Apache Geode安装及使用

1. 下载及解压

*wget <https://mirrors.tuna.tsinghua.edu.cn/apache/geode/1.6.0/apache-geode-1.6.0.zip>*

*unzip apache-geode-1.6.0.zip -C /opt*

*cd /opt/apache-geode-1.6.0*

1. 配置环境变量

*vim /etc/profile*

*export JAVA\_HOME=/usr/jdk64/jdk1.8.0\_77*

*export EGODE\_HOME=/opt/apache-geode-1.6.0*

*export PAHT=$PATH:$jJAVA\_HOME/bin:$EGODE\_HOME/bin*

测试:

*# gfsh version*

*1.6.0*

gfsh是命令行界面，用于运行、管理、监控geode进程、数据和应用：

* locator，JMX管理器，监测和管理geode集群
* geode服务器，用于存放长期存放的数据区域
* pulse，Web应用，基于浏览器的Pulse监控工具，提供图形化的监控界面，监控geode集群、成员、数据区域的重要的健康状态和性能。目前不支持Linux

1. 启动Locator

*gfsh>start locator --name=locator1*

*Starting a Geode Locator in /opt/apache-geode-1.6.0/config/locator1...*

*.....*

*Successfully connected to: JMX Manager [host=fys1.cmss.com, port=1099]*

*Cluster configuration service is up and running.*

*gfsh>*

查看进程如下：

*# jps*

*4998 LocatorLauncher*

*# netstat -anp|grep 4998*

*tcp6 0 0 :::1099 :::\* LISTEN 4998/java*

*tcp6 0 0 :::7070 :::\* LISTEN 4998/java*

*tcp6 0 0 :::10334 :::\* LISTEN 4998/java*

1. 启动Server

*gfsh>start server --name=server1 --server-port=40411*

*Starting a Geode Server in /opt/apache-geode-1.6.0/config/server1...*

查看进程：

*# jps*

*5337 ServerLauncher*

*# netstat -anp|grep 5337*

*tcp6 0 0 10.139.4.98:55606 :::\* LISTEN 5337/java*

*tcp6 0 0 :::40411 :::\* LISTEN 5337/java*

1. 创建Replication、持久化区域

region（区域）是geode集群的核心基石并提供管理数据的方法，该区域使用复制功能在集群成员中复制数据，并使用持久化数据保存到磁盘

* 创建复制、持久区域

*gfsh>create region --name=regionA --type=REPLICATE\_PERSISTENT*

*Member | Status*

*------- | --------------------------------------*

*server1 | Region "/regionA" created on "server1"*

regionA存放到了server1上

* 查看集群中的区域

*gfsh>list regions*

*List of regions*

*---------------*

*regionA*

* 列出集群上的成员，其开启的locator和缓存服务器出现在列表中

*gfsh>list members*

*Name | Id*

*-------- | -------------------------------------------------------------*

*locator1 | 10.139.4.98(locator1:4998:locator)<ec><v0>:1024 [Coordinator]*

*server1 | 10.139.4.98(server1:5337)<v1>:1025*

* 查看区域说明

*gfsh>describe region --name=regionA*

*..........................................................*

*Name : regionA*

*Data Policy : persistent replicate*

*Hosting Members : server1*

*Non-Default Attributes Shared By Hosting Members*

*Type | Name | Value*

*------ | ----------- | --------------------*

*Region | data-policy | PERSISTENT\_REPLICATE*

*| size | 0*

*| scope | distributed-ack*

1. 操作区域中的数据及持久化

Geode以键值方式管理数据，在大多数应用程序中，Java应用添加、删除和修改保存的数据，也可以用gfsh命令来添加或者查询

* 使用put命令将数据添加到区域中

*gfsh>put --region=regionA --key="1" --value="one"*

*gfsh>put --region=regionA --key="2" --value="two"*

* 查询数据

*gfsh>query --query="select \* from /regionA"*

*Result : true*

*Limit : 100*

*Rows : 2*

*Result*

*------*

*two*

*one*

1. 停止缓存服务器，数据会持久化磁盘中

*gfsh>stop server --name=server1*

*Stopping Cache Server running in /opt/apache-geode-1.6.0/config/server1 on fys1.cmss.com[40411] as server1...*

