



## 数据迁移工具 KDTS

### 5.1 概述

金仓数据迁移工具 KDTS (Kingbase Data Transformation Service) 用于将 Oracle、MySQL、SQL Server、GBase、PostgreSQL、DM、KES 等数据库中的数据库对象 (如序列、表、视图、触发器、函数、存储过程、注释等) 及数据平滑地迁移到 KES 数据库, 能有效降低数据库的迁移成本, 缩短周期, 提升数据库迁移效率。

金仓数据迁移工具 KDTS 有以下特点。

#### 1. 灵活的迁移配置

支持多种迁移任务配置, 满足不同迁移场景需求; 支持按条件、单表、多表、结构、数据等多种迁移方式进行数据迁移。

#### 2. 迁移任务执行高效、稳定

以任务为驱动, 基于多线程异步处理机制, 保证单机多线程的迁移任务达到最佳的效率。同时具备完善的容错和重试机制, 避免受外部因素的干扰, 例如网络闪断、数据源不稳定等因素可能导致迁移过程报错或停止。

#### 3. 支持数据库对象二次迁移

迁移任务结束以后如果有迁移失败的对象, 可以在迁移工具中修改出错的 SQL 语句, 保存后进行二次迁移。

#### 4. 高效的性能

结合多线程异步任务、并行计算、多线程任务编排、任务调度等技术进行架构设计, 引入读和写分离线程池 (数据和元数据分离、读数据和写数据分离), 使用双向队列, 按记录数和大小 (含大对象) 对表进行排序, 实现“快慢车道”分离, 从而实现迁移系统的高性能。

#### 5. 完备的迁移报告

迁移完成后提供数据库对象和表数据的迁移结果报告, 以表格报告和图表的方式直观展示迁移结果。

金仓数据迁移工具 KDTS 目前支持的常用源端数据库如表 5-1 所示。KDTS 还支持 OceanBase、GaussDB、Greemplum、HANA、GBase、MongoDB 等数据库, 以后还会支持更多的源端数据库, 实时信息可以查看金仓数据库的官方文档。



表 5-1 KDTs 目前支持的常用源端数据库

源端数据库	支持版本	查询版本的 SQL 语句
Oracle	9i,10g,11g,12c,19c	select * from v\$version
MySQL	5.x,8.x	select version()
SQL Server	2000、2005、2008、2012、2014、2016、2017、2019	select @@version
GBase	8s,8g	select * from v\$version;
PostgreSQL	12	select version()
DM	8	select * from v\$version; select * from v\$instance;
Db2	11	select service_level from sysibmadm.env_inst_info
神通 (OSCAR)	7	selectversion()
KES	V7、V8R3、V8R6、V8R6C7、V9	select version()

数据库迁移工具 KDTs 有 Web 版和 SHELL 版。Web 版通过浏览器以可视化界面方式设置配置项完成数据迁移。SHELL 版通过手动修改配置文件的方式设置配置项完成数据迁移。SHELL 版常用于通过堡垒机形式访问主机、通过远程终端形式访问主机和访问未安装图形化界面主机这几种场景。

