Getting Started with Hazelcast

Listening to Events in the Cluster

Grant Little
http://www.grantlittle.me
grant@grantlittle.me

Module Overview

- Entry Listeners (Map & MultiMap)
- Continuous Queries - Entry Listeners with filtering
- Item Listeners (Set, List, Queue)
- Partition Lost Listeners
- Other Listeners

Entry Listener

Map

MultiMap

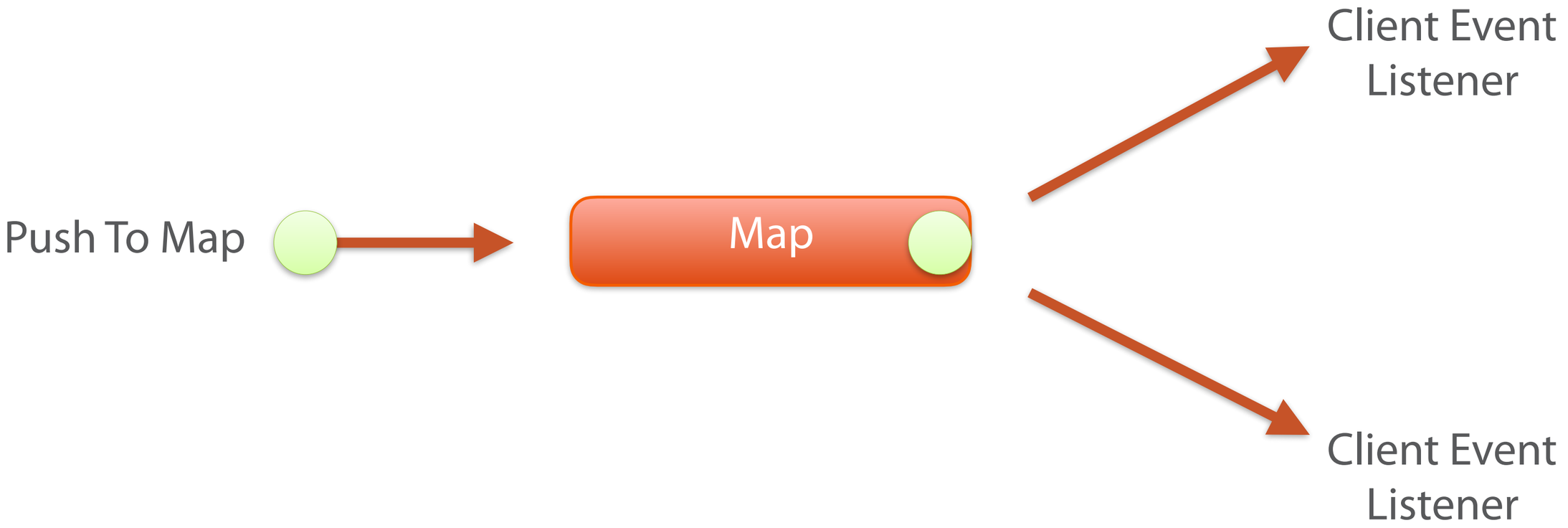
- EntryAddedListener
- EntryUpdatedListener
- EntryRemovedListener

EntryListener Example

EntryEvent

	EntryAdded	*EntryUpdated*	*EntryRemoved*
getValue()	*Latest Value in Map*	*Latest Value in Map*	*Null*
getOldValue()	*Null*	*Previous Value In Map*	*Previous Value In Map*

