# **akka学习教程(十三) akka分布式**



[快乐崇拜234](https://me.csdn.net/fgyibupi" \t "https://blog.csdn.net/liubenlong007/article/details/_blank) 2017-01-18 15:55:27 IMG_257 12237 IMG_258 收藏 5

分类专栏： [分布式+高并发](https://blog.csdn.net/fgyibupi/category_6604930.html" \t "https://blog.csdn.net/liubenlong007/article/details/_blank) [akka学习教程](https://blog.csdn.net/fgyibupi/category_9267949.html" \t "https://blog.csdn.net/liubenlong007/article/details/_blank) 文章标签： [akka](https://www.csdn.net/gather_26/MtzaMg1sMDQwNzMtYmxvZwO0O0OO0O0O.html" \t "https://blog.csdn.net/liubenlong007/article/details/_blank)

版权

[](https://blog.csdn.net/fgyibupi/category_10410717.html" \t "https://blog.csdn.net/liubenlong007/article/details/_blank)

**[分布式事务seata](https://blog.csdn.net/fgyibupi/category_10410717.html" \t "https://blog.csdn.net/liubenlong007/article/details/_blank)**

[详细介绍各种流行的分布式事务解决方案，并且以阿里seata为例，详解其实现原理，应用实战，与源码分析](https://blog.csdn.net/fgyibupi/category_10410717.html" \t "https://blog.csdn.net/liubenlong007/article/details/_blank)

[快乐崇拜234](https://blog.csdn.net/fgyibupi/category_10410717.html" \t "https://blog.csdn.net/liubenlong007/article/details/_blank)

**¥9.90**

订阅博主

# **akka系列文章目录**

* [akka学习教程(十四) akka分布式实战](http://blog.csdn.net/liubenlong007/article/details/54603296)
* [akka学习教程(十三) akka分布式](http://blog.csdn.net/liubenlong007/article/details/54601998)
* [akka学习教程(十二) Spring与Akka的集成](http://blog.csdn.net/liubenlong007/article/details/54601448)
* [akka学习教程(十一) akka持久化](http://blog.csdn.net/liubenlong007/article/details/54584303)
* [akka学习教程(十) agent](http://blog.csdn.net/liubenlong007/article/details/54576611)
* [akka学习教程(九) STM软件事务内存](http://blog.csdn.net/liubenlong007/article/details/54576412)
* [akka学习教程(八) Actor中的Future-询问模式](http://blog.csdn.net/liubenlong007/article/details/54575312)
* [akka学习教程(七) 内置状态转换Procedure](http://blog.csdn.net/liubenlong007/article/details/54574878)
* [akka学习教程(六) 路由器Router](http://blog.csdn.net/liubenlong007/article/details/54574064)
* [akka学习教程(五) inbox消息收件箱](http://blog.csdn.net/liubenlong007/article/details/54377820)
* [akka学习教程(四) actor生命周期](http://blog.csdn.net/liubenlong007/article/details/54093889)
* [akka学习教程(三) 不可变对象](http://blog.csdn.net/liubenlong007/article/details/54093794)
* [akka学习教程(二)HelloWord](http://blog.csdn.net/liubenlong007/article/details/53782971)
* [akka学习教程(一)简介](http://blog.csdn.net/liubenlong007/article/details/53782966)

本文内容主要来自于来自于官方文档   
- [Cluster Usage](http://doc.akka.io/docs/akka/2.4/java/cluster-usage.html)   
- [Cluster Specification](http://doc.akka.io/docs/akka/2.4/common/cluster.html)

# **akka集群概述**

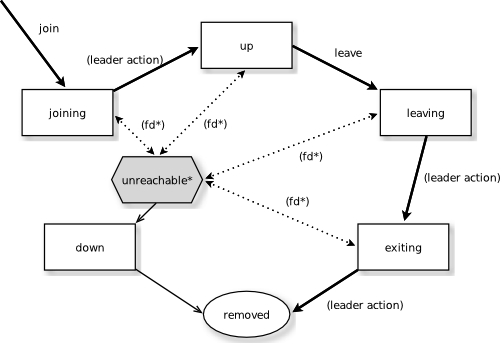
Akka群集提供容错分散的对等群集成员服务，没有单点故障或单点瓶颈。 它使用gossip协议和自动故障检测器。

