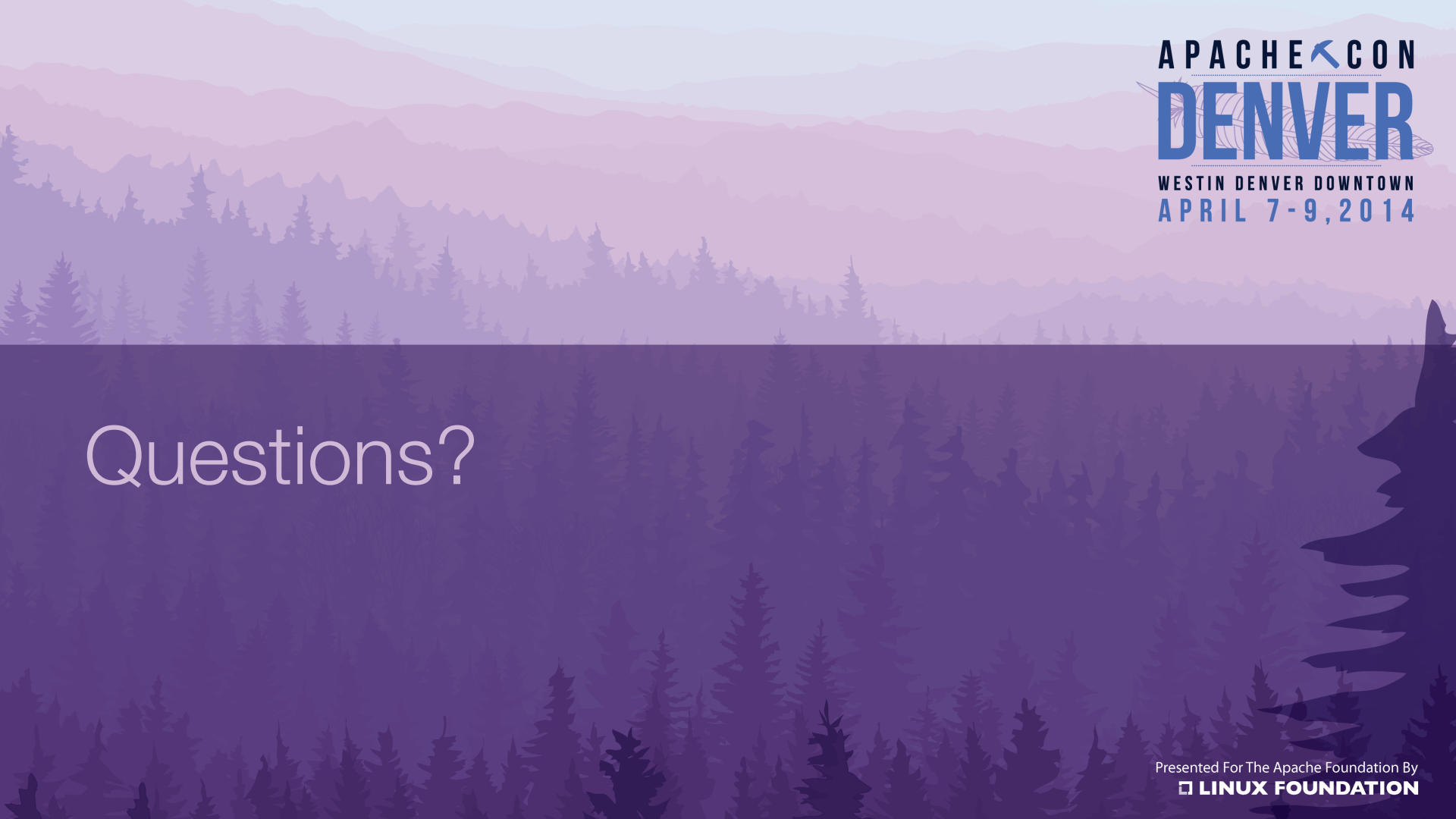
Hue

- Apache licensed, but not an ASF project
- A nice, flexible UI for Hadoop bigdata management platform
- Follows an extensible app model


APACHECON
DENVER
WESTIN DENVER DOWNTOWN
APRIL 7-9, 2014
Demo time!
Presented For The Apache Foundation By
LINUX FOUNDATION

Where to go from here




APACHECON
DENVER
WESTIN DENVER DOWNTOWN
APRIL 7-9, 2014
Questions?
Presented For The Apache Foundation By
LINUX FOUNDATION