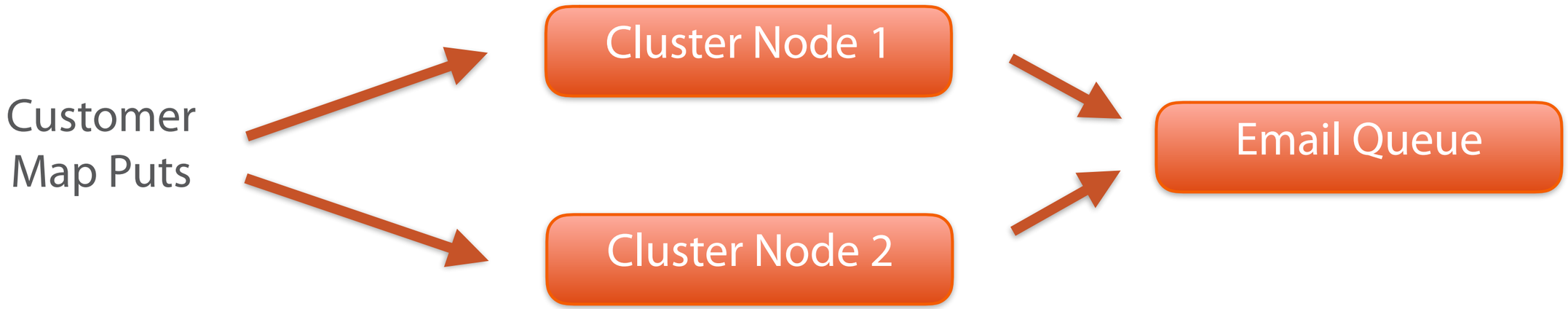
Global Event Listeners



```
ILock lock = hazelcastInstance.getLock(“CustomerEmailGenerator”);
try {
      lock.lock();
      hazelcastInstance.getMap(“customers”).addEntryListener(this, true);
      //Block thread until EntryListener is removed
} finally {
      lock.unlock();
}
```

Single Cluster Map Listener

Local Event Listeners



```
public class MyEntryListener implements com.hazelcast.core.EntryListener,
HazelcastInstanceAware {
      HazelcastInstance hazelcastInstance;
      public void setHazelcastInstance(HazelcastInstance instance) {
          this.instance = instance
      }
      //Implement the EntryListener Interface
}
```

Installing EntryListener via Config - Step 1

```
MapConfig mapConfig = new MapConfig(“name-of-map);
EntryListenerConfig entryListenerConfig = new EntryListenerConfig();
entryListenerConfig.setImplementation(new MyEntryListener());
entryListener.setLocal(true);
mapConfig.addEntryListenerConfig(entryListenerConfig);
```

Installing EntryListener via Config - Step 2

When creating Hazelcast Config object on storage node

Other Entry Listeners

- EntryEvictedListener
- MapClearedListener
- MapEvictedListener

```
IMap<Long, Customer> map = hazelcastInstance.getMap(“customers”);

Long key = 1L;

map.addEntryListener(listener, key, true);
```

Continuous Query

Key based continuous query

```
IMap<Long, Customer> map = hazelcastInstance.getMap(“customers”);

SqlPredicate predicate

     = new SqlPredicate(“email LIKE %@pluralsight.com);

map.addEntryListener(listener, predicate, true);
```

Continuous Query

SqlPredicate based continuous query

```
String ref = map.addEntryListener(listener, predicate, true);
// To remove the entry listener at a later stage you need the
// ref value
map.removeEntryListener(ref);
```

Removing EntryListeners

Item Listeners

Set
(ISet)

Queue
(IQueue)

List
(IList)

PartitionLostListener

```
MapConfig mapConfig = new MapConfig(“customers”);
//Increase the number of synchronous backups of the data
mapConfig.setBackupCount(3);
//Or increase the number of asynchronous backups of the data
//mapConfig.setAsyncBackupCount(3);
```

Increasing Backup Counts

Increase the number of backup copies of your data

Other Event Listeners

- Membership Listeners
- Distributed Object Listeners
- Migration Listeners
- Lifecycle Listeners
- Client Listeners

http://docs.hazelcast.org/docs/3.5/manual/html/distributedevents.html

Module Overview

- Entry Listeners (Map & MultiMap)
- Continuous Queries - Entry Listeners with filtering
- Item Listeners (Set, List, Queue)
- Partition Lost Listener
- Other Listeners

Next - Serialisation and Network Performance