## **术语**

* 节点   
  群集的逻辑成员。 一台物理机上可能有多个节点。 定义格式是【主机名：port：uid】
* 集群   
  通过成员服务连接在一起的一组节点。
* leader   
  集群中充当领导者的单个节点。 管理集群和成员状态转换。

## **节点状态**

* akka.cluster.allow-weakly-up-members=off 时：



* akka.cluster.allow-weakly-up-members=off 时：

IMG_262

* 状态（Member States）
  + joining： 假如集群的临时状态
  + weakly up： 出现网络分区时的临时状态（仅当akka.cluster.allow-weakly-up-members = on时）
  + up： 正常工作状态
  + leaving / exiting： 正常删除
  + down： marked as down (no longer part of cluster decisions)
  + removed： tombstone state (no longer a member)
* 动作（User Actions）
  + join： 一个节点加入到集群（can be explicit or automatic on startup if a node to join have been specified in the configuration）
  + leave： 优雅移除节点
  + down： 将一个节点标记为关闭
* leader动作（Leader Actions）
  + 领导人有以下职责：将成员移入和移出群集
    - 加入 -> 正常运行
    - 退出 -> 删除
* 故障检测和不可达性
  + fd \*
    - the failure detector of one of the monitoring nodes has triggered causing the monitored node to be marked as unreachable
  + unreachable\*
    - unreachable is not a real member states but more of a flag in addition to the state signaling that the cluster is unable to talk to this node, after being unreachable the failure detector may detect it as reachable again and thereby remove the flag

# **初始配置**

## **引入akka-cluster的maven依赖**

<dependency>

<groupId>com.typesafe.akka</groupId>

<artifactId>akka-cluster\_2.11</artifactId>

<version>2.4.16</version></dependency>

## **修改配置文件reference.conf**

akka {

loglevel = "INFO"

actor {

provider = "akka.cluster.ClusterActorRefProvider"

}

remote {

log-remote-lifecycle-events = off

netty.tcp {

hostname = "127.0.0.1"

port = 2551

}

}

cluster {

seed-nodes = [

"akka.tcp://akkaClusterTest@127.0.0.1:2551",

"akka.tcp://akkaClusterTest@127.0.0.1:2552"]

#//#snippet

# excluded from snippet

auto-down-unreachable-after = 10s

#//#snippet

# auto downing is NOT safe for production deployments.

# you may want to use it during development, read more about it in the docs.

#

auto-down-unreachable-after = 10s

# Disable legacy metrics in akka-cluster.

metrics.enabled=off

}

}

# 持久化相关

akka.persistence.journal.plugin = "akka.persistence.journal.inmem"# Absolute path to the default snapshot store plugin configuration entry.

akka.persistence.snapshot-store.plugin = "akka.persistence.snapshot-store.local"

## **编写测试代码（来自于自官网示例）**

package akka.myCluster;

import akka.actor.ActorRef;import akka.actor.ActorSystem;import akka.actor.Props;import akka.actor.UntypedActor;import akka.cluster.Cluster;import akka.cluster.ClusterEvent;import akka.cluster.ClusterEvent.MemberEvent;import akka.cluster.ClusterEvent.MemberUp;import akka.cluster.ClusterEvent.MemberRemoved;import akka.cluster.ClusterEvent.UnreachableMember;import akka.event.Logging;import akka.event.LoggingAdapter;import com.typesafe.config.ConfigFactory;

public class SimpleClusterListener extends UntypedActor {

LoggingAdapter log = Logging.getLogger(getContext().system(), this);

Cluster cluster = Cluster.get(getContext().system());

*//subscribe to cluster changes*

@Override

public void preStart() {

*//#subscribe*

cluster.subscribe(getSelf(), ClusterEvent.initialStateAsEvents(),

MemberEvent.class, UnreachableMember.class);

*//#subscribe*

}

*//re-subscribe when restart*

@Override

public void postStop() {

cluster.unsubscribe(getSelf());

