DataXone 交换集成管理平台 安装手册





法律声明

- 迪思杰集团拥有本产品及相关文档的全部版权,受法律保护。
- 未经本公司书面许可,任何单位及个人不得以任何方式或理由对本产品的任何部分进行 复制、抄录和传播或与其他产品捆绑使用、销售。
- 本文档中所提及的其他商标或者产品名称均为各自持有者所拥有的商标或产品名称,版 权归相应的持有者所有。

联系我们

迪思杰集团欢迎您通过尽可能多的渠道向我们提供信息,您的意见和问题都会得到我们的重视和妥善处理,请将反馈信息投递到下述地址:

迪思杰集团北京总部			
地址:	北京市西城区新外大街 28 号办公楼 B 座 210 室		
邮编:	100088		
电话:	010-82051581(82/84/85/86/87)		
传真:	010-82051590		
E-mail:	Market@dsgdata.com		
上海办事处			
地址:	上海市徐汇区漕宝路 86 号光大会展中心 F 座 2103 室		
电话:	021-64320873		
广州办事处			
地址:	广州市天河区侨林街 63 号桥林苑 B 座 502 室		
电话:	020-38827900		
成都办事处			
地址:	成都市武侯区桐梓林北路二号凯莱帝景花园 B 座 15 楼 B 室		
电话:	028-61540211		
沈阳办事外			
地址・	沈阳市沈河区奉天街 373-1 号领委 e 家 C2 座 10 楼 3 号		
电话.	024-88525233		
南 百九東外	027-00323233		
用承须爭处			
地址:	用示甲取假区汉甲路 8 亏金牝国际厂场 1820 至		



电话:	025-84773839
杭州办事处	
地址:	杭州市西湖区文三路 477 号华星科技大厦 430 室
电话:	0571-56883636
长沙办事处	
地址:	湖南省长沙市雨花区芙蓉中路二段 459 号芙蓉公馆 A 栋 1 单元 2312 室
电话:	0731-85860224
福州办事处	
地址:	福州市井大路 113 号七星佳寓 1210 房
电话:	0591-87400366
武汉办事处	
地址:	武汉市东湖新技术开发区鲁巷光谷街尚都酒店二栋 2208 室
电话:	027-87644355
西安办事处	
地址:	西安市雁塔区高新开发区高新一路5号A座306室
电话:	029-83151908
济南办事处	
地址:	济南市高新区新泺大街 1299 号鑫盛大厦 1 号楼 14B
电话:	0531-88813281



目录

1	部署前准备		
	1.1	软硬	件资源要求1
		1.1.1	硬件需求1
		1.1.2	软件需求1
	1.2	Data	Xone 平台软件获取1
2	部	署介绍	43
	2.1	部署	说明3
	2.2	部署	流程3
3	部	署过程	<u>-</u> 4
	3.1	创建	dsg 用户及 DataXone 安装目录 4
	3.2	MyS	QL 数据库部署6
		3.2.1	安装 MySQL 数据库 6
		3.2.2	创建 MySQL 数据库用户 dsg9
		3.2.3	创建 MySQL 数据库10
3.3 JDK 部署		JDK	部署10
	3.4	Data	Xone&Tomcat 部署11
		3.4.1	部署 DataXone 主平台11
		3.4.2	部署 DataXone 监控主题模块15
		3.4.3	启动 Tomcat18
	3.5	登录	DataXone 平台19



1 部署前准备

介绍 DataXone 交换集成管理平台安装前的准备工作。

本章节主要包含以下主题:

- 软硬件资源要求
- **DataXone** 平台软件获取

1.1 软硬件资源要求

在安装 DataXone 平台软件前,请确保安装环境满足硬件及软件要求。

1.1.1 硬件需求

DataXone 平台安装运行所需操作系统硬件资源如表 1 所示。

需求端	硬件名称	安装目录最少需求
DataXone 平台服务器	磁盘空间	2048MB
	内存	2048MB
	网络	至少 100MB 网络

表 1 操作系统硬件资源

软件部署前,请先按照表中要求预先准备一台服务器。

1.1.2 软件需求

DataXone 平台支持的软件需求如表 2 所示。

表 2 软件需求

需求端	操作系统	数据库版本
DataXone 平台服务器	Linux(redhat6/7)	MySQL5.5

1.2 DataXone 平台软件获取

DataXone 平台软件包的存取如表 3 所示。

表 3 DataXone 平台软件获取

存放机器: 192.168.1	.186	
软件包	软件版本	存放目录



MySQL 软件包	 MySQL-5.5.61-2.el6.x86_64.rpm -bundle.tar (redhat_v6) MySQL-5.5.61-2.el7.x86_64.rpm 	/dataXone/mysql	
	-bundle.tar (redhat_v7)		
	(建议使用 MySQL5.5 版本)		
JDK 软件包	jdk-8u73-linux-x64.tar.gz	/dataXone/jdk	
	(建议使用 jdk1.8 版本)		
Tomcat 软件包	apache-tomcat-8.5.32.tar.gz	/dataXone/tomcat	
	(建议使用 tomcat8.5 版本)		
DataXone 平台软 件包	● autoMaticEngineBoot-1.0.0.war_ 时间戳	/dataXone/dataXone_ver 时间戳.tar.gz	
	● dataXone_theme_dsg_project-0. 0.1.war_时间戳		
	● DSG_时间戳.tar.gz		
DataXone 平台	● automation_dsg_db_时间戳.sql	/dataXone/dataXone_ver	
SQL	● dataxone_c_info_时间戳.sql	町町載.tai.yz	
	● dataxone_bigdatas_db_时间 戳.sql		
	 pgmd_info_db.sql 		
	pgmd_history_db.sql		
DataXone 平台模 块包	SuperSync_Package(ds.tar.gz/dt.tar. gz)	/dataXone/SODT_D_M/ Supersync/	



2 部署介绍

介绍 DataXone 交换集成管理平台部署流程介绍。

本章节主要包含以下主题:

- 部署说明
- 部署流程

2.1 部署说明

DataXone 交换集成管理平台,需要部署 MySQL 数据库、JDK、Tomcat,并配置 DataXone 项目信息,以实现平台搭建。

● MySQL 部署

为存储 DataXone 元数据信息,服务器中需要安装 MySQL 数据库。

● JDK、Tomcat 部署

在服务器中需要部署 JDK 及 Tomcat,设置环境变量。

● DataXone 程序配置

在 Tomcat 中加入 DataXone 项目,并配置 DataXone 与 MySQL 数据库,使其可 正常展示信息。

2.2 部署流程

DataXone 交换集成管理平台部署流程的详细介绍请参见表 3。

表 3 DataXone 部署流程介绍

步骤	操作	
1	创建 dsg 用户和 DataXone 安装目录	在服务器中创建 dsg 用户,并创建 DataXone 安装目录,具体操作请参见"3.1 创建 dsg 用户 及 DataXone 安装目录"。
2	部署 MySQL 数据库	在服务器中部署 MySQL 数据库,用于存储 DataXone 元数据信息,具体操作请参见 "3.2MySQL 数据库部署"。
3	部署 JDK	服务器中需要安装 JDK,以支持 DataXone 平 台运行,具体操作请参见"3.3JDK 部署"。
4	部署 DataXone&Tomcat	在服务器中安装 Tomcat,并在 Tomcat 中部署 DataXone项目,同时配置 DataXone 与 MySQL 数据库的连接,具体操作请参见 "3.4DataXone&Tomcat 部署"。



3 部署过程

介绍 DataXone 交换集成管理平台部署过程。

本章节主要包含以下主题:

- 创建 dsg 用户及 DataXone 安装目录
- MySQL 数据库部署
- JDK 部署
- DataXone&Tomcat 部署

3.1 创建 dsg 用户及 DataXone 安装目录

在需要部署 DataXone 交换集成管理平台的服务器中,创建 dsg 用户,该用户用于部署 DataXone 平台,并创建 DataXone 平台安装需要使用的目录。

前提条件

已获取服务器系统用户 root 的权限。

操作步骤

1、 使用 root 用户登录服务器, 创建 dsg 用户。

useradd dsg

passwd dsg

- 2、 切换到 dsg 用户, 创建 DataXone 平台需要使用的安装目录。
 - # su dsg

\$ mkdir -p dataxone mysql jdk theme_tomcat tomcat engineDsg/c engineDsg/java

🛄 说明

- dataxone 目录:用于存放 DataXone 平台需要使用的所有软件包。
- mysql 目录:用于存放 MySQL 软件包解压后的文件。
- jdk 目录:用于 JDK 软件的安装目录。
- tomcat 目录:用于 DataXone 主平台 Tomcat 软件的安装目录。
- theme_tomcat 目录:用于 DataXone 监控模块 Tomcat 软件的安装目录。
- engineDsg/c 目录:用于存放 SuperSync 源端和目标端软件包,如源端 ds.tar.gz、目标端 dt.tar.gz。



- engineDsg/java 目录:用于存放 SuperSync 使用的 java 版本软件包,如 Yxad_socket_kafka10.tar.gz。
- 3、 使用 dsg 用户,从 192.168.1.186 上下载 DataXone 平台需要使用的软件包。
 - a. 通过ftp 连接到 192.168.1.186。

```
# su - dsg
```

```
$ cd dataxone
```

```
$ ftp 192.168.1.186
```

