

Getting Started with Hazelcast

Debugging & Monitoring



Grant Little

<http://www.grantlittle.me>

grant@grantlittle.me

Module Overview



Debugging

Monitoring

Possible Issues



Functional
Issues

Cluster,
Scale &
Performance

Test Cases

```
Config config = new Config()  
HazelcastInstance instance1 = Hazelcast.newHazelcastInstance(config);  
HazelcastInstance instance2 = Hazelcast.newHazelcastInstance(config);  
  
instance1.getMap("customers").put("customer1", "Mr Customer One");  
assertEquals(1, instance2.getMap("customers").size());
```

Test Cases

- Multiple Hazelcast instances allowed per JVM - small cluster
 - Test functional requirements
 - Provides a level of integration testing
- Tests are standalone and can be run in a debug session
- Not suitable for large amounts of data

Remote Debugging

```
-agentlib:jdwp=transport=dt_socket,server=y,  
suspend=n,address=<port number>
```

```
Config config = new Config();  
config.getProperties()  
.setProperty("hazelcast.logging.type", "jdk");
```

Logging - Options

jdk	slf4j
log4j	none

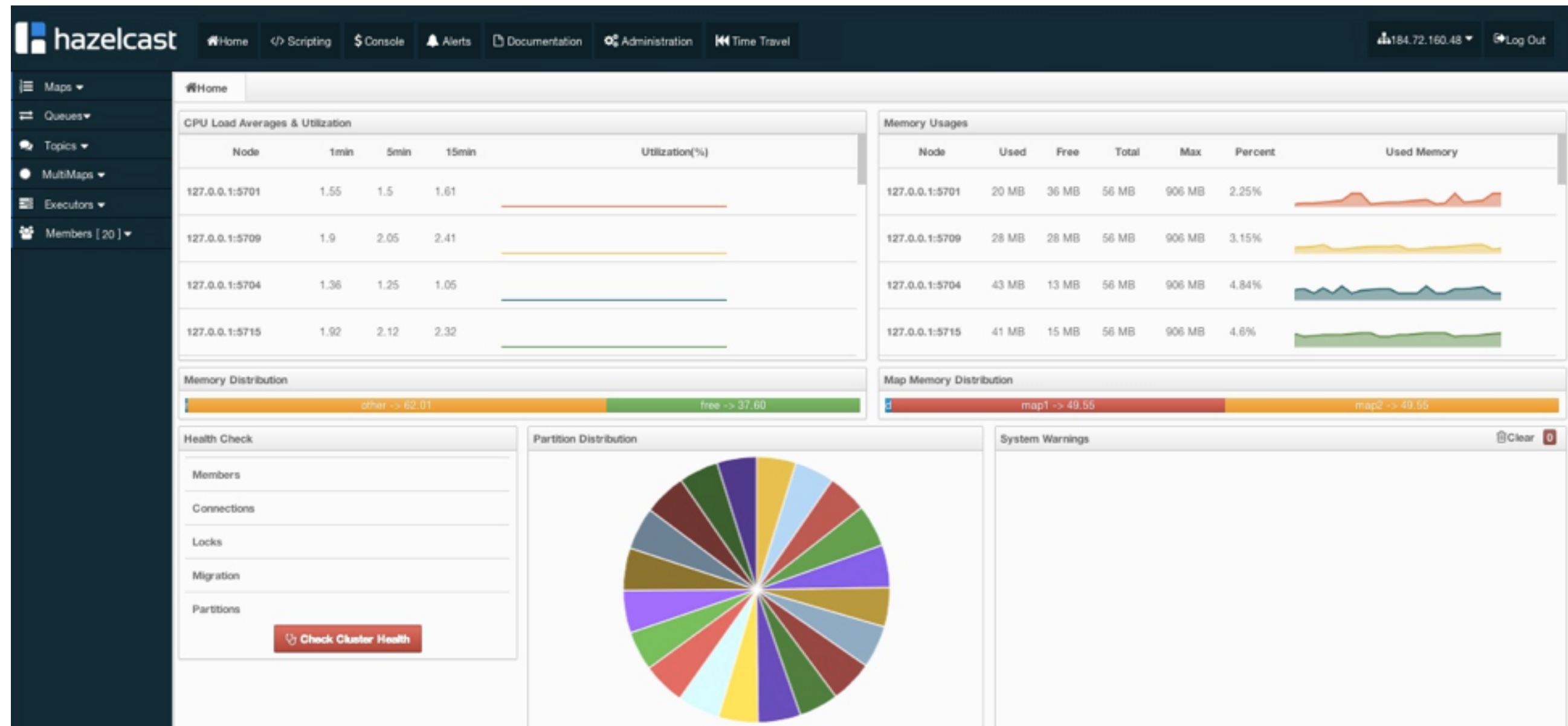
Third Party Applications

- Profiling
 - YourKit
 - JProfiler

Monitoring

Monitoring - Management Centre

(Enterprise Edition)



Per Node JMX

- Monitors Single Node
 - Cluster wide information is node available
- Lots of information
 - Queues
 - Maps
 - Partitions
- Available under the `com.hazelcast` MBean root

Possible JMX Clients

- JConsole (provided with the Oracle JDK)
- JVisualVM (provided with the Oracle JDK)
- Jolokia enables JMX via REST

Enabling Per Node JMX

```
-Dcom.sun.management.jmxremote  
-Dcom.sun.management.jmxremote.port=<port number>  
-Dcom.sun.management.jmxremote.local.only=false  
-Dcom.sun.management.jmxremote.authenticate=false  
-Dcom.sun.management.jmxremote.ssl=false  
-Dhazelcast.jmx=true
```

Module Review

- Debugging
 - Test Cases (in a single JVM)
 - Remote Debugging
 - Logging
- Monitoring
 - JMX
 - JConsole
 - JVisualVM

The End!