}

@Override

public void onReceive(Object message) {

if (message instanceof MemberUp) {

MemberUp mUp = (MemberUp) message;

log.info("Member is Up: {}", mUp.member());

} else if (message instanceof UnreachableMember) {

UnreachableMember mUnreachable = (UnreachableMember) message;

log.info("Member detected as unreachable: {}", mUnreachable.member());

} else if (message instanceof MemberRemoved) {

MemberRemoved mRemoved = (MemberRemoved) message;

log.info("Member is Removed: {}", mRemoved.member());

} else if (message instanceof MemberEvent) {

*// ignore*

} else {

unhandled(message);

}

}

public static void main(String [] args){

System.out.println("Start simpleClusterListener");

ActorSystem system = ActorSystem.create("akkaClusterTest", ConfigFactory.load("reference.conf"));

system.actorOf(Props.create(SimpleClusterListener.class), "simpleClusterListener");

System.out.println("Started simpleClusterListener");

}

}

将以上项目复制一份，并修改其端口为2552

## **运行结果**

先启动2551服务：

Start simpleClusterListener

[INFO] [01/18/2017 15:39:14.886] [main] [akka.remote.Remoting] Starting remoting

[INFO] [01/18/2017 15:39:15.728] [main] [akka.remote.Remoting] Remoting started; listening on addresses :[akka.tcp://akkaClusterTest@127.0.0.1:2551]

[INFO] [01/18/2017 15:39:15.744] [main] [akka.cluster.Cluster(akka://akkaClusterTest)] Cluster Node [akka.tcp://akkaClusterTest@127.0.0.1:2551] - Starting up...

[INFO] [01/18/2017 15:39:15.853] [main] [akka.cluster.Cluster(akka://akkaClusterTest)] Cluster Node [akka.tcp://akkaClusterTest@127.0.0.1:2551] - Registered cluster JMX MBean [akka:type=Cluster]

[INFO] [01/18/2017 15:39:15.853] [main] [akka.cluster.Cluster(akka://akkaClusterTest)] Cluster Node [akka.tcp://akkaClusterTest@127.0.0.1:2551] - Started up successfully

Started simpleClusterListener

[WARN] [01/18/2017 15:39:15.900] [akkaClusterTest-akka.actor.default-dispatcher-18] [akka.tcp://akkaClusterTest@127.0.0.1:2551/system/cluster/core/daemon/downingProvider] Don't use auto-down feature of Akka Cluster in production. See 'Auto-downing (DO NOT USE)' section of Akka Cluster documentation.

[WARN] [01/18/2017 15:39:17.057] [akkaClusterTest-akka.remote.default-remote-dispatcher-8] [akka.tcp://akkaClusterTest@127.0.0.1:2551/system/endpointManager/reliableEndpointWriter-akka.tcp%3A%2F%2FakkaClusterTest%40127.0.0.1%3A2552-0] Association with remote system [akka.tcp://akkaClusterTest@127.0.0.1:2552] has failed, address is now gated for [5000] ms. Reason: [Association failed with [akka.tcp://akkaClusterTest@127.0.0.1:2552]] Caused by: [Connection refused: no further information: /127.0.0.1:2552]

[INFO] [01/18/2017 15:39:17.073] [akkaClusterTest-akka.actor.default-dispatcher-2] [akka://akkaClusterTest/deadLetters] Message [akka.cluster.InternalClusterAction$InitJoin$] from Actor[akka://akkaClusterTest/system/cluster/core/daemon/firstSeedNodeProcess-1#1750766313] to Actor[akka://akkaClusterTest/deadLetters] was not delivered. [1] dead letters encountered. This logging can be turned off or adjusted with configuration settings 'akka.log-dead-letters' and 'akka.log-dead-letters-during-shutdown'.

[INFO] [01/18/2017 15:39:17.073] [akkaClusterTest-akka.actor.default-dispatcher-2] [akka://akkaClusterTest/deadLetters] Message [akka.cluster.InternalClusterAction$InitJoin$] from Actor[akka://akkaClusterTest/system/cluster/core/daemon/firstSeedNodeProcess-1#1750766313] to Actor[akka://akkaClusterTest/deadLetters] was not delivered. [2] dead letters encountered. This logging can be turned off or adjusted with configuration settings 'akka.log-dead-letters' and 'akka.log-dead-letters-during-shutdown'.