Name (192.168.1.186:root): file

Password: file_admin@dsg

b. 获取 DataXone 平台软件包和 SQL 文件。

ftp> cd dataXone

ftp> get dataXone_ver 时间戳.tar.gz

c. 获取 MySQL 安装包,根据服务器的系统版本下载对应的 MySQL 软件包:

ftp> cd mysql

Linux redhat6 对应软件包:

ftp> get MySQL-5.5.61-2.el6.x86_64.rpm-bundle.tar.gz

Linux redhat6 对应软件包:

ftp> get MySQL-5.5.61-2.el7.x86_64.rpm-bundle.tar.gz

d. 获取 JDK 安装包。

ftp> cd ../jdk

ftp> get jdk-8u73-linux-x64.tar.gz

e. 获取 Tomcat 安装包.

ftp> cd ../tomcat

ftp> get apache-tomcat-8.5.32.tar.gz

f. 获取 Kafka 目标端软件包 (根据需求可选):

cd ../kafka

get dt.tar.gz

get Yxad_socket_kafka10.tar.gz

\$ cp dt.tar.gz Yxad_socket_kafka10.tar.gz /home/dsg/engineDsg/java

g. 获取 SuperSync 平台各种数据库同步软件包(根据需求选择相应的数据库及 操作系统的源端、目标端软件包):

ftp> cd ../SODT_D_M/Supersync/具体的数据库及操作系统



如:路径为 /home/file/dataXone/SODT_D_M/Supersync/ORACLE/Linux2.6/64bit/oracle .11.2.0.1.0

ftp> get ds.tar.gz

ftp> get dt.tar.gz

4、退出 ftp,根据实际情况在/home/dsg/engineDsg/c路径中创建 SODT_D_M/Supersync/.....,将 SuperSync 软件包拷贝到 /home/dsg/engineDsg/c/SODT_D_M/Supersync/......目录中。

ftp> bye

\$ mkdir -p /home/dsg/engineDsg/c/SODT_D_M/Supersync/.....

如 186 上路径为 /home/file/dataXone/SODT_D_M/Supersync/ORACLE/Linux2.6/64bit/oracle.11. 2.0.1.0,则创建的路径为 /home/dsg/engineDsg/c/SODT_D_M/Supersync/ORACLE/Linux2.6/64bit/oracle. 11.2.0.1.0

\$ cp ds.tar.gz dt.tar.gz /home/dsg/engineDsg/c/SODT_D_M/Supersync/.....

3.2 MySQL 数据库部署

在 DataXone 服务器中安装 MySQL 数据库,并创建 DataXone 数据库和 dsg 用户。

3.2.1 安装 MySQL 数据库

在 DataXone 服务器中安装 MySQL 数据库。MySQL 请安装 5.5 版本。

前提条件

- 已创建 dsg 用户和 DataXone 安装目录,并获取 DataXone 中需要使用的安装包, 具体操作请参见 "3.1 创建 DSG 用户及 DataXone 安装目录"。
- 己获取服务器系统用户 root 的权限。

操作步骤

- 1、使用 root 用户登陆服务器查看系统中已经安装的 MySQL 数据库,如果数据库版本 不满足要求,请进行卸载,卸载前需要停止已经启动的 MySQL 服务。
 - \$ su root
 - # rpm -qa |grep -i mysql
 - # ps -ef | grep mysql
 - # service mysqld stop
 - # rpm -e --nodeps mysql 版本
 - # find / -name mysql



rm -rf /etc/my.cnf* /usr/bin/mysql /usr/local/mysql /var/lib/mysql /usr/share/mysql

🛄 说明

请尽量删除干净以前的 MySQL 版本文件。

2、 查看 Linux 系统版本,根据具体的系统版本,安装相应的 MySQL。

cd /home/dsg/dataxone

cat /etc/redhat-release

- 如果服务器版本为 Redhat Linux 6:
 - a. 将 MySQL 软件包解压到 mysql 目录中。

tar zxf MySQL-5.5.61-2.el6.x86_64.rpm-bundle.tar.gz -C /home/dsg/mysql

b. 安装 MySQL 所有软件包。

cd /home/dsg/mysql/

rpm -ivh --nodeps mysql-community-client-5.5.61-2.el6.x86_64.rpm

rpm -ivh --nodeps
mysql-community-devel-5.5.61-2.el6.x86_64.rpm

rpm -ivh --nodeps
mysql-community-bench-5.5.61-2.el6.x86_64.rpm

rpm -ivh --nodeps
mysql-community-common-5.5.61-2.el6.x86_64.rpm

rpm -ivh --nodeps
mysql-community-embedded-5.5.61-2.el6.x86_64.rpm

rpm -ivh --nodeps
mysql-community-embedded-devel-5.5.61-2.el6.x86_6
4.rpm

rpm -ivh --nodeps
mysql-community-libs-5.5.61-2.el6.x86_64.rpm

rpm -ivh --nodeps mysql-community-libs-compat-5.5.61-2.el6.x86_64.rp m

```
# rpm -ivh --nodeps
mysql-community-server-5.5.61-2.el6.x86_64.rpm
```