KDTs 的产品结构如图 5-1 所示,主要包括以下几个模块。

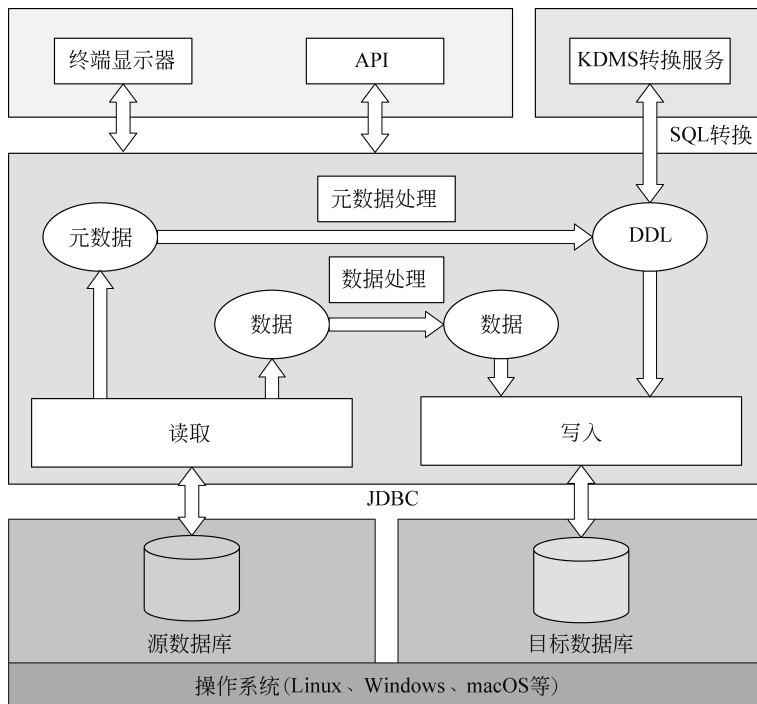


图 5-1 KDTs 的产品结构

- (1) 源数据库读取模块：包括元数据和数据的读取。
- (2) 目标数据库写入模块：包括 DDL 和数据的写入。
- (3) 元数据处理模块：根据用户配置的元数据库处理规则(包括类型映射规则、默认值映射规则、字段映射规则、表映射规则等),形成对应的 DDL,再交由数据写入模块写入目标数据库。
- (4) 数据处理模块,根据用户配置的数据处理规则(如数据过滤条件、表拆分条件等)提取数据并做转换后交由数据写入模块写入目标数据库。

系统以日志、报告等形式向用户反馈迁移结果,API 为 B/S 版迁移程序提供统一的数据迁移服务,并以回调的方式即时反馈处理结果。

KDTS 数据迁移可以将源端数据库的所有表结构和历史数据全量迁移到目标端数据库,包括表结构迁移、表数据迁移以及其他数据库对象迁移。

表结构迁移过程如下。

- (1) 获取源端库表结构(字段、主码)元信息;
- (2) 根据源端及目标端数据类型进行字段类型映射;
- (3) 根据数据库模式、表以及字段的转换配置进行处理操作,生成建表 SQL 语句;
- (4) 目标端执行建表 SQL 语句。

表数据迁移过程如下。

- (1) 根据源端数据库类型构建待搬迁数据表的 SQL 查询语句;
- (2) 执行 SQL 查询语句获取待搬迁的数据;
- (3) 将 DB 类型的数据转为 Java 类型的数据进行封装供中间数据流转;
- (4) 通过 copy 或 insert 方式将数据批量加载到目标端。

其他数据库对象迁移过程如下。

- (1) 获取源端数据库数据对象(如触发器、序列、函数、存储过程、包、包体等)元信息。
- (2) 根据源端及目标端的 SQL 语法进行 SQL 语句转换,生成对象创建 SQL 语句。需要注意的是,目前对于 PL/SQL 只是做了关键语法结构的修改,需要 KES 进行适配。
- (3) 在目标端执行对象创建 SQL 语句。



## 5.2 安装部署

金仓数据迁移工具 KDTS 作为金仓数据库 KES 的客户端工具,随金仓数据库一起安装部署。安装金仓数据库的客户端后,就可以在安装目录下看到 KDTS,例如下面从安装 KES 的目录下可以看到安装 KDTS 工具的 KDts 服务,该目录下的两个子目录分别对应 KDTS 工具的 Web 版和 SHELL 版,如下所示。

```
[kingbase@kingbase KDts]$ pwd
/KingbaseES/V9/ClientTools/guitools/KDts
[kingbase@kingbase KDts]$ ls
KDTS-CLI  KDTS-WEB
```

本节实验使用 KDTS 工具将 Oracle 数据库中的 TPCC 数据迁移到 KES 数据库系统中的 tpcc 数据库中。源端使用 Oracle 数据库(192.168.40.40),目标端使用 KES 数据库(192.



