**SymmetricDS Community Edition**

1. **Firstly Download the SymmetricDS Community edition from** [**www.jumpmind.com**](http://www.jumpmind.com) **website. Unzip the file symmetric-server-3.9.1zip on both Master Node(MS-SQL server) and Slave Node(Postgresql server).**
2. **On master node Copy the corp-000.properties file from symmetric-server-3.9.1\samples path and paste this file to symmetric-server-3.9.1\engines path. Follow same procedure on target node but copy store-001.properties file.**
3. **Now do the changes in the parameter of the corp-000.properties and store-001.properties files.**

**In corp-000.properties**

**engine.name = corp-000**

**db.driver=net.sourceforge.jtds.jdbc.Driver**

**db.url=jdbc:jtds:sqlserver://hostip:port/databasename;useCursors=true;bufferMaxMemory=10240;lobBuffer=5242880**

**db.user=put the database user here**

**db.password=put the database password here**

**sync.url=http://hostip:31415/sync/corp-000**

**group.id=corp**

**external.id=000**

**In store-001.properties**

**engine.name=store-001**

**sync.url =** [**http://hostip:31415/sync/store-001**](http://hostip:31415/sync/store-001)

**db.driver=org.postgresql.Driver**

**db.url=jdbc:postgresql://hostip:port/databasename?protocolVersion=2&stringtyp e=unspecified**

**db.user=put the database user here**

**db.password=put the database password here**

**registration.url=http://masternodeip:31415/sync/corp-000**

**group.id=store**

**external.id=001**

1. **Create system tables on master node by running following command on master node:**

symmetric-server-3.9.1/bin/symadmin --engine corp-000 create-sym-tables

1. Now Insert the values in the system tales:

**insert** **into** SYM\_NODE\_GROUP

(node\_group\_id)values(‘store’);

**insert** **into** SYM\_NODE\_GROUP\_LINK

(source\_node\_group\_id, target\_node\_group\_id, data\_event\_action)**values** ('corp', 'store', 'P');

Where P = Push

**insert** **into** SYM\_NODE\_GROUP\_LINK

(source\_node\_group\_id, target\_node\_group\_id, data\_event\_action)

**values** ('store', 'corp', 'W');

Where W = Wait for pull

**insert** **into** SYM\_ROUTER (router\_id,

source\_node\_group\_id, target\_node\_group\_id, create\_time,

last\_update\_time) **values** ('corp-2-store','corp', 'store',current\_timestamp, current\_timestamp);

**insert** **into** SYM\_CHANNEL (channel\_id, processing\_order, max\_batch\_size, max\_batch\_to\_send,extract\_period\_millis, batch\_algorithm, enabled, description)**values** ('patient\_dtl', 10, 1000, 10, 0, 'default', 1, ‘Patient Details');

**insert** **into** SYM\_TRIGGER (trigger\_id, source\_table\_name,

channel\_id, last\_update\_time, create\_time)**values** ('patient\_dtl', 'patient\_dtl', 'patient\_dtl', current\_timestamp, current\_timestamp);

**insert** **into** SYM\_TRIGGER\_ROUTER(trigger\_id, router\_id, initial\_load\_order, create\_time,last\_update\_time)**values** ('patient\_dtl', 'corp-2-store', 1, current\_timestamp,

current\_timestamp);

The Setup and configuration is now completed.

1. **We will now start both SymmetricDS nodes**

**On master node**:

symmetric-server-3.9.1/bin/sym

**On Slave Node:**

symmetric-server-3.9.1/bin/

and run ./sym

This command on slave node system tables automatically. It begins polling the server node in order to register.

1. **Registering a Node**Open registration for the slave node server by executing the below command on master node cmd:

symmetric-server-3.9.1/bin/symadmin –-engine corp-000 open-registration store 001

 The registration is now opened for a node group called “store” with an external identifier of “001”.

1. **Sending an Initial Load**

Next, we will send an initial load of data to our store (that is, the slave node)

Send an initial load of data to the client node server by executing following command on master node cmd:

symmetric-server-3.9.1/bin/symadmin –-engine corp-000 reload-node 001