



## Report Title

BUILDING A DISTRIBUTED KEY VALUE STORE-  
BIG DATA 2017 - CLASS PROJECT

SNo	Name	USN	Class/Section
1	JAMPALA SREECHANDANA	01FB15ECS132	5 <sup>th</sup> /c
2	KEERTHANA NAGARAJ	01FB15ECS147	5 <sup>th</sup> /c
3	SHRAVANTHI.R	01FB15ECS219	5 <sup>th</sup> /D
4	ROHINI.D.V	01FB15ECS242	5 <sup>th</sup> /E

## Introduction

A distributed data store is a computer network where information is stored on more than one node, often in a replicated fashion. It is usually specifically used to refer to either a distributed database where users store information on a number of nodes, or a computer network in which users store information on a number of peer network nodes.

## Related work

<https://zookeeper.apache.org/>

Studied the main pages to understand the server client connections.

## ALGORITHM/DESIGN

1. Creation of a Client and Server java files using socket programming.
2. Start ZooKeeper and creation of Master Znode
3. Check for the presence of Master using ZooKeeper

### SERVER OPERATION

- Clients will send requests to the server
- Server will determine request type – put, get
- Server will determine if it can process the request or the request has to be serviced by other servers
- For self-served requests – it will process the request and send back status of response

### SERVER REPLICATION

- Based on the server name a hash code function assigns a random number.
- This random number is used to assign the last 8 bytes of the IP Address for the server.

Example: If the hash code returns a value 1234. Then required value =  $(1234) \% 255$   
Required value = 214

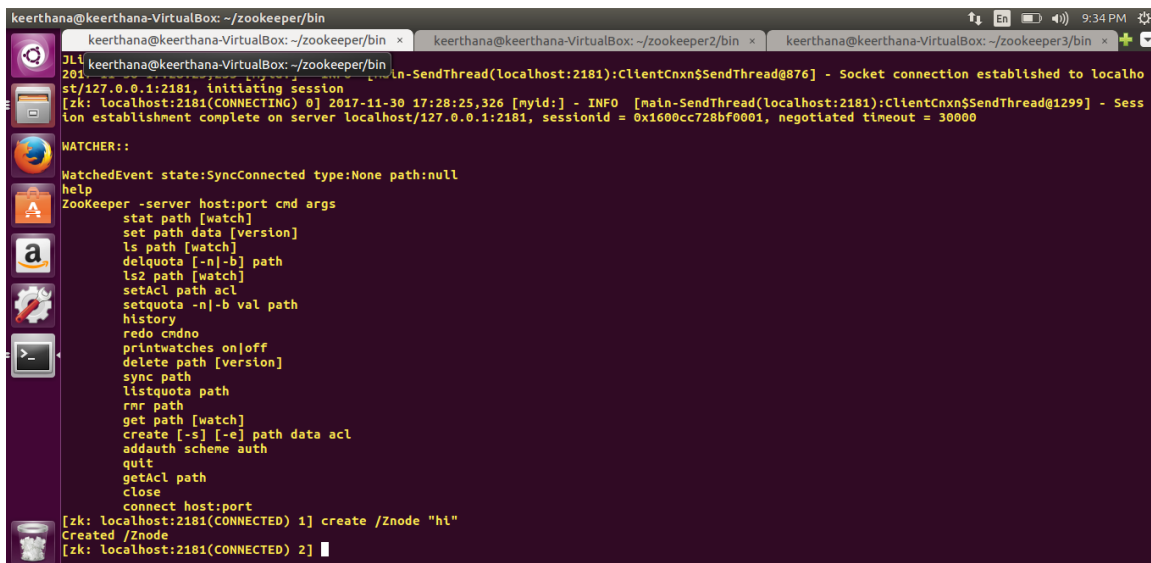
So, IP Address for the server will be 127.0.0.214 Its replica would be hash code return value of `hashCode(servername+r)`.

## HANDLING SERVER FAILURE

- Client tries connecting to server with key.
- On server failure, connects to master to get new list of keyserver mapping.
- Talks to the replica to retrieve data