168.40.111)。

### 1. 源端实验环境部署

在源端使用 BenchmarkSQL 给 Oracle 数据库中的 TPCC 用户加载了 10 个 Warehouse 的数据。首先安装 BenchmarkSQL 测试软件,步骤如下所示。

(1) 使用 root 用户安装相关依赖包。

```
[root@oracle40 ~]# yum install gcc glibc-headers gcc-c++ gcc-gfortran readline-devel  
libXt-devel pcre-devel libcurl libcurl-devel -y  
  
[root@oracle40 ~]# yum install ncurses ncurses-devel autoconf automake zlib zlib-devel  
bzip2 bzip2-devel xz-devel -y  
  
[root@oracle40 ~]# yum install java-1.8.0-openjdk ant -y
```

(2) 使用 Oracle 用户安装 BenchmarkSQL。

```
# 上传 BenchmarkSQL 到虚拟机  
[oracle@oracle40 ~]$ ls -l b*  
-rw-r--r-- 1 oracle oinstall 2263539 12月 24 10:26 benchmarksql-5.0.zip  
# 解压 benchmarksql 安装包  
[oracle@oracle40 ~]$ unzip benchmarksql-5.0.zip  
[oracle@oracle40 ~]$ ls  
benchmarksql-5.0 benchmarksql-5.0.zip 公共 模板 视频 图片 文档 下载 音乐 桌面  
  
# 编译 benchmarksql  
[oracle@oracle40 ~]$ cd benchmarksql-5.0/  
[oracle@oracle40 benchmarksql-5.0]$ ant  
Buildfile: /home/oracle/benchmarksql-5.0/build.xml  
... ..  
BUILD SUCCESSFUL  
Total time: 1 second  
[oracle@oracle40 benchmarksql-5.0]$ ls  
build build.xml dist doc HOW-TO-RUN.txt lib README.md run src
```

(3) 配置 BenchmarkSQL 可以访问 Oracle 数据库。

把 Oracle 的 JDBC 驱动复制到 BenchmarkSQL 的相应目录下。

```
[oracle@oracle40 oracle]$ pwd  
/home/oracle/benchmarksql-5.0/lib/oracle  
[oracle@oracle40 oracle]$ cp /u01/app/oracle/product/11.2.0/dbhome_1/jdbc/lib/ojdbc6.jar  
/home/oracle/benchmarksql-5.0/lib/oracle  
[oracle@oracle40 oracle]$ ls  
ojdbc6.jar README.txt
```

(4) 在 Oracle 中创建数据库用户并授权。

```
[oracle@oracle40 ~]$ sqlplus / as sysdba  
连接到:  
Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production  
With the Partitioning, OLAP, Data Mining and Real Application Testing options  
SQL> create user tpcc identified by "tpcc";  
用户已创建。
```

```
SQL> grant dba to tpcc;
授权成功。
```

(5) 编辑 BenchmarkSQL 的配置文件,给 Oracle 数据库加载数据。

```
[oracle@oracle40 run]$ pwd
/home/oracle/benchmarksql-5.0/run
[oracle@oracle40 run]$ vi props.ora
db=oracle
driver=oracle.jdbc.driver.OracleDriver
conn=jdbc:oracle:thin:@192.168.40.40:1521:orcl
user=tpcc
password=tpcc

warehouses=10
loadWorkers=4

terminals=1

#执行加载数据的命令
[oracle@oracle40 run]$ ./runDatabaseBuild.sh props.ora
```

(6) 查看 Oracle 数据库 TPCC 用户的数据。

```
[oracle@oracle40 ~]$ sqlplus tpcc/tpcc
连接到:
Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
SQL> select table_name from user_tables;
TABLE_NAME
-----
BMSQL_DISTRICT
BMSQL_CONFIG
BMSQL_WAREHOUSE
BMSQL_CUSTOMER
BMSQL_HISTORY
BMSQL_NEW_ORDER
BMSQL_OORDER
BMSQL_ORDER_LINE
BMSQL_ITEM
BMSQL_STOCK
已选择 10 行。
SQL> select count(*) from bmsql_warehouse;
COUNT(*)
-----
10
SQL> select count(*) from tpcc.bmsql_order_line;
COUNT(*)
-----
2999807
```

