



APUSIC
固若长城
睿比世界

快速入门

中间件产品 - 金蝶Apusic分布式配置中心V1.0 for zookeeper

版权所有 © 深圳市金蝶天燕云计算股份有限公司2026。保留所有权利。

版权声明

本档所涉及的软件著作权、版权等知识产权已依法进行了注册，由金蝶天燕云计算股份有限公司合法拥有。受《中华人民共和国著作权法》《计算机软件保护条例》《知识产权保护条例》和相关国际版权条约、法律、法规以及其它知识产权法律和条约的保护。未经授权许可，不得非法使用。

免责声明

本档包含的版权信息由金蝶天燕云计算股份有限公司合法拥有，受法律的保护，金蝶天燕云计算股份有限公司对本档可能涉及到的非金蝶天燕云计算股份有限公司的信息不承担任何责任。在法律允许的范围内，您可以查阅并仅能够在《中华人民共和国著作权法》规定的合法范围内复制和打印本档。任何单位和个人未经金蝶天燕云计算股份有限公司书面授权许可，不得使用、修改、再发布本档的任何部分和内容，否则将被视为侵权，金蝶天燕云计算股份有限公司有依法追究其责任的权利。

本档如有更新，不另行通知。对本档中的问题您可向金蝶天燕云计算股份有限公司告知或查询。未经本公司明确授予的任何权利均予保留。

商标声明

 是深圳市金蝶天燕云计算股份有限公司向中华人民共和国国家商标局申请注册的注册商标，注册商标专用权由金蝶天燕合法拥有，受法律保护。未经金蝶天燕的书面许可，任何单位及个人不得以任何方式或理由对该商标的任何部分进行使用、复制、修改、传播、抄录或与其它产品捆绑使用销售。凡侵犯金蝶天燕商标权的，金蝶天燕将依法追究其法律责任。本档提及的其他所有商标或注册商标，由各自的所有人拥有。

目录

- 1 概述
 - 1.1 摘要
 - 1.2 基础介绍
 - 1.3 术语
- 2 系统环境要求
- 3 产品安装
- 4 单机模式操作
 - 4.1 启动
 - 4.2 查看状态
 - 4.3 停止
- 5 集群模式
 - 5.1 安装
 - 5.2 配置
 - 5.3 验证
- 6 连接到ADCC
- 7 产品授权
 - 7.1 获取特征码
 - 7.2 授权文件

1 概述

1.1 摘要

本快速入门指南主要介绍金蝶Apsic分布式配置中心V1.0 for zookeeper（简称“ADCC for zk”）基础使用过程，适用于使用金蝶Apsic分布式配置中心 for zookeeper产品进行开发的开发人员、生产环境的系统管理员、运维人员等。

1.2 基础介绍

金蝶Apsic分布式配置中心软件for zookeeper（Apsic Distributed Config Center for zookeeper，简称：ADCC for zk）为分布式应用系统提供高性能协调服务的产品，针对分布式技术而产生的一种新型的配置手段，通过简单的接口能够对分布式应用系统的配置进行统一的管理、实时更新，并支持配置更改的订阅，可实现不需要重启服务器而动态的修改配置文件的内容，有效提高工作效率。。

1.3 术语

ADCC_HOME：ADCC安装目录

2 系统环境要求

系统环境要求如下：

类型	要求
操作系统	国产操作系统如银河麒麟系列、中标麒麟系列、普华、中科红旗、深度等； Windows系列； Linux Red Hat 5.2或以上。
CPU	国产架构：鲲鹏、海光、飞腾、龙芯、兆芯等； 国外架构：如Intel等。
内存、硬盘	建议内存4G或以上；可用空间100G或以上
JDK	JDK8、JDK11、JDK12、JDK17稳定版本

3 产品安装

获取到稳定的ADCC for zk版本后，如 `ADCC-V1.0.371-Zookeeper-20241219.tar.gz`；拷贝到目录，如 `/opt`；执行解压，得到有目录 `adcc/`，此时 `/opt/adcc/` 就是 `${ADCC_HOME}`。

```
tar -zxvf ADCC-V1.0.371-Zookeeper-20241219.tar.gz
```

详看《金蝶Apusic分布式配置中心软件for zookeeper安装手册》。

若想卸载产品，将安装目录，如 `/opt/ADCC/adcc-server-1.0`，删除即完成卸载操作。注意，卸载前需确认产品处于停止运行状态。

4 单机模式操作

4.1 启动

在单机模式下设置ADCC for zk服务器非常简单。服务器包含在一个JAR文件中，因此安装包括创建配置。

启动前需要确认data目录，默认值为 `../data`。可在配置文件 `conf/zoo.cfg` 中配置：

```
tickTime=2000
dataDir=../data
clientPort=2181
```