rpm -ivh --nodeps
mysql-community-test-5.5.61-2.el6.x86_64.rpm

- 如果服务器版本为 Redhat Linux 7:
 - a. 将 MySQL 软件包解压到 mysql 目录中。

```
# tar zxf MySQL-5.5.61-2.el7.x86_64.rpm-bundle.tar.gz
-C /home/dsg/mysql
```



cd /home/dsg/mysql/

b. 安装 MySQL 所有软件包。

rpm -ivh --nodeps
mysql-community-common-5.5.61-2.el7.x86_64.rpm

rpm -ivh --nodeps
mysql-community-libs-5.5.61-2.el7.x86_64.rpm

rpm -ivh --nodeps
mysql-community-client-5.5.61-2.el7.x86_64.rpm

rpm -ivh --nodeps mysql-community-bench-5.5.61-2.el7.x86_64.rpm

rpm -ivh --nodeps
mysql-community-devel-5.5.61-2.el7.x86_64.rpm

rpm -ivh --nodeps
mysql-community-embedded-5.5.61-2.el7.x86_64.rpm

rpm -ivh --nodeps
mysql-community-embedded-devel-5.5.61-2.el7.x86_6
4.rpm

rpm -ivh --nodeps
mysql-community-minimal-debuginfo-5.5.61-2.el7.x86
_64.rpm

rpm -ivh --nodeps
mysql-community-test-5.5.61-2.el7.x86_64.rpm

rpm -ivh --nodeps
mysql-community-server-5.5.61-2.el7.x86_64.rpm

3、 修改/etc/my.cnf 配置文件,并添加 MySQL 字符集特殊设置。

vi /etc/my.cnf

a. 在[mysqld]下方增加以下参数:

#skip-grant-tables

max_connections=10000

max_user_connections=1000

lower_case_table_names=1

character-set-server=utf8

innodb_flush_log_at_trx_commit=0

b. 添加[client]并在下方增加参数:

[client]

default-character-set=utf8

- 4、 启动 MySQL 数据库。
 - **#** service mysqld start



5、 登录 MySQL 数据库,修改 MySQL 的 root 用户密码。

mysql

mysql> use mysql

mysql> update user set password=password('new password')
where user='root';

mysql> flush privileges;

6、 查看 MySQL 字符集,字符集设置正常,实现 DataXone 平台界面中文显示。

mysql> show variables like '%char%';

Variable_name	Value
character_set_client character_set_connection character_set_database character_set_filesystem character_set_results character_set_server character_set_system character_sets_dir	utf8 utf8 utf8 binary utf8 utf8 utf8 utf8 /usr/share/mysql/charsets/

3.2.2 创建 MySQL 数据库用户 dsg

在 MySQL 数据库中创建 dsg 用户,并为该用户赋予本机及其他主机都可以访问数据库 的权限。

前提条件

- 服务器中已安装了 MySQL 数据库,具体操作请参见"3.2.1 安装 MySQL 数据库"。
- 拥有 MySQL 数据库的 root 操作权限。

操作步骤

1、 使用 root 用户登录 MySQL 数据库,为数据库创建 dsg 用户。

mysql -uroot -p 密码

mysql> insert into mysql.user(Host,User,Password) values
('%','dsg',password('dsg'));

mysql> insert into mysql.user(Host,User,Password) values ('主 机名','dsg',password('dsg'));

2、为dsg用户赋予本机访问数据库权限。

mysql> grant all privileges on *.* to dsg@'localhost' identified by 'dsg';

mysql> grant all privileges on *.* to dsg@'主机名' identified by 'dsg';

mysql> grant all privileges on *.* to dsg@'%' identified by 'dsg';

🛄 说明



主机名请根据实际主机名修改。

3、 使 dsg 用户权限立即生效。

mysql> flush privileges;

3.2.3 创建 MySQL 数据库

在 MySQL 数据库中创建数据库 automation_dsg_db、dataxone_c_info、 dataxone_bigdatas_db、pgmd_info_db 和 pgmd_history_db,用于存放 DataXone 的 初始化配置数据。