## 2. 目标端实验环境部署

在目标端创建 tpcc 用户和模式,用于存放迁移后的数据。



```
[kingbase@kingbase ~]$ ksql -Usystem test
用户 system 的口令:
test=#create user tpcc with password 'tpcc';
CREATE ROLE
test=#alter user tpcc superuser;
ALTER ROLE
test=#\q
[kingbase@kingbase ~]$ ksql -Utpcc test
用户 tpcc 的口令:
test=#create schema tpcc;
CREATE SCHEMA
```



## 5.3 KDTS 的使用流程

本节以从 Oracle 数据库中迁移 TPCC 的测试数据到 KES 数据库为例,介绍数据库迁移工具 KDTS Web 版的使用方式。

### 5.3.1 访问 KDTS

进行数据迁移之前,首先需要启动 KDts 服务,如下所示。

```
[kingbase@kingbase bin]$ pwd
/KingbaseES/V9/ClientTools/guitools/KDts/KDTS-WEB/bin
[kingbase@kingbase bin]$ ls
shutdown.bat shutdown.sh startup.bat startup.sh version.bat version.sh
[kingbase@kingbase bin]$ ./startup.sh
... ..
application pid: 47553
```

Linux 平台启动 KDts 服务后程序在后台运行,如果需要查看运行日志,可使用 tail 命令: tail -f logs/kdts-app-console\_2025-12-24\_11-56-15.log。

如果需要停止运行中的程序,可执行./shutdown.sh。

启动 KDts 服务后,就可以在网络通畅的浏览器中访问该服务了,推荐使用谷歌浏览器进行访问,访问地址为 http://ip:54523/。例如,访问刚才启动的 KDts 服务,可以使用访问地址 http://192.168.40.111:54523/。

需要注意的是,虚拟机的网络需要正确配置,以保证浏览器可以访问到该虚拟机。

首先出现的是登录界面,如图 5-2 所示,默认用户名/密码为 kingbase/Kb\_DI@2019。

登录成功后显示概览页面,如图 5-3 所示。该界面以图表形式展示数据库连接总数、任务安排总数、作业执行总数,以及作业数与任务数在指定时间段内每天执行的次数,便于快速了解系统使用情况。