[INFO] [01/18/2017 15:39:17.915] [akkaClusterTest-akka.actor.default-dispatcher-21] [akka://akkaClusterTest/deadLetters] Message [akka.cluster.InternalClusterAction$InitJoin$] from Actor[akka://akkaClusterTest/system/cluster/core/daemon/firstSeedNodeProcess-1#1750766313] to Actor[akka://akkaClusterTest/deadLetters] was not delivered. [3] dead letters encountered. This logging can be turned off or adjusted with configuration settings 'akka.log-dead-letters' and 'akka.log-dead-letters-during-shutdown'.

[INFO] [01/18/2017 15:39:18.914] [akkaClusterTest-akka.actor.default-dispatcher-17] [akka://akkaClusterTest/deadLetters] Message [akka.cluster.InternalClusterAction$InitJoin$] from Actor[akka://akkaClusterTest/system/cluster/core/daemon/firstSeedNodeProcess-1#1750766313] to Actor[akka://akkaClusterTest/deadLetters] was not delivered. [4] dead letters encountered. This logging can be turned off or adjusted with configuration settings 'akka.log-dead-letters' and 'akka.log-dead-letters-during-shutdown'.

[INFO] [01/18/2017 15:39:19.914] [akkaClusterTest-akka.actor.default-dispatcher-2] [akka://akkaClusterTest/deadLetters] Message [akka.cluster.InternalClusterAction$InitJoin$] from Actor[akka://akkaClusterTest/system/cluster/core/daemon/firstSeedNodeProcess-1#1750766313] to Actor[akka://akkaClusterTest/deadLetters] was not delivered. [5] dead letters encountered. This logging can be turned off or adjusted with configuration settings 'akka.log-dead-letters' and 'akka.log-dead-letters-during-shutdown'.

[INFO] [01/18/2017 15:39:21.008] [akkaClusterTest-akka.actor.default-dispatcher-19] [akka.cluster.Cluster(akka://akkaClusterTest)] Cluster Node [akka.tcp://akkaClusterTest@127.0.0.1:2551] - Node [akka.tcp://akkaClusterTest@127.0.0.1:2551] is JOINING, roles []

[INFO] [01/18/2017 15:39:21.024] [akkaClusterTest-akka.actor.default-dispatcher-19] [akka.cluster.Cluster(akka://akkaClusterTest)] Cluster Node [akka.tcp://akkaClusterTest@127.0.0.1:2551] - Leader is moving node [akka.tcp://akkaClusterTest@127.0.0.1:2551] to [Up]

[INFO] [01/18/2017 15:39:21.024] [akkaClusterTest-akka.actor.default-dispatcher-2] [akka://akkaClusterTest/user/simpleClusterListener] Member is Up: Member(address = akka.tcp://akkaClusterTest@127.0.0.1:2551, status = Up)

此时再启动2552端口,2552的输出日志为：

Start simpleClusterListener

[INFO] [01/18/2017 15:40:35.671] [main] [akka.remote.Remoting] Starting remoting

[INFO] [01/18/2017 15:40:36.458] [main] [akka.remote.Remoting] Remoting started; listening on addresses :[akka.tcp://akkaClusterTest@127.0.0.1:2552]

[INFO] [01/18/2017 15:40:36.474] [main] [akka.cluster.Cluster(akka://akkaClusterTest)] Cluster Node [akka.tcp://akkaClusterTest@127.0.0.1:2552] - Starting up...

[INFO] [01/18/2017 15:40:36.552] [main] [akka.cluster.Cluster(akka://akkaClusterTest)] Cluster Node [akka.tcp://akkaClusterTest@127.0.0.1:2552] - Registered cluster JMX MBean [akka:type=Cluster]

[INFO] [01/18/2017 15:40:36.552] [main] [akka.cluster.Cluster(akka://akkaClusterTest)] Cluster Node [akka.tcp://akkaClusterTest@127.0.0.1:2552] - Started up successfully

Started simpleClusterListener

[WARN] [01/18/2017 15:40:36.567] [akkaClusterTest-akka.actor.default-dispatcher-15] [akka.tcp://akkaClusterTest@127.0.0.1:2552/system/cluster/core/daemon/downingProvider] Don't use auto-down feature of Akka Cluster in production. See 'Auto-downing (DO NOT USE)' section of Akka Cluster documentation.

