PYCASSA:

Setting up and using Apache Cassandra with Python (in Windows)

By Tobi Bosede

WHAT IS CASSANDRA?

- open source noSQL database
- non-relational, distributed database designed to handle large amounts of data across many servers, providing high availability with no single point of failure.
- brings together the distributed systems technologies from Dynamo and the data model from Google's BigTable.

HOW IT SOLVES PROBLEMS

- Scalability: Store massive amounts of data
- Schema-less: Store unstructured data
- Fault Tolerance: Robust storing such that loss of data is almost impossible and flexible in maintenance insures
- Masterless Cluster: simplifies the ardor of managing data stored in different locations
- Fast retrieval: key-value store model results in indexed data

CASSANDRA INSTALLATION

- http://cassandra.apache.org/download/
- Download Tar.gz file and unzip
- Set up Java Home env variable (must have java)
  - http://php-cms-job.blogspot.com/2012/09/how-to-setting-javahome-variable-in.html
- Edit config file- cassandra.yaml and add directories
  - http://php-cms-job.blogspot.de/2012/09/how-to-install-cassandra-and-configure.html
- Navigate to location of C:\cassandra
- Test by typing “bin\cassandra –f”
- Should say “listening for thrift clients”

WHAT IS THRIFT?

- Interface Definition language(IDL)
- describes a software component's interface in a language-independent way, enabling communication between software components that do not share a language
- Allows portable access to the Cassandra
- generates source code for python (in this case) based on a Thrift IDL file

PYCASSA INSTALLATION

- Thrift is a prereq- “pip install thrift” (comes with anaconda distro)
- Manually install:
  - Use git to clone github repository (https://github.com/pycassa/pycassa/)
  - “git clone git://github.com/pycassa/pycassa.git”
- Technically you should also be able to use pip
  - I found that the pycassaShell was missing when I pip installed

CONNECT TO CASSANDRA

- In python import pycassa: no errors=> it correctly downloaded
- Start pycassaShell
  - In new terminal go to pycassa directory
  - Type “python pycassaShell”

```
C:\Users\Tobi\Documents\GitHub\pycassa>python pycassaShell
---------------------------------
Cassandra Interactive Python Shell
---------------------------------

Keyspace: None
Host: localhost:9160

ColumnFamily instances are only available if a keyspace is specified with -k/--k
eyspace

Schema definition tools and cluster information are available through SYSTEM_MAN
AGER.

In [1]:
```

CONNECT TO CASSANDRA CONTD

```
SYSTEM_MANAGER.create_keyspace('Keyspace1', strategy_options={"replication_factor": "1"})

SYSTEM_MANAGER.create_column_family('Keyspace1', 'ColumnFamily1')

from pycassa.pool import ConnectionPool

from pycassa.columnfamily import ColumnFamily
```

- The keyspace is the container for your application data, similar to a schema in a relational database. Keyspaces are used to group column families together.
- Column family is like a table in Cassandra

WHAT IS A SCHEMA?

- Predefine columns and data types
- Cassandra uses practically limited log-structured merge-tree storage engine rather than RDBMS' b-trees
- In a sparse-column engine, space is only used by columns present in each row
- No nulls taking up space for empty cells

7b976c48...	name: Bill Watterson	state: DC	birth_date: 1953
7c8f33e2...	name: Howard Tayler	state: UT	birth_date: 1968
7d2a3630...	name: Randall Monroe	state: PA	
7da30d76...	name: Dave Kellett	state: CA	

ADD DATA TO CASSANDRA

- Not necessary to create a schema, but good practice to do so => predefines every column and data type
- http://www.datastax.com/dev/blog/schema-in-cassandra-1-1

```
In [6]: pool = ConnectionPool('Keyspace1')

In [7]: col_fam = ColumnFamily(pool, 'ColumnFamily1')

In [8]: col_fam.insert('row_one', {'column': 'value'})
Out[8]: 1415588895151000L
```

RETRIEVE DATA FROM CASSANDRA

```
In [9]: col_fam.get('row_one')
Out[9]: OrderedDict([('column', 'value')])

In [10]: col_fam.get_count('row_one')
Out[10]: 1
```

- More ways to manipulate the database here:
  - http://pycassa.github.io/pycassa/tutorial.html

WHAT IS A NODE?

- Nodes are servers that help distribute and replicate data
- Multiple nodes can form a cluster with varying topology
- Assigned unique token to determine what partition key it is a replica for
- How to set up: http://www.datastax.com/documentation/cassandra/1.2/cassandra/initialize/initializeSingleDS.html

ADDING NODES TO CLUSTER

- Can defer until necessary
- Better to use virtual nodes
  - Calculating tokens and assigning them to each node is no longer required.
  - Rebalancing a cluster is no longer necessary because a node joining the cluster assumes responsibility for an even portion of the data.
- Physical nodes, in contrast, require calculation of tokens and rebalancing

QUESTIONS??

Thank you!