*Process ID: 5337*

*Log File: /opt/apache-geode-1.6.0/config/server1/server1.log*

1. 重新启动后，再次查询

*gfsh>start server --name=server1 --server-port=40411*

*Starting a Geode Server in /opt/apache-geode-1.6.0/config/server1..*

*gfsh>query --query="select \* from /regionA"*

*Result : true*

*Limit : 100*

*Rows : 2*

*Result*

*------*

*two*

*one*

因为regionA使用了持久化，将数据写入到了磁盘，当存有regionA的服务器启动，数据将在缓存中

1. 检查复制效果

开启了第二台缓存服务器，因为regionA是可复制的，数据将在任何一台存有RegionA区域的服务器中可用

* 开启第二台缓存服务器

*gfsh>start server --name=server2 --server-port=40412*

*Starting a Geode Server in /opt/apache-geode-1.6.0/config/server2...*

* 运行describe region查看regionA信息

*gfsh>describe region --name=regionA*

*..........................................................*

*Name : regionA*

*Data Policy : persistent replicate*

*Hosting Members : server2*

*server1*

*Non-Default Attributes Shared By Hosting Members*

*Type | Name | Value*

*------ | ----------- | --------------------*

*Region | data-policy | PERSISTENT\_REPLICATE*

*| size | 2*

*| scope | distributed-ack*

当gfsh启动服务器时，从集群配置服务请配置，然后分发共享的配置到任何一台加入到集群的新服务器

* 添加第三方数据，并查询

*gfsh>put --region=regionA --key="3" --value="three"*

*Result : true*

*Key Class : java.lang.String*

*Key : 3*

*Value Class : java.lang.String*

*Old Value : <NULL>*

*gfsh>query --query="select \* from /regionA"*

*Result : true*

*Limit : 100*

*Rows : 3*

*Result*

*------*

*two*

*one*

*three*

* 停止一台服务器，从剩余的服务器中查询数据

*gfsh>stop server --name=server1*

*Stopping Cache Server running in /opt/apache-geode-1.6.0/config/server1 on fys1.cmss.com[40411] as server1...*

*Process ID: 5911*

*Log File: /opt/apache-geode-1.6.0/config/server1/server1.log*

*...*

*gfsh>query --query="select \* from /regionA"*

*Result : true*

*Limit : 100*

*Rows : 3*

*Result*

*------*

*two*

*one*

*three*

1. 在第二个命令行窗口，切换到工作目录并且启动gfsh

*gfsh>connect --locator=10.139.4.98[10334]*

*Connecting to Locator at [host=10.139.4.98, port=10334] ..*

*Connecting to Manager at [host=cmhhost4.novalocal, port=1099] ..*

*Successfully connected to: [host=cmhhost4.novalocal, port=1099]*

*gfsh>start server --name=server1 --server-port=40411*

*gfsh>list members //查看成员*

*Name | Id*

*-------- | -------------------------------------------------------------*

*server1 | 10.139.4.94(server1:20031)<v6>:1024*

*locator1 | 10.139.4.98(locator1:4998:locator)<ec><v0>:1024 [Coordinator]*

*server2 | 10.139.4.98(server2:6230)<v4>:1026*

1. 在第二台集群中查询数据

*gfsh>query --query="select \* from /regionA"*

*Result : true*

*Limit : 100*

*Rows : 3*

*Result*

*------*

*two*

*one*

*three*

1. 在第一台集群中添加数据

*gfsh>put --region=regionA --key="5" --value="five"*

1. 在第二台机器上查询

*gfsh>query --query="select \* from /regionA"*

*Result : true*

*Limit : 100*

*Rows : 4*

*Result*

*------*

*two*

*five*

*one*

*three*

数据完成了同步，至此简单介绍了Geode的安装和使用。

1. 通过Java API访问Geode，程序如下：

*public class HelloGeode {  
 public static void main(String[] args) {  
 ClientCache cache = new ClientCacheFactory()  
 .addPoolLocator("10.139.4.98",10334)  
 .create();  
 Region<String,String> region = cache.<String,String>createClientRegionFactory(  
 ClientRegionShortcut.CACHING\_PROXY)  
 .create("regionA");  
 region.put("Proxy","Hello Client");  
 for(Map.Entry<String,String> entry : region.entrySet()) {  
 System.out.format("key = %s, value=%s \n", entry.getKey(), entry.getValue());  
 }  
 cache.close();  
 }*

1. 通过客户端查询数据

*gfsh>query --query="select \* from /regionA"*

*Result : true*

*Limit : 100*

*Rows : 5*

*Result*

*------------*

*two*

*Hello Client //Java API 插入的数据*

*five*

*one*

*three*

1. 停止Geode集群

*gfsh>shutdown --include-locators=true*

*gfsh>exit*

*Exiting...*

*http://bewithme.iteye.com/blog/2317455*

*http://bewithme.iteye.com/blog/2318730*