



# Zookeeper分布式系统开发实战 第10课

**【声明】** 本视频和幻灯片为炼数成金网络课程的教学资料，所有资料只能在课程内使用，不得在课程以外范围散播，违者将可能被追究法律和经济责任。

课程详情访问炼数成金培训网站

<http://edu.dataguru.cn>

- Dataguru ( 炼数成金 ) 是专业数据分析网站 , 提供教育 , 媒体 , 内容 , 社区 , 出版 , 数据分析业务等服务。我们的课程采用新兴的互联网教育形式 , 独创地发展了逆向收费式网络培训课程模式。既继承传统教育重学习氛围 , 重竞争压力的特点 , 同时又发挥互联网的威力打破时空限制 , 把天南地北志同道合的朋友组织在一起交流学习 , 使到原先孤立的学习个体组合成有组织的探索力量。并且把原先动辄成千上万的学习成本 , 直线下降至百元范围 , 造福大众。我们的目标是 : 低成本传播高价值知识 , 构架中国第一的网上知识流转阵地。
- 关于逆向收费式网络的详情 , 请看我们的培训网站 <http://edu.dataguru.cn>

## ■ Zookeeper重要健康指标

### — 物理资源

- Cpu 、内存、 磁盘空间、 磁盘IO

### — 应用资源

- 连接数
- 注册的Watcher数
- ZNode是否可读，可写（ ACL ）
- ZK事件通知的延时是否过大
- Zookeeper的节点健康（ 是否宕机 ）
- 数据状态
- 请求延迟状态
- 节点角色

## ■ Taookeeper

- 阿里的开源产品
- 主要功能
  - **CPU/MEM/LOAD**的监控
  - **ZK日志目录**所在磁盘剩余空间监控
  - **单机连接数**的峰值报警
  - **单机 Watcher数**的峰值报警
  - 节点健康状态检测
  - 少量统计报表
- 不足：
  - 缺少数据目录查看
  - 缺少磁盘iops统计
  - 缺少网络数据统计