前提条件

- 服务器中已安装了 MySQL 数据库,具体操作请参见"3.2.1 安装 MySQL 数据库"。
- 拥有 MySQL 数据库的 root 操作权限。

操作步骤

1、使用 root 用户登录 MySQL 数据库,创建下列数据库。 # mysql -uroot -p 密码 mysql> create database automation_dsg_db; mysql> create database dataxone_c_info; mysql> create database dataxone_bigdatas_db; mysql> create database pgmd_info_db;

mysql> create database pgmd_history_db;

3.3 JDK 部署

DataXone 平台基于 JAVA 开发,用户需要在服务器中安装 JDK 软件。

前提条件

已创建 dsg 用户和 DataXone 安装目录,并获取 DataXone 中需要使用的安装包,具体操作请参见"3.1 创建 DSG 用户及 DataXone 安装目录"。

操作步骤

1、 使用 root 用户,在服务器中卸载系统自带的 OpenJDK。

rpm -qa |grep jdk

- # rpm -e --nodeps OpenJDK 版本
- 2、 使用 dsg 用户安装 JDK。
 - # su dsg
 - \$ cd /home/dsg/dataxone
 - \$ tar zxf jdk-8u73-linux-x64.tar.gz -C /home/dsg/jdk



3、 修改当前用户的环境变量配置文件。

\$ vi ~/.bash_profile

- 4、在.bash_profile 文件的末尾添加 JAVA_HOME、PATH 变量。
 export JAVA_HOME=/home/dsg/jdk/jdk1.8.0_73
 export PATH=\$PATH:\$JAVA_HOME/bin
- 5、 保存文件修改,并使环境变量生效。
 - \$:wq

\$ source ~/.bash_profile

🛄 说明

用户也可以根据操作系统需求,自行下载 JDK 软件安装,请保证 JDK 版本为 1.8 及以上。

3.4 DataXone&Tomcat 部署

在 Tomcat 中部署 DataXone 项目,完成平台 DataXone 部署。

用户需要使用两个 Tomcat 分别部署 DataXone 主平台和 DataXone 监控主题模块。

- DataXone 主平台: 主要用于同步流程部署、数据脱敏、数据探测等功能操作。
- DataXone 监控主题模块:主要是用于对同步流程进行监控。

3.4.1 部署 DataXone 主平台

对 DataXone 主平台进行部署。

DataXone 主平台: 主要用于同步流程部署、数据脱敏、数据探测等功能操作。

前提条件

- 己创建 dsg 用户和 DataXone 安装目录,并获取 DataXone 中需要使用的安装包, 具体操作请参见 "3.1 创建 DSG 用户及 DataXone 安装目录"。
- 服务器中已安装了 MySQL 数据库,具体操作请参见"3.2 MySQL 数据库部署"。
- 服务器中已安装 JDK 软件,具体操作请参见"3.3 JDK 部署"。

操作步骤

1、 使用 dsg 用户安装 DataXone 主平台需要使用的 tomcat,对已上传的 tomcat 软件 安装包进行解压。

su - dsg

- \$ cd /home/dsg/dataxone
- \$ tar -zxvf apache-tomcat-8.5.32.tar.gz -C /home/dsg/tomcat



- **2**、 将 DataXone 平台软件包 "dataXone_ver 时间戳.tar.gz"进行解压。
 - \$ cd /home/dsg/dataXone
 - \$ tar -zxvf dataXone_ver 时间戳.tar.gz
- **3**、 将解压目录中的 "autoMaticEngineBoot-1.0.0.war_时间戳"和 "DSG_时间 戳.tar.gz" 拷贝到 tomcat 解压目录/ webapps 中,并对其进行解压。
 - \$ cd /home/dsg/dataXone/dataXone_ver 时间戳

\$ cp autoMaticEngineBoot-1.0.0.war_时间戳 DSG_时间戳.tar.gz /home/dsg/tomcat/apache-tomcat-8.5.32/webapps

```
$ cd /home/dsg/tomcat/apache-tomcat-8.5.32/webapps
```

\$ unzip -oq autoMaticEngineBoot-1.0.0.war_时间戳 -d autoMaticEngineBoot-1.0.0

```
$ tar -zxvf DSG_时间戳.tar.gz
```

- 4、 使用 dsg 用户登录 MySQL 数据库,将 DataXone 的 SQL 脚本"automation_dsg_db_时间戳.sql"导入到 MySQL 的 automation_dsg_db 库。
 - \$ cd /home/dsg/dataXone/dataXone_ver 时间戳
 - \$ mysql -udsg -pdsg

mysql> use automation_dsg_db

mysql> source automation_dsg_db_时间戳.sql

mysql> truncate table op_save_flowmodule_info_tb;