更改dataDir的值以指定一个现有的（首先为空）目录。以下是每个字段的含义：

tickTime: ADCC for zk使用的基本时间单位（毫秒）。它用于控制 ADCC服务器的心跳机制和会话超时等多个重要功能的时间间隔。简单来说，`tickTime` 是 ADCC 进行各种时间相关操作的基本节拍。

dataDir: 存储内存中数据库快照的位置，除非另有规定，还存储数据库更新的事务日志。

clientPort: 侦听客户端连接的端口。

进入安装目录 `${ADCC_HOME}/bin`，执行启动命令，`adccServer.sh start`。

```
adccServer.sh start
```

ADCC for zk使用log4j记录消息。根据log4j配置，您将看到控制台（默认）和/或日志文件中的日志消息。

此处步骤以单机模式运行ADCC for zk，没有复制，因此如果ADCC for zk进程失败，服务将关闭。

4.2 查看状态

进入安装目录 `${ADCC_HOME}/bin`，执行查看状态命令，`adccServer.sh status`。

```
[user@test3 bin]# ./adccServer.sh status
ADCC JMX enabled by default
Using config: /opt/ADCC/adcc-server-1.0/bin/./conf/zoo.cfg
Client port found: 2181. Client address: localhost. Client SSL: false.
Mode: standalone
```

4.3 停止

进入安装目录 `${ADCC_HOME}/bin` , 执行停止命令, `adccServer.sh stop` 。

```
[user@test3 bin]# ./adccServer.sh stop
ADCC JMX enabled by default
Using config: /opt/ADCC/adcc-server-1.0/bin/./conf/zoo.cfg
Stopping ADCC ... STOPPED
```

5 集群模式

在单机模式下运行ADCC for zk便于评估、某些开发和测试。但在生产中，建议在集群模式下运行ADCC for zk。同一应用程序中的一组复制服务器称为仲裁，在集群模式下，仲裁中的所有服务器都具有相同配置文件的副本。

5.1 安装

获取到稳定的ADCC for zk版本后，如 `ADCC-V1.0.371-Zookeeper-20241219.tar.gz`；拷贝到目录，如 `/opt`；执行解压，得到有目录 `adcc/`，此时 `/opt/adcc/` 就是 `${ADCC_HOME}`。

```
tar -zxvf ADCC-V1.0.371-Zookeeper-20241219.tar.gz
```

在集群的每个节点服务器都需要安装ADCC；如果节点在同一服务器，需要注意端口不能冲突。

5.2 配置

集群模式的 `conf/zoo.cfg` 文件与单机模式中使用的类似，但有一些区别。下面以在172.20.140.17、172.20.140.13和172.20.12的机器部署集群为环境进行介绍，在每一个环境上ADCC for zk的 `zoo.cfg` 的集群配置文件，需要增加如下的最后3行（如果存在端口冲突，则需要修改端口）：

```
tickTime=2000
dataDir=../data
clientPort=2181
initLimit=10
syncLimit=5
server.1=172.20.140.17:2888:3888
server.2=172.20.140.13:4888:5888
server.3=172.20.140.12:6888:7888
```

表单的条目 `server.x` 列出组成ADCC for zk集群服务的服务器。

```
dataDir=./data
# the port at which the clients will connect
clientPort=2181
# the maximum number of client connections.
# increase this if you need to handle more clients
#maxClientCnxns=60
#
# Be sure to read the maintenance section of the
# administrator guide before turning on autopurge.
#
#
# The number of snapshots to retain in dataDir
autopurge.snapRetainCount=10
# Purge task interval in hours
# Set to "0" to disable auto purge feature
autopurge.purgeInterval=12

### Metrics Providers
#
# https://prometheus.io Metrics Exporter
#metricsProvider.className=org.apache.zookeeper.metrics.prometheus.PrometheusMetricsProvider
#metricsProvider.httpPort=7000
#metricsProvider.exportJvmInfo=true

#
# the port of adminserver,default is 8080
#admin.serverPort=8080
#
#the config of cluster,must create myid file in dataDir and set value respectively
server.1=172.20.140.17:2888:3888
server.2=172.20.140.13:4888:5888
server.3=172.20.140.12:6888:7888
~
~
```