# 第十讲 Taookeeper

## ■ 安装部署

### — 源码安装

- <https://github.com/alibaba/taookeeper>

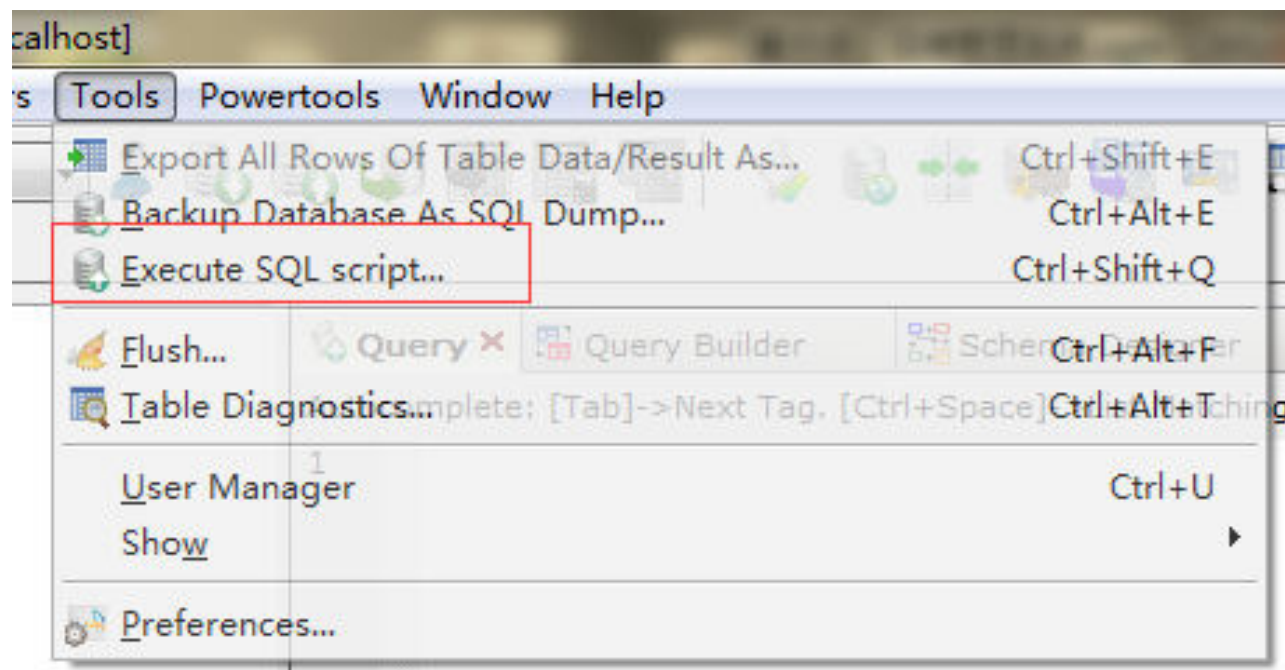
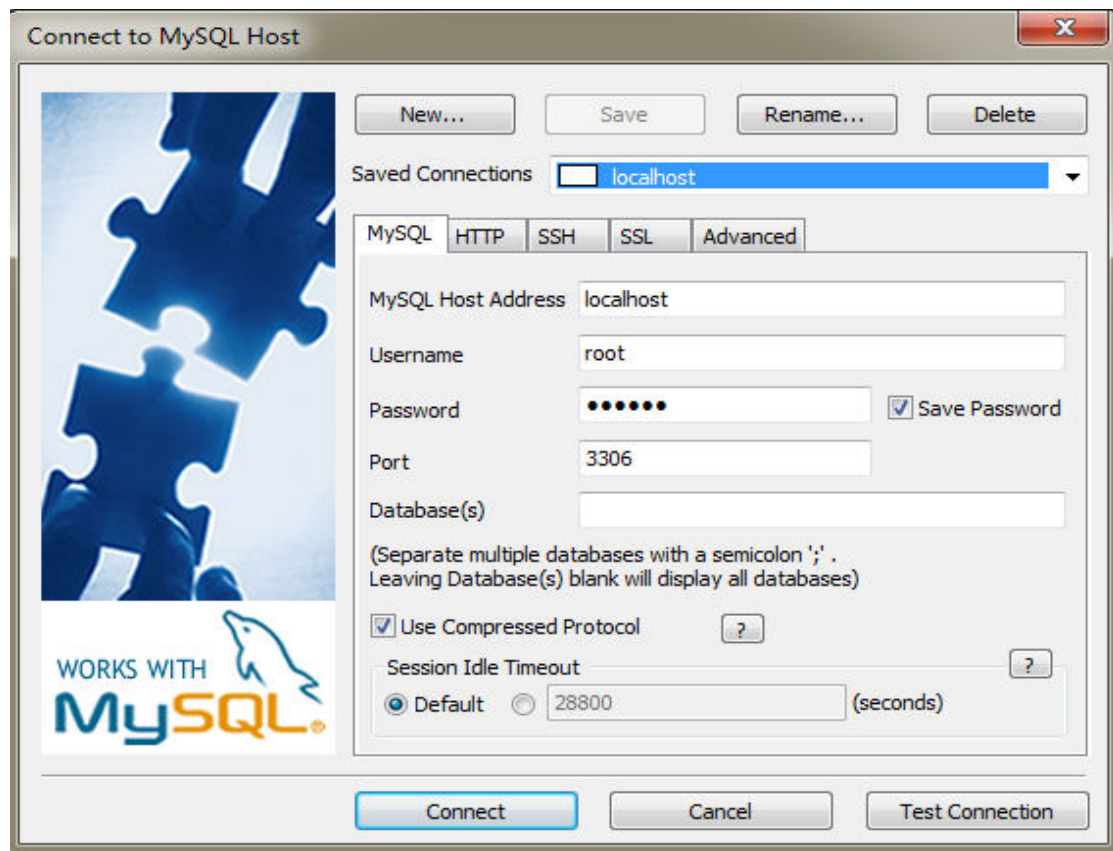
### — 包安装

- 下载初始化脚本：<http://vdisk.weibo.com/s/CQ-NI7qhaSR0>
- 下载部署包：<http://vdisk.weibo.com/s/CQ-NI7qMzVNx>
- 下载配置文件模板：<http://vdisk.weibo.com/s/CQ-NI7qhaSzS>

# 第十讲 Taookeeper

## ■ 包安装步骤

- 导入mysql脚本taookeeper.sql ( mysql工具SQLyog )



# 第十讲 Taookeeper

## ■ 包安装步骤

- 配置相关属性，通过配置文件：taookeeper-monitor-config.properties

```
systemInfo.envName=DAILY-mysql
```

```
#DBCP
```

```
dbcp.driverClassName=com.mysql.jdbc.Driver
```

```
dbcp.dbJDBCUrl=jdbc:mysql://127.0.0.1:3306/taookeeper
```

```
dbcp.characterEncoding=GBK
```

```
dbcp.username=root
```

```
dbcp.password=111111
```

配置连接池

```
dbcp.maxActive=30
```

```
dbcp.maxIdle=10
```

```
dbcp.maxWait=10000
```

```
#SystemConstant
```