[INFO] [01/18/2017 15:40:37.067] [akkaClusterTest-akka.actor.default-dispatcher-2] [akka.cluster.Cluster(akka://akkaClusterTest)] Cluster Node [akka.tcp://akkaClusterTest@127.0.0.1:2552] - Welcome from [akka.tcp://akkaClusterTest@127.0.0.1:2551]

[INFO] [01/18/2017 15:40:37.082] [akkaClusterTest-akka.actor.default-dispatcher-4] [akka://akkaClusterTest/user/simpleClusterListener] Member is Up: Member(address = akka.tcp://akkaClusterTest@127.0.0.1:2551, status = Up)

[INFO] [01/18/2017 15:40:37.925] [akkaClusterTest-akka.actor.default-dispatcher-15] [akka://akkaClusterTest/user/simpleClusterListener] Member is Up: Member(address = akka.tcp://akkaClusterTest@127.0.0.1:2552, status = Up)

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* 12

同时2551会增加几行日志：

[INFO] [01/18/2017 15:40:36.942] [akkaClusterTest-akka.actor.default-dispatcher-2] [akka.cluster.Cluster(akka://akkaClusterTest)] Cluster Node [akka.tcp://akkaClusterTest@127.0.0.1:2551] - Node [akka.tcp://akkaClusterTest@127.0.0.1:2552] is JOINING, roles []

[INFO] [01/18/2017 15:40:37.893] [akkaClusterTest-akka.actor.default-dispatcher-3] [akka.cluster.Cluster(akka://akkaClusterTest)] Cluster Node [akka.tcp://akkaClusterTest@127.0.0.1:2551] - Leader is moving node [akka.tcp://akkaClusterTest@127.0.0.1:2552] to [Up]

[INFO] [01/18/2017 15:40:37.893] [akkaClusterTest-akka.actor.default-dispatcher-4] [akka://akkaClusterTest/user/simpleClusterListener] Member is Up: Member(address = akka.tcp://akkaClusterTest@127.0.0.1:2552, status = Up)

* 1
* 2
* 3

上面日志中可以看到Akka集群中各个节点的状态迁移信息，第一个种子节点正在加入自身创建的集群时的状态时JOINING，由于第一个种子节点将自己率先选举为Leader，因此它还将自己的状态改变为Up。后面它还将第二个种子节点和第三个节点从JOINING转换到Up状态。

同样再添加一个2553端口的服务，与上面一样，就不多说了。   
我们现在吧2553服务停止，看看日志输出：

[akka.cluster.Cluster(akka://akkaClusterTest)] Cluster Node [akka.tcp://akkaClusterTest@127.0.0.1:2551] - Leader is auto-downing unreachable node [akka.tcp://akkaClusterTest@127.0.0.1:2553]. Don't use auto-down feature of Akka Cluster in production. See 'Auto-downing (DO NOT USE)' section of Akka Cluster documentation.

[akka.cluster.Cluster(akka://akkaClusterTest)] Cluster Node [akka.tcp://akkaClusterTest@127.0.0.1:2551] - Marking unreachable node [akka.tcp://akkaClusterTest@127.0.0.1:2553] as [Down]

[akka.cluster.Cluster(akka://akkaClusterTest)] Cluster Node [akka.tcp://akkaClusterTest@127.0.0.1:2551] - Leader is removing unreachable node [akka.tcp://akkaClusterTest@127.0.0.1:2553]

[akka://akkaClusterTest/user/simpleClusterListener] Member is Removed: Member(address = akka.tcp://akkaClusterTest@127.0.0.1:2553, status = Removed)

* 1
* 2
* 3
* 4

2553被标记为Removed

注意： 我们在配置文件中配置了auto-down-unreachable-after=10s。所以在2553关闭10s后才会真正将其移除。

# **参考资料**

* 书籍《java高并发程序设计》
* [AKKA官方文档](http://akka.io/docs/)