- mysql> truncate table log_engine_sysinfo;
- mysql> truncate table op_machine_conf_tb;
- mysql> truncate table sys_database_host_tb;
- 5、 将"dataxone_c_info_时间戳.sql"文件拷贝到 dataxone_c_info 库中。

mysql> use dataxone_c_info mysql> source dataxone_c_info_时间戳.sql mysql> truncate table os_status_detail; mysql> truncate table state_detail; mysql> truncate table state_profile;

6、 将 "dataxone_bigdatas_db_时间戳.sql" 拷贝到 dataxone_bigdatas_db 库中。

mysql> use dataxone_bigdatas_db mysql> source dataxone_bigdatas_db_时间戳.sql

7、 将 "pgmd_info_db.sql" 拷贝到 pgmd_info_db 库中。

mysql> use pgmd_info_db; mysql> source pgmd_info_db.sql



8、 将 "pgmd_history_db.sql" 拷贝到 pgmd_history_db 库中。

mysql> use pgmd_history_db;

mysql> source pgmd_history_db.sql

9、 SQL 脚本快速导入完成后,为保证数据库安全,需注释/etc/my.cnf 中参数,并重启 MySQL 数据库。

\$ su - root

vi /etc/my.cnf

将以下参数进行注释:

innodb_flush_log_at_trx_commit=0

重启 MySQL 数据库:

service mysqld restart

10、使用 dsg 用户进入 Tomcat 解压目录/webapps/ autoMaticEngineBoot-1.0.0/WEB-INF/classes 目录中, 打开 application.properties 文件。

su - dsg

\$ cd

/home/dsg/tomcat/apache-tomcat-8.5.32/webapps/autoMati cEngineBoot-1.0.0/WEB-INF/classes

\$ vi application.properties

11、 在 application.properties 文件中,修改黑体参数,配置 MySQL 数据库连接信息。

module.c.mysql.ip_address=192.168.1.149

module.c.mysql.port=3306

module.c.mysql.db_name=pgmd

spring.datasource.url =

jdbc:mysql://**192.168.1.9:3306/automation_dsg_db**?useUnicode=true&charact erEncoding=utf-8

spring.datasource.username = dsg

spring.datasource.password = dsg

.....

#sys template path

this.system.template.path= /home/dsg/tomcat/apache-tomcat-8.5.32/webapps/DSG #Engine Package path

this.engine.package.path= /home/dsg/engineDsg

#The Number Of Timers Executed by Remote Host After Executing Shell-script

this.engine.run.number=8

🛄 说明

- module.c.mysql.ip_address: 配置连接 MySQL 所在的服务器的 IP 地址。
- module.c.mysql.port: 配置连接的 MySQL 端口号, MySQL 端口号一般 默认为 3306。
- module.c.mysql.db_name:配置连接的 MySQL 数据库名,默认配置 pgmd 即可。
- spring.datasource.url:用于配置连接 MySQL 中已经创建的 DataXone 数据库,请根据实际情况将黑体参数修改为对应的 IP 地址、MySQL 端口号、连接的 MySQL 数据库名。"automation_dsg_db"为 MySQL 中创建的DataXone 数据库,数据库的创建请参见"3.2.3 创建 MySQL 数据库"。
- spring.datasource.username 和 spring.datasource.password 用于配置 MySQL 访问用户,该参数请配置为 dsg, dsg 用户的创建请参见 "3.2.2 创建 MySQL 数据库用户 dsg"。
- this.system.template.path: 当前对应的 DSG 目录,请配置到 /home/dsg/tomcat/apache-tomcat-8.5.32/webapps/DSG。
- this.engine.package.path: 配置 SuperSync 程序包所在位置,请配置的 到 engineDsg 路径。
- this.engine.run.number:因现场客户提供的设备性能不一,当出现上传完成后解压的文件缺失时可以增加这个数值延长解压等待时间,默认值为8。
- **12、** 进入 tomcat 解压目录/webapps/DSG/libs 目录中修改 common.js 文件, 配置 IP 和 t 端口号。

\$ cd

/home/dsg/tomcat/apache-tomcat-8.5.32/webapps/DSG/libs

\$ vi common.js

var basePath='http:// 192.168.1.149:8080/autoMaticEngineBoot-1.0.0/'

var basePath_theme = 'http://**192.168.1.9:8070**/dataXone_theme_dsg_project-0.0.1/'

🛄 说明

- basePath: 配置 DataXone 主平台公共接口的基础路径,即部署本机 IP 地址和 Dataone 主平台的 tomcat 端口号。DataXone 主平台端口号建议 直接配置为 8080,
- basePath_theme: 配置 DataXone 监控主题模块的基础路径,即部署本机 IP 地址和 DataXone 监控模块平台的 tomcat 端口号。该端口号与DataXone 主平台端口号必须不同。

