

Manual for Multicast NSQ

Multicast Implementation

To be able to run multicast NSQ, your network should have multicast enabled and setup (or at least the network should broadcast all multicast traffic).

In current implementation of multicast, the multicast address (group address) is statically assigned to NSQ daemon. NSQ daemon will create multicast connection for default channel [*topic+n#ephemeral*]

For later implementation, we will let the lookup daemon manage the assignment of multicast address for each topic.

Compile the Code

To make sure you can compile the code successfully, do the following steps

- Add path `$GOPATH/bin` into your environment variable `$PATH` , one way to do that is add a line in your `~/.bash_profile` file.

```
$PATH=$PATH:$GOPATH/bin
```

and then run command `source ~/.bash_profile`

- Checkout the source code (it does not matter if you have any compiling error at this step). Skip this step if you already have the depository.

```
go get github.com/WU-CPSL/RTM-0.1/...
```

- Run command which compiles our message daemon, client-library, lookup daemon. Currently only rtm, the

rate control version is supported.

```
go install -tags rtm github.com/WU-CPSL/RTM-0.1/nsqio/nsq/...
```

- Compile the test programs

```
go install -tags rtm github.com/WU-CPSL/RTM-0.1/nsqio/Test/...
```

- Copy `$GOPATH/src/github.com/WU-CPSL/RTM-0.1/nsqio/Test/multicast-test` directory to a user directory, say `~/tmp/test/`, go to the directory and create two directories `htmp` and `ltmp`.

Run Multicast Test

To run the multicast test with the scripts in multicast-test directory.

Quick Run

- Before initiating a daemon, create a folder under your test script directory, with folder name `htmp` and `ltmp` as specified in parameter "data-path".
- Run nsqlookup daemon on one server

```
# start the nsqlookup daemon, listen to 4161 port.  
nsqlookupd -multicast_flag &
```

Flag `multicast_flag` is used to enable multicast in nsqlookupd. Nsqlookupd will return multicast address to nsqd when new topic is registered. nsqd will setup default multicast channel `multicast`.

- On the same server run NSQ Daemon with script `daemon_create.sh`

```
# interface_name is the interface in the machine that you want to use for multicast  
sh daemon_create.sh [interface_name]
```

- On the same server start publisher(s) that pub messages for topic.

```
# start producers all connected to high priority, you can pass lookup daemon address and port  
sh publisher_create.sh [addr [port]]
```

- On another server start subscriber(s) that reads topic message

```
# start subscribers, you can pass the lookup daemon address and port  
.  
sh subscriber_create.sh [addr [port]]
```

- After you finish the test or want to terminate test run `terminater.sh` script

```
# it kills all related processes  
sh terminate.sh
```

Customize your test

- **daemon_create.sh** script creates two NSQ daemons, which correspondingly register themselves as "HIGH" priority or "LOW" priority daemon at the lookup daemon. We use "chrt" command to configure the scheduling priority of these two daemon processes.
 - Some arguments you can change in `daemon_create.sh` base on your situation.
 - "connection-type" indicate the connection type, it can be either TCP (0) or Multicast (1).
 - "broadcast-address" indicate the address IP address and port in the form "addr:port"
 - "lookupd-tcp-address" to your own lookup daemon address
 - `daemon_create_tcp.sh` provides a quick example for using tcp for daemon

- **`publisher_single_topic.sh`** script creates multiple publishers
 - Some arguments you can change
 - "num" indicates number of publishers to create
 - "nsqlookupaddr" and "nsqlookupport" are the lookup address and port for lookup daemon
- **`subscriber_single_topic.sh`** script creates multiple subscribers to same topic and channel
 - Some arguments you can change
 - "num" indicates number of subscribers to create
 - "nsqlookupaddr" and "nsqlookupport" are the lookup address and port for lookup daemon