Overview of Apache ZooKeeper

Tom Wheeler
Cloudera, Inc.
What’s Ahead?

- Tonight I will explain
  - What ZooKeeper is
  - What problems it can help you solve
  - How it works
  - How to install, configure and run it
  - Where you can learn more
What is ZooKeeper?

- A distributed coordination service
  - Reliable and highly-available
  - Inspired by Google’s Chubby lock service
    - But quite a bit different in design philosophy
- A top-level Apache project
  - Originally created at Yahoo!
What’s So Great About it?

- Flexible
  - Library
  - Corresponding network service
- Simple
  - Primitives
  - Recipes
- Loosely-coupled
- Built-in security
Why is ZooKeeper Needed?

- Imagine you’ve got a multithreaded program
  - And you need a lock to coordinate among threads
  - So you use the java.util.concurrent package

- And later your program has trouble scaling up
  - So you decide to scale out

- How do you handle locking across machines?
Why is ZooKeeper Needed?

“The network is reliable”

- Peter Deutsch, et al.
What Can You Do With It?

- Distributed locks
- Distributed queues
- Group membership
- Master elections
- Distributed configuration
- And much more…
Other ZooKeeper Properties

- Operations are ordered
  - Distributed state can lag, but it’s never wrong

- Updates are atomic
  - They either succeed completely or fail completely
  - There are no partially applied modifications

- Changes are durable
  - A change, once applied, will persist
  - Even if the machine fails. Even if Godzilla attacks.
Who Is Using It?

- ZooKeeper is part of the “Hadoop Ecosystem”
- Many Hadoop-related projects depend on it
  - HBase
  - HDFS High Availability
  - Flume
- But it’s not specific to Hadoop
  - No external dependencies (aside from Java)
Who Else Uses It?

- Other open source projects are using it too
  - Neo4J
  - Apache Solr (Cloud Edition)
  - Eclipse Communication Framework

- Many organizations also use ZooKeeper
  - Yahoo
  - Rackspace
  - Lots of others who choose not to be named…
ZooKeeper’s Data Model

- ZooKeeper models a hierarchical filesystem
  - Nodes in this tree are called *znodes*
  - A znode may contain data and/or other znodes

```
/  
|   |
|---|--
|  / |
| /   |
|      |
|      |
```

- blues
- rock
- chicago
delta
metal
punk
Znode Paths

- Every znode exists at some path
  - Paths are always both absolute and canonical
  - The API uses UNIX-style paths (e.g. `/rock/punk`)
The ZooKeeper API

- The API defines just a few operations, mainly
  - Create a node
  - Check if a node exists / Access the node
  - Delete a node
  - Get / set children
  - Get / set data
  - Plus a few others
    - Synchronizing state, registering watches, handling ACLs
Znode Types

- There are two main types of znodes
  - Persistent
    - Available until explicitly removed
  - Ephemeral
    - Tied to the session of the client which created it
    - Only available for the duration of that session
    - Ephemeral nodes cannot have children

- The type is specified at time of creation
Sequential Znodes

- Znodes optionally allow a sequence number
  - Just set a flag when creating the node
  - Actual name based on a counter’s current value
    - For example, foo becomes foo-0000000001
  - This is handy for maintaining a global order
    - Such as when creating a distributed lock
Security

- ZooKeeper now supports Kerberos security
- Authorization is done via ACLs
- Supports several types of restrictions
  - Message digest
  - Hostname
  - IP address
- Can limit access by function
  - Read, write, delete, etc.
ZooKeeper Standalone Mode

- Standalone mode is mainly used for development
- There is a single ZooKeeper daemon running
  - Handles both read and write requests from clients
ZooKeeper Clustered Mode

- There’s an *ensemble* of servers
  - One server is elected as the leader
  - Followers only service read requests
How Do You Install It?

- Get it from a mirror (zookeeper.apache.org)

  - $ tar -zxvf zookeeper-3.4.3.tar.gz
  - $ cd zookeeper-3.4.3
  - $ export PATH=$PATH:`pwd`/bin

- It’s also part of CDH
  - Cloudera’s Distribution including Apache Hadoop
  - You can install from packages (yum, apt-get, etc.)
  - This offers other conveniences (init scripts, etc.)
How Do You Configure It?

Three required configuration parameters

- **tickTime**: basic unit of time in ZooKeeper
- **dataDir**: local filesystem where data is stored
- **clientPort**: TCP port to which clients connect

If using cluster mode, list other ZK nodes too
How Do You Run It?

- If you installed from a tarball
  
  ```
  $ zkServer.sh start
  ```

- If you installed from CDH packages
  
  ```
  $ sudo service zookeeper-server start
  ```
How Do You Use It?

- Put the ZooKeeper JAR in your project
  - Just as you would for any other library
- Use the API to create an application
Where Do You Learn More?

- Apache ZooKeeper Web site
  - http://zookeeper.apache.org/

- Cloudera’s CDH4 documentation
  - http://www.cloudera.com/

- Hadoop: The Definitive Guide (O’Reilly)
  - Chapter 14 covers ZooKeeper in detail