7、 修改 log4j.properties 文件, 配置 Tomcat 日志输出类型。

\$ cd

/home/dsg/tomcat/apache-tomcat-8.5.32/webapps/autoMati cEngineBoot-1.0.0/WEB-INF/classes/log4j.properties

8、 修改 log4j.rootLogger 和 log4j.logger.com.neo.dao 两个属性,只显示告警和错误 类型。

log4j.rootLogger=warn,error

log4j.logger.com.neo.dao=ERROR,sql

9、 tomcat 启动后,将显示对应的日志类型。

3.4.2 部署 DataXone 监控主题模块

对 DataXone 监控主题模块进行配置。需要在新的 Tomcat 中部署。

DataXone 监控主题模块: 主要是用于对同步流程进行监控。

前提条件

- 己创建 dsg 用户和 DataXone 安装目录,并获取 DataXone 中需要使用的安装包, 具体操作请参见 "3.1 创建 DSG 用户及 DataXone 安装目录"。
- 服务器中已安装了 MySQL 数据库,具体操作请参见"3.2 MySQL 数据库部署"。
- 服务器中已安装 JDK 软件,具体操作请参见"3.3 JDK 部署"。

操作步骤

- 1、 使用 dsg 用户安装 DataXone 监控主题模块需要使用的 tomcat,对已上传的 tomcat 软件安装包进行解压。
 - # su dsg

\$ cd /home/dsg/dataxone

\$ tar -zxvf apache-tomcat-8.5.32.tar.gz -C
/home/dsg/theme_tomcat

\$ cd /home/dsg/dataXone/dataXone_ver 时间戳

\$ cp dataXone_theme_dsg_project-0.0.1.war_时间戳 /home/dsg/theme_tomcat/apache-tomcat-8.5.32/webapps

\$ cd

/home/dsg/theme_tomcat/apache-tomcat-8.5.32/webapps

\$ upzip -oq dataXone_theme_dsg_project-0.0.1.war_时间戳 -d dataXone_theme_dsg_project-0.0.1



3、 进入 theme_tomcat 解压目录

/webapps/dataXone_theme_dsg_project-0.0.1/WEB-INF/classes $\exists 录中$,修改 application.properties 文件。

\$ cd /home/dsg/theme_tomcat/apache-tomcat-8.5.32/webapps/d ataXone_theme_dsg_project-0.0.1/WEB-INF/classes

\$ vi application.properties

module.c.mysql.ip_address=192.168.1.9

module.c.mysql.port=3306

spring.jmx.enabled=false

spring.datasource.driverClassName = com.mysql.jdbc.Driver

spring.datasource.url =
jdbc:mysql://192.168.1.9:3306/automation_dsg_db?useUnicode=true&charact
erEncoding=utf-8

spring.datasource.username = dsg

spring.datasource.password = dsg

.

#sys template path

this.system.template.path=/home/dsg/tomcat/apache-tomcat-8.5.32/webapps /DSG

#Engine Package path

this.engine.package.path=/home/dsg/engineDsg

#The Number Of Timers Executed by Remote Host After Executing Shell-script

this.engine.run.number=8

🛄 说明

- module.c.mysql.ip_address: 配置连接 MySQL 所在的服务器的 IP 地址。
- module.c.mysql.port: 配置连接的 MySQL 端口号, MySQL 端口号一般 默认为 3306。
- spring.datasource.url:用于配置连接 MySQL 中已经创建的 DataXone 数据库,请根据实际情况将黑体参数修改为对应的 IP 地址、MySQL 端口号、连接的 MySQL 数据库名。"automation_dsg_db"为 MySQL 中创建的DataXone 数据库,数据库的创建请参见"3.2.3 创建 MySQL 数据库"。



- spring.datasource.username 和 spring.datasource.password 用于配置 MySQL 访问用户, 该参数请配置为 dsg, dsg 用户的创建请参见 "3.2.2 创建 MySQL 数据库用户 dsg"。
- this.system.template.path: 当前对应的 DSG 目录,请配置到 /home/dsg/tomcat/apache-tomcat-8.5.32/webapps/DSG。
- this.engine.package.path: 配置 SuperSync 程序包所在位置,请配置的 到 engineDsg 路径。
- this.engine.run.number:因现场客户提供的设备性能不一,当出现上传完成后解压的文件缺失时可以增加这个数值延长解压等待时间,默认值为8。
- **4、**对 theme_tomcat 端口号进行修改,以避免 theme_tomcat 端口号与 DataXone 主 平台 tomcat 端口号冲突。

\$ cd /home/dsg/theme_tomcat/apache-tomcat-8.5.32/conf

\$ vi server.xml

<Server port="8015" shutdown="SHUTDOWN">

<Listener className="org.apache.catalina.startup.VersionLoggerListener" />

<!-- Security listener. Documentation at /docs/config/listeners.html

<Listener className="org.apache.catalina.security.SecurityListener" />

.....