当服务器启动时，它通过在数据目录中查找文件 `myid` 来知道它是哪个服务器。该文件包含服务器号 `x`，所以集群配置还需要如下的修改：

在172.20.140.12中的 `dataDir` 目录创建 `myid` 文件，即在 `../data` 下创建 `myid` 文件，并输入上面配置中 `server.1` 中的序号 `1`，并保存；

同理在172.20.140.17中的 `dataDir` 目录创建 `myid` 文件，即在 `../data` 下创建 `myid` 文件，并输入上面配置中 `server.2` 中的序号 `2`，并保存；

同理在172.20.140.43中的 `dataDir` 目录创建 `myid` 文件，即在 `../data` 下创建 `myid` 文件，并输入上面配置中 `server.3` 中的序号 `3`，并保存；

如下图为 `server.1` 服务器的 `myid` 配置：

```
[root@myRabbitA data]# vim myid
[root@myRabbitA data]# ls
myid  version-2
[root@myRabbitA data]# cat myid
1
```

到这里，3个节点的集群就已经配置完成，然后分别启动即可。注意：在3个节点未完全启动完前，会答应连接不上的信息，这是正常的。

上面配置的其他说明：

initLimit: ADCC for zk用来限制quorum中ADCC for zk服务器必须连接到领导者的时间长度的超时。条目syncLimit限制了服务器与Leader之间的过期距离。

对于这两种超时，您可以使用tickTime指定时间单位。在本例中，initLimit的超时为5次，每次为2000毫秒，即10秒。

最后，请注意每个服务器名称后面的两个端口号，如"2888"和"3888"。对等体使用前一个端口连接到其他对等体。这样的连接是必要的，以便对等体可以进行通信，例如商定更新的顺序。更具体地说，ADCC for zk服务器使用此端口将follower连接到Leader。当出现新的Leader时，follower使用此端口打开与Leader的TCP连接。由于默认Leader选举也使用TCP，因此我们当前需要另一个端口进行Leader选举。这是服务器条目中的第二个端口。

5.3 验证

依次启动ADCC集群节点。

```
./adccServer.sh start
```

查看ADCC节点状态。

```
./adccServer.sh status
```

下图为示例参考图，主要查看Mode属性值和运行状态，其他值可能会与实际略不同。

可以看到有一个是leader，两个是follower。

```
[root@tck-server bin]# ./adccServer.sh status
ADCC JMX enabled by default
Using config: /opt/ADCC/adcc-server-1.0/bin/./conf/zoo.cfg
Client port found: 2181. Client address: localhost. Client SSL: false.
Mode: leader
[root@tck-server bin]#
```

```
[root@test3 bin]# ./adccServer.sh status
ADCC JMX enabled by default
Using config: /opt/ADCC/adcc-server-1.0/bin/./conf/zoo.cfg
Client port found: 2181. Client address: localhost. Client SSL: false.
Mode: follower
[root@test3 bin]#
```

```
[root@myRabbitA bin]# ./adccServer.sh status
ADCC JMX enabled by default
Using config: /opt/ADCC/adcc-server-1.0/bin/./conf/zoo.cfg
Client port found: 2181. Client address: localhost. Client SSL: false.
Mode: follower
[root@myRabbitA bin]#
```

先把其中一个follower的节点停止，查看其他节点的状态。可以看到其他节点的状态依旧正常，集群可以正常运行。

```
[root@myRabbitA bin]# ./adccServer.sh stop
ADCC JMX enabled by default
Using config: /opt/ADCC/adcc-server-1.0/bin/./conf/zoo.cfg
Stopping ADCC ... STOPPED
[root@myRabbitA bin]# ./adccServer.sh status
ADCC JMX enabled by default
Using config: /opt/ADCC/adcc-server-1.0/bin/./conf/zoo.cfg
Client port found: 2181. Client address: localhost. Client SSL: false.
Error contacting service. It is probably not running.
[root@myRabbitA bin]#
```

```
[root@tck-server bin]# ./adccServer.sh status
ADCC JMX enabled by default
Using config: /opt/ADCC/adcc-server-1.0/bin/./conf/zoo.cfg
Client port found: 2181. Client address: localhost. Client SSL: false.
Mode: leader
[root@tck-server bin]#
```

```
[root@test3 bin]# ./adccServer.sh status
ADCC JMX enabled by default
Using config: /opt/ADCC/adcc-server-1.0/bin/./conf/zoo.cfg
Client port found: 2181. Client address: localhost. Client SSL: false.
Mode: follower
[root@test3 bin]#
```