监控程序本地机器临时存储数据的目录

```
SystemConstant.dataStoreBasePath=D:\\soft\\apache-tomcat-7.0.26-new\\apache-tomcat-7.0.26\\taookeeperData
```

```
SystemConstant.userNameOfSSH=root
```

```
SystemConstant.passwordOfSSH=111111
```

配置远程zk服务器的ssh用户名和密码，如果zk部署在多个物理服务器上，则用户名和密码都要一样

```
#Optional
```

```
SystemConstant.configOfMsgCenter=
```



## ■ 包安装步骤

- 配置tomcat的启动系统属性，加载上一步的配置文件
  - Windows : catalina.bat
  - Linux : catalina.sh

### Windows配置:

```
set JVMOPTS=-Xms1024m -Xmx1024M -XX:PermSize=512M -XX:MaxPermSize=512M -DconfigFilePath=D:\soft\apache-tomcat-7.0.26-new\apache-tomcat-7.0.26\taookeeper-monitor-config.properties
set JAVA_OPTS=%JAVA_OPTS% %LOGGING_MANAGER% %JVMOPTS%
```

### linux配置:

```
JAVA_OPTS="$JAVA_OPTS -DconfigFilePath=/home/zk/taookeeper-monitor-config.properties"
```

## ■ 安装注意事项

- 在配置文件中的配置项指定的目录下新建目录ZooKeeperClientThroughputStat，因为程序不会自己去创建
- 如果zk集群是新构建起来的，还没有统计数据，部署的程序会报错，需要修改一个类文件
  - java\com\taobao\taookeeper\monitor\core\task\runable\ZKServerStatusCollector.java中的方法

storeTaoKeeperStatToDB

```
if(zooKeeperStatus.getReceived()==null || "".equals(zooKeeperStatus.getReceived()))  
    zooKeeperStatus.setReceived("0");  
if(zooKeeperStatus.getSent()==null || "".equals(zooKeeperStatus.getSent()))  
    zooKeeperStatus.setSent("0");  
reportDAO.addTaoKeeperStat( new TaoKeeperStat( clusterId,  
    zooKeeperStatus.getIp(),  
    DateUtil.getNowTime( DateFormat.DateTime ),  
    DateUtil.getNowTime( DateFormat.Date ),  
    MapUtil.size( zooKeeperStatus.getConnections() ),  
    zooKeeperStatus.getWatches(),  
    Long.parseLong( zooKeeperStatus.getSent() ),  
  
    Long.parseLong( zooKeeperStatus.getReceived() ),  
    zooKeeperStatus.getNodeCount(), rwStatistics ) );
```

红框内的代码是新增的

# 第十讲 Taookeeper

■ 访问地址：http://ip:port/taookeeper-monitor/

■ 主要功能：

— 集群配置：

- 以逗号分开集群的机器列表
- 如果节点都部署在同一个机器上，可以借助域名的方式，否则不会显示所有的节点信息

windows-test ▼

| 配置项                                   | 设置参数   |
|---------------------------------------|--|
| ZooKeeper集群名称                         | <input type="text" value="windows-test"/>                      |
| 机器列表                                  | <input type="text" value="z1.test.com:2181,z2.test.com:2182"/> |
| 描述                                    | <input type="text"/>   |
| <input type="button" value="Update"/> |  |

## ▼ Monitor

- 集群配置
- 集群监控
- 机器监控
- 报警设置

## ▼ Admin

- 报警开关
- 系统设置

# 第十讲 Taookeeper

windows-test ▼

| Node IP     | Role | 连接数 | Watch数 | Watched /Total Path | 数据量 Sent/Received | 状态 | 节点自检状态 | 查看趋势 |
|-------------|------|-----|--------|---------------------|-------------------|----|--------|------|
| z1.test.com | F    | 3   | 1      | 1/13                | 37123/35220       |    | OK     |      |
| z2.test.com | L    | 1   | 0      | 0/13                | 13296/11399       |    | OK     |      |

提示:

1. 节点自检 是指对集群中每个IP所在ZK节点上的PATH: **/YINSHI.MONITOR.ALIVE.CHECK** 定期进行三次如下流程：  
**节点连接 - 数据发布 - 修改通知 - 获取数据 - 数据对比**, 三次流程均成功视为该节点处于正常状态。