<Service name="Catalina">

.....

<Connector port="8070" protocol="HTTP/1.1"

connectionTimeout="20000"

redirectPort="8443" />

.....

<!-- Define an AJP 1.3 Connector on port 8009 -->

<Connector port="8019" protocol="AJP/1.3" redirectPort="8443" />

.....

🛄 说明

- 第一个端口:修改 tomcat 的关闭端口号,
- 第二个端口号:修改 tomcat 的 HTTP Connector 端口号。该端口号必须与 DataXone 主平台 tomcat 解压目录/webapps/DSG/libs/common.js 文件中 basePath_theme 参数配置的端口号一致:

var basePath_theme =
'http://192.168.1.9:8070/dataXone_theme_dsg_project-0.0.1/'

● 第三个端口号: 修改 tomcat 的 JD Connector 端口号。



5、 修改 log4j.properties 文件的参数 "log4j.rootLogger", 配置 theme_tomcat 日志输 出类型只显示告警和错误日志。

\$ cd

/home/dsg/theme_tomcat/apache-tomcat-8.5.32/webapps/d ataXone_theme_dsg_project-0.0.1/WEB-INF/classes

\$ vi log4j.properties

log4j.rootLogger=warn,error

6、 tomcat 启动后,将显示对应的日志类型。

3.4.3 启动 Tomcat

对已经部署的 DataXone 主平台和监控主题模块的 Tomcat 进行启动操作。请先启动 DataXone 监控主题模块,在启动 DataXone 主平台。

前提条件

- 服务器中已安装了 MySQL 数据库, 具体操作请参见"3.2.1 安装 MySQL 数据库"。
- 服务器中已安装 JDK 软件,具体操作请参见"3.3 JDK 部署"。
- 己部署 DataXone 主平台,具体操作请参见"3.4.1 部署 DataXone 主平台"
- 已部署 DataXone 监控主题平台,具体操作请参见 "3.4.2 部署 DataXone 监控主题模块"。

操作步骤

- 1、 检查当前是否有正在运行的 tomcat。
 - \$ su root
 - # ps -ef | grep tomcat
 - # kill -9 进程号
- 2、 使用 dsg 用户, 启动 DataXone 监控主题模块的 tomcat。
 - # su dsg
 - \$ cd /home/dsg/theme_tomcat/apache-tomcat-8.5.32/bin

\$ sh startup.sh

3、 启动 DataXone 主平台的 tomcat。

\$ cd /home/dsg/tomcat/apache-tomcat-8.5.32/bin

\$ sh startup.sh

- 4、 启动后,可以查看 tomcat 日志。
 - \$ cd /home/dsg/theme_tomcat/apache-tomcat-8.5.32/logs
 - \$ tail -f catalina.out
 - \$ cd /home/dsg/tomcat/apache-tomcat-8.5.32/logs



\$ tail -f catalina.out

图 1 DataXone 登录界面

3.5 登录 DataXone 平台

DataXone 交换集成管理平台部署完成后,可通过浏览器登录平台,确认 DataXone 部 署是否正常。

前提条件

- 服务器中已安装了 MySQL 数据库,具体操作请参见"3.2 MySQL 数据库部署"。
- 服务器中已安装 JDK 软件,具体操作请参见"3.3 JDK 部署"。
- 已部署并启动 DataXone 主平台和监控主题模块,具体操作请参见
 "3.3DataXone&Tomcat 部署"。

操作步骤

1、 在 Windows 环境中,使用浏览器(建议使用谷歌浏览器)访问 "http://DataXone 服务器 IP:8080/DSG/login.html",系统打开 DataXone 登录界面,如图 1 所示。

<image><image><complex-block><complex-block><complex-block>

2、 输入登录用户名/密码: admin/12345, 单击"登录", 系统登录 DataXone 交换集 成管理平台, 如图 2 所示。

图 2 DataXone 交换集成管理平台



DSG ×		808
← → C ① 不安全 192.168.1.9:8080/DSG/pa	ge/homePageScreen/page.html	아 ☆ 🔾
	test	
ds	at	ds
50 - 40 - 30 - 10 - 12 12 10 - 12 12 10 - 12 12 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2		
CPU 8002 9908 = 2019 : 0 8002 • 0		