再停止一个节点，查看结果。可以看到节点停止后，此时运行的节点少于集群总数量的一半，剩下的一个节点也停止了。

停止运行的节点：

```
[root@tck-server bin]# ./adccServer.sh stop
ADCC JMX enabled by default
Using config: /opt/ADCC/adcc-server-1.0/bin/./conf/zoo.cfg
Stopping ADCC ... STOPPED
[root@tck-server bin]# ./adccServer.sh status
ADCC JMX enabled by default
Using config: /opt/ADCC/adcc-server-1.0/bin/./conf/zoo.cfg
Client port found: 2181. Client address: localhost. Client SSL: false.
Error contacting service. It is probably not running.
[root@tck-server bin]#
```

没有执行停止运行命令的节点：

```
[root@test3 bin]# ./adccServer.sh status
ADCC JMX enabled by default
Using config: /opt/ADCC/adcc-server-1.0/bin/./conf/zoo.cfg
Client port found: 2181. Client address: localhost. Client SSL: false.
Error contacting service. It is probably not running.
[root@test3 bin]#
```

上一步停止运行的节点：

```
[root@myRabbitA bin]# ./adccServer.sh status
ADCC JMX enabled by default
Using config: /opt/ADCC/adcc-server-1.0/bin/./conf/zoo.cfg
Client port found: 2181. Client address: localhost. Client SSL: false.
Error contacting service. It is probably not running.
[root@myRabbitA bin]#
```

再把其中一个已停止的节点启动，查看结果。会看到集群的节点运行数量过半后，集群又恢复正常运行状态，并且会重新推举出一个leader。

执行启动命令的节点：

```
[root@tck-server bin]# ./adccServer.sh start
ADCC JMX enabled by default
Using config: /opt/ADCC/adcc-server-1.0/bin/./conf/zoo.cfg
Starting ADCC ... STARTED
[root@tck-server bin]# ./adccServer.sh status
ADCC JMX enabled by default
Using config: /opt/ADCC/adcc-server-1.0/bin/./conf/zoo.cfg
Client port found: 2181. Client address: localhost. Client SSL: false.
Mode: follower
[root@tck-server bin]#
```

没有执行停止命令的节点：

```
[root@test3 bin]# ./adccServer.sh status
ADCC JMX enabled by default
Using config: /opt/ADCC/adcc-server-1.0/bin/./conf/zoo.cfg
Client port found: 2181. Client address: localhost. Client SSL: false.
Mode: leader
[root@test3 bin]#
```

此时再加入节点也是follower节点，不会影响到现任的leader节点。

6 连接到ADCC

进入安装目录 `${ADCC_HOME}/bin` , 执行连接命令, `adccCli.sh -server IP:clientPort` 。

```
\$ adccCli.sh -server 127.0.0.1:2181
```

7 产品授权

ADCC需要有对应的许可证才能正常使用，通常情况下，金蝶天燕会根据用户购买的产品版本配套对应的许可证。

许可证位置为 `${ADCC_HOME}/license.xml`。

7.1 获取特征码

如果在使用过程中出现许可证过期或无效等问题，建议优先联系对接的天燕服务人员，重新申请对应许可证。重新申请对应许可证时，需要将产品的特征码(auth code)提供到天燕对接人员。

在 `${ADCC_HOME}/bin`，执行 `adccServer.sh -ac host`，host取值为ip地址或者网卡名称，类似如下：

```
./adccServer.sh -ac 172.20.140.17
```

打印特征码信息，类似如下，Auth Code=[特征码内容]：

```
ADCC JMX enabled by default  
Auth Code=SZTY777387783
```

获取特征码后再提供特征码申请授权文件。

7.2 授权文件

获取到授权文件后，命名为 `license.xml`，放置ADCC安装目录下即可。

全国统一服务热线
4008-555-800



金蝶天燕云计算股份有限公司(简称“金蝶天燕云”)成立于2000年,前身为“金蝶中间件公司”,是金蝶集团旗下新一代软件基础云平台服务商,云计算国家标准制定企业,国家信创产业核心软件企业。金蝶天燕是国家863重点研发计划与核高基重大专项承接企业,也是“两网一站四库十二金”国家重点工程的基础平台提供商,产品广泛应用于政府、军工、金融、能源等关键行业,累计服务客户总数超过10万家。

Apusic
金蝶天燕

云计算国家标准制定企业
金蝶集团旗下基础软件企业
信息技术应用创新核心企业
官网: www.apusic.com