## EXPERIMENTAL RESULTS

- Successfully established connection between client and server.
- Successful querying of keys by the client from various servers with distributed key value pairs.
- Server failure handled, its contents replicated in a replica-server and client retrieval from replica-server.



```
keerthana@keerthana-VirtualBox: ~/zookeeper/bin
keerthana@keerthana-VirtualBox: ~/zookeeper/bin x  keerthana@keerthana-VirtualBox: ~/zookeeper2/bin x  keerthana@keerthana-VirtualBox: ~/zookeeper3/bin x
JL4:11:30:25.326 [myid:] - INFO [main-SendThread(localhost:2181):ClientCnxn$SendThread@876] - Socket connection established to localho
st/127.0.0.1:2181, initiating session
[zk: localhost:2181(CONNECTING) 0] 2017-11-30 17:28:25.326 [myid:] - INFO [main-SendThread(localhost:2181):ClientCnxn$SendThread@1299] - Sess
ion establishment complete on server localhost/127.0.0.1:2181, sessionId = 0x1600cc728bf0001, negotiated timeout = 30000

WATCHER::
WatchedEvent state:SyncConnected type:None path:null
help
ZooKeeper -server host:port cmd args
stat path [watch]
set path data [version]
ls path [watch]
delquota [-n|-b] path
ls2 path [watch]
setAcl path acl
setquota [-n|-b val path
history
redo cmdno
printwatches on|off
delete path [version]
sync path
listquota path
rmr path
get path [watch]
create [-s] [-e] path data acl
addauth scheme auth
quit
getAcl path
close
connect host:port
[zk: localhost:2181(CONNECTED) 1] create /Znode "hi"
Created /Znode
[zk: localhost:2181(CONNECTED) 2] █
```

```
keerthana@keerthana-VirtualBox: ~/zookeeper2/bin
keerthana@keerthana-VirtualBox: ~/zookeeper/bin x  keerthana@keerthana-VirtualBox: ~/zookeeper2/bin x  keerthana@keerthana-VirtualBox: ~/zookeeper3/bin x
rro:)
2017-11-30 17:32:50,978 [myid:] - INFO [main-SendThread(localhost:2181):ClientCnxn$SendThread@876] - Socket connection established to localho
st/127.0.0.1:2181, initiating session
2017-11-30 17:32:50,997 [myid:] - INFO [main-SendThread(localhost:2181):ClientCnxn$SendThread@1299] - Session establishment complete on serve
r localhost/127.0.0.1:2181, sessionId = 0x1600cc728bf0002, negotiated timeout = 30000

WATCHER::

WatchedEvent state:SyncConnected type:None path:null
help
ZooKeeper -server host:port cmd args
    stat path [watch]
    set path data [version]
    ls path [watch]
    delquota [-n|-b] path
    ls2 path [watch]
    setAcl path acl
    setquota -n|-b val path
    history
    redo cmdno
    printwatches on|off
    delete path [version]
    sync path
    listquota path
    rmr path
    get path [watch]
    create [-s] [-e] path data acl
    addauth scheme auth
    quit
    getAcl path
    close
    connect host:port
[zk: localhost:2181,localhost:2182,localhost:2183(CONNECTED) 1] ls /
[Znode, zookeeper]
[zk: localhost:2181,localhost:2182,localhost:2183(CONNECTED) 2] █
```

```
keerthana@keerthana-VirtualBox: ~/zookeeper3/bin
keerthana@keerthana-VirtualBox: ~/zookeeper/bin x  keerthana@keerthana-VirtualBox: ~/zookeeper2/bin x  keerthana@keerthana-VirtualBox: ~/zookeeper3/bin x
Jline support is enabled
2017-11-30 19:26:57,884 [myid:] - INFO [main-SendThread(localhost:2181):ClientCnxn$SendThread@876] - Socket connection established to localho
st/127.0.0.1:2181, initiating session
2017-11-30 19:26:57,923 [myid:] - INFO [main-SendThread(localhost:2181):ClientCnxn$SendThread@1299] - Session establishment complete on serve
r localhost/127.0.0.1:2181, sessionId = 0x1600cc728bf0003, negotiated timeout = 30000

WATCHER::

WatchedEvent state:SyncConnected type:None path:null
[zk: localhost:2181,localhost:2182,localhost:2183(CONNECTED) 0] help
ZooKeeper -server host:port cmd args
    stat path [watch]
    set path data [version]
    ls path [watch]
    delquota [-n|-b] path
    ls2 path [watch]
    setAcl path acl
    setquota -n|-b val path
    history
    redo cmdno
    printwatches on|off
    delete path [version]
    sync path
    listquota path
    rmr path
    get path [watch]
    create [-s] [-e] path data acl
    addauth scheme auth
    quit
    getAcl path
    close
    connect host:port
[zk: localhost:2181,localhost:2182,localhost:2183(CONNECTED) 1] ls /
[Znode, zookeeper]
[zk: localhost:2181,localhost:2182,localhost:2183(CONNECTED) 2] █
```

## FUTURE ENHANCEMENTS

Handling additional servers, more than three

## REFERENCES

<https://askubuntu.com/questions/>

<https://tutorialpoint.com/zookeeper>

<https://zookeeper.apache.org/hadoop/zookeeper>

<https://myjeeva.com/zookeeper/clustering/setup.html>

<https://javatpoint.com/socket-programming>