2. 角色分类: L: Leader, F: Follower, O: Observer, S: Standalone

## ZooKeeper实时读写TPS

|             |   |             |   |
|-------------|---|-------------|---|
| z1.test.com | RWStatistics<br>[getChildren2=0.0,<br>createSession=0.0,<br>closeSession=0.0,<br>setData=0.0,<br>setWatches=0.0,<br>getChildren=0.0,<br>delete=0.0,<br>create=0.0,<br>exists=0.0,<br>getData=0.0] | z2.test.com | RWStatistics<br>[getChildren2=0.0,<br>createSession=0.0,<br>closeSession=0.0,<br>setData=0.0,<br>setWatches=0.0,<br>getChildren=0.0,<br>delete=0.0,<br>create=0.0,<br>exists=0.0,<br>getData=0.0] |
|-------------|---|-------------|---|

详情

关闭

lx552486d7d560009: /192.168.1.101, send to server: 4879, receive from server: 4879  
lx552486d7d560000: /127.0.0.1, send to server: 4879, receive from server: 4879

关闭

连接详情

详情

关闭

Watcher详情

0x5524a8604ec0284  
/YINSHI.MONITOR.ALIVE.CHECK

关闭

## ■ 主要功能：

### — 集群监控：

- Zk本身的代码是不支持rwps的监控的，taookeeper的作者创建了一个分支，它修改了部分源码，并且提交到了zk的官方问题列表：<https://issues.apache.org/jira/browse/ZOOKEEPER-1804>
- Rwps支持需要做的工作
  - 修改文件：NIOServerCnxn NettyServerCnxn ServerCnxn ServerStats ZooKeeperServer；详情见上面问题列表页面下面带的附件：<https://issues.apache.org/jira/secure/attachment/12612555/ZOOKEEPER-1804.patch>
  - 在zkServer中增加启动属性-Dserver\_process\_stats=true
  - 修改代码中的bug：类文件ZKServerStatusCollector中的line.contains( "getData" )，写错成line.contains("getDate")
  - 修改bug：如果每秒处理数小于1时会全部显示为0，不合理；类文件ZKServerStatusCollector中所有计算的地方 $\text{Math.round}(\text{Double.valueOf}(\text{StringUtil.trimToEmpty}(\text{line})) * 100) / 100$  后面的100改成100d，表示double类型
- 节点自检是指对集群中每个IP所在ZK节点上的PATH: **/YINSHI.MONITOR.ALIVE.CHECK** 定期进行三次如下流程：**节点连接 - 数据发布 - 修改通知 - 获取数据 - 数据对比**, 三次流程均成功视为该节点处于正常状态

# 第十讲 Taookeeper

## ■ 监控报警：

- 默认提供了阿里旺旺和短信的报警机制，但是通道是阿里的，需要修改成自己的，修改地方为：

- 类com.taobao.taookeeper.reporter.alarm.TbMessageSender

```
LOG.info( "[Taokeeper]Send message: " + url );  
return "ok".equalsIgnoreCase( NetUtil.getContentOfUrl( url ) );
```

- 增加新的报警机制

- 实现接口com.taobao.taookeeper.reporter.alarm.MessageSender
- 在需要报警的地方调用发送消息

- ZKServerStatusCollector类如下地方：

- ZKServerPerformanceCollector：

```
ThreadPoolManager  
    .addJobToMessageSendExecutor(new TbMessageSender(  
        new Message(phoneList,
```

```
// 2.1.1  
if ( GlobalInstance.needAlarm.get() ) {  
    ThreadPoolManager.addJobToMessageSendExecutor( new TbMessageSender(  
        new Message( wangwangList, "ZK Server cpu usage too high-" +  
        Message.MessageType.WANGWANG ),
```

```
2016-01-17 17:05:01.482 - WARN [pool-10-thread-2:runnable.ZKServerStatusCollector@582] - ZooKeeper连接数, Watcher数报警z1.test.com 上的连接  
数达到了: 3, 超过设置的报警阈值: 2.  
2016-01-17 17:05:01.491 - WARN [pool-10-thread-2:runnable.ZKServerStatusCollector@594] - begin send message-----  
-----MyMessageSender-----
```

## ■ Taookeeper总结：

### — 优点：

- 实现了zk的监控功能从无到有
- 增加了读写的tps统计和监控
- 有一定的报表统计功能

### — 缺点

- 代码扩展性差，比如没有封装好消息的发送，提供的接口只是一个线程接口
- 系统bug较多，只是部署起来，并没有深测就发现了3个bug
- 基于maven进行项目管理，但是使用的仓库是淘宝自己的仓库，而这个仓库是在外部不能访问的，导致编译失败
- 没有提供数据的管理功能（查看节点）
- 也没有提供手动清理日志的功能

# Thanks

**FAQ时间**