1. azkaban主要有3个组件：
2. Mysql（关系数据库）
3. AzkabanWebServer（web服务器）
4. AzkabanExecutorServer（执行服务器）
5. Mysql作用

1、websever的使用：

A、project管理

B、跟踪执行状态

C、历史记录

D、调度

E、SLA

2、executorServer的使用：

1、检索工程文件

2、跟踪执行状态

3、日志

4、工作流依赖

三、AzkabanWebServer

主要是管理功能。包括：项目管理、调度、监控执行状态、权限管理。

四、AzkabanExecutorServer

五、模式

1、solo-server模式

2、双服务器模式

3、多executor模式

六、azkaban任务提交

1、创建工程

工程（project）包含一个或多个flows，一个flows包含多个job。

2、job的创建。Job名称就是文件名称

定义一个以“.job”结尾的文本文件。如下：

type=command  
command=echo "data 2 hive"

3、flows创建。添加dependencies属性，属性值为依赖的job。示例如下：

type=command  
# 执行sh脚本，建议这样做，后期只需维护脚本就行了，azkaban定义工作流程  
command=sh /job/o2o\_2\_hive.sh  
dependencies=o2o\_clean\_data

七、通过API提交任务

1、登录：

post请求示例：

http://106.75.115.165:8081?action=login&username=lifuqiang&password=ftFPWDGHrf

curl -k -X POST --data "action=login&username=azkaban&password=azkaban" https://localhost:8443

返回结果示例：

{

"status" : "success",

"session.id" : "c001aba5-a90f-4daf-8f11-62330d034c0a"

}

2、创建工程

curl -k -X POST --data "session.id=9089beb2-576d-47e3-b040-86dbdc7f523e&name=aaaa&description=11" https://localhost:8443/manager?action=create

返回结果示例：

{

"status":"success",

"path":"manager?project=aaaa",

"action":"redirect"

}

3、删除工程

curl -k --get --data "session.id=bca1d75d-6bae-4163-a5b0-378a7d7b5a91&delete=true&project=test-delete-project" https://localhost:8443/manager

4、上传工程文件

curl -k -i -H "Content-Type: multipart/mixed" -X POST --form 'session.id=e7a29776-5783-49d7-afa0-b0e688096b5e' --form 'ajax=upload' --form 'file=@myproject.zip;type=application/zip' --form 'project=MyProject;type/plain' https://localhost:8443/manager

返回结果示例：

{

"error" : "Installation Failed.\nError unzipping file.",

"projectId" : "192",

"version" : "1"

}

5、获取工程flows

curl -k --get --data "session.id=6c96e7d8-4df5-470d-88fe-259392c09eea&ajax=fetchprojectflows&project=azkaban-test-project" https://localhost:8443/manager

返回结果示例：

{

"project" : "test-azkaban",

"projectId" : 192,

"flows" : [ {

"flowId" : "test"

}, {

"flowId" : "test2"

} ]

}

6、获取flows的jobs

curl -k --get --data "session.id=bca1d75d-6bae-4163-a5b0-378a7d7b5a91&ajax=fetchflowgraph&project=texter-1-1&flow=test" https://localhost:8445/manager

返回结果示例：

{

"project" : "azkaban-test-project",

"nodes" : [ {

"id" : "test-final",

"type" : "command",

"in" : [ "test-job-3" ]

}, {

"id" : "test-job-start",

"type" : "java"

}, {

"id" : "test-job-3",

"type" : "java",

"in" : [ "test-job-2" ]

}, {

"id" : "test-job-2",

"type" : "java",

"in" : [ "test-job-start" ]

} ],

"flow" : "test",

"projectId" : 192

}