该界面的右上角是登录的用户名,把光标移到用户名上,出现下拉菜单,其中包括“个人信息”和“退出登录”两个选项。

单击“个人信息”,出现如图 5-4 所示的 KDTS 个人信息界面,在该界面可以修改个人信息以及登录密码。

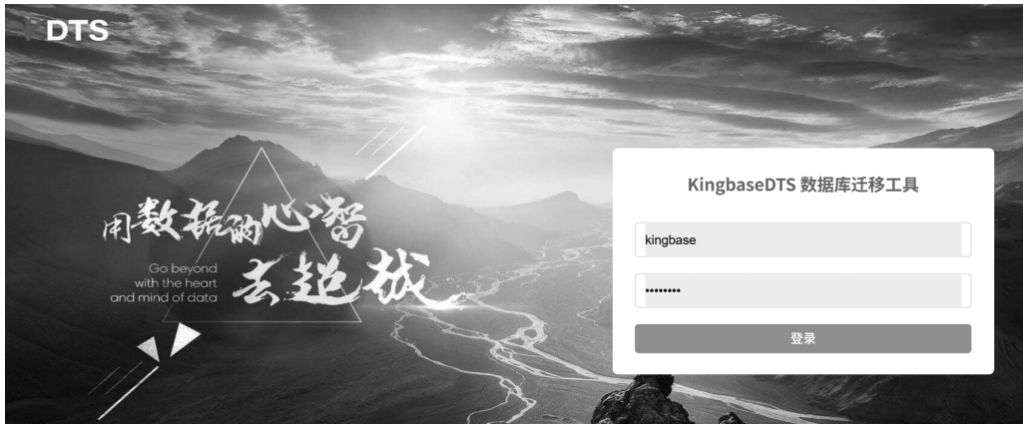


图 5-2 KDTS 登录界面

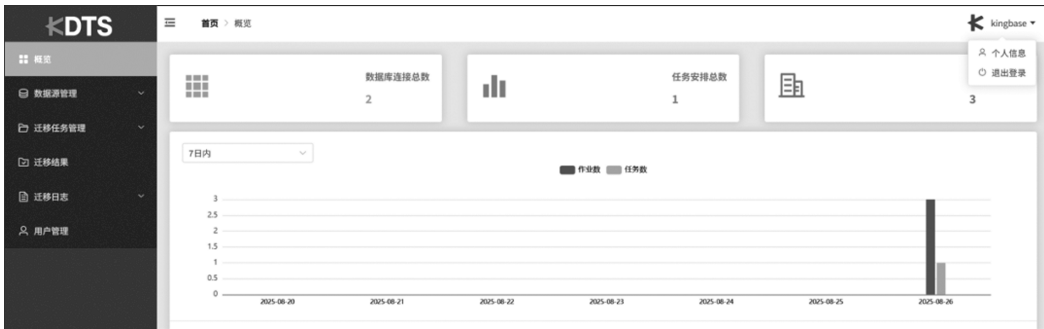


图 5-3 KDTS 概览界面

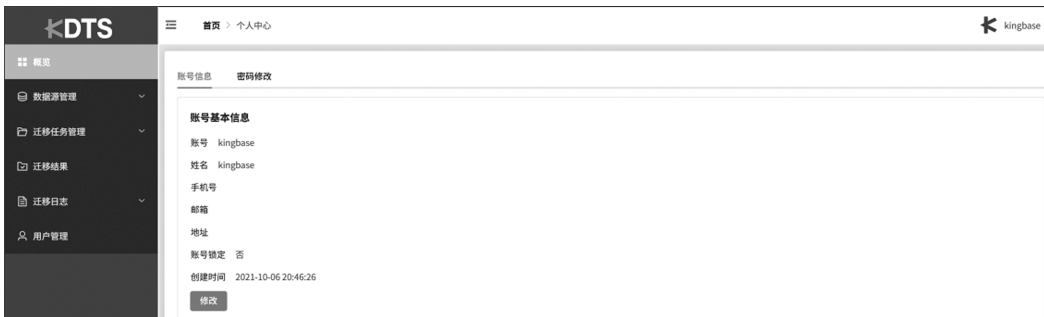


图 5-4 KDTS 个人信息界面

### 5.3.2 数据源管理

为了完成数据迁移工作,需要管理数据源信息。数据源分为源数据库与目标数据库两种,包括添加和删除数据源连接信息,添加时填写数据库基本信息等。

#### 1. 源数据库

源数据库在数据迁移时作为迁出数据的数据库使用。源数据库的管理界面如图 5-5 所示。



图 5-5 源数据库的管理界面

列表显示所建立的所有数据源连接的详细信息,包括“连接名称”“数据库”“数据库类型”“数据库版本”“服务器地址”“用户名”“操作”。目前还没有源数据库的信息。

在搜索框内输入连接名称的关键字,然后单击“查询”按钮可快速找到对应的源数据库,支持模糊搜索。如果存在大量数据源连接,“查询”按钮可用于对这些数据源连接的快速检索。

首先“新建”一个数据源,单击“+新建”按钮,出现如图 5-6 所示的界面。



图 5-6 新建源数据库连接界面

在弹出的对话框中需要输入下面的信息。

(1) 连接名称: 输入自定义的连接名称,用于标注此数据源连接,该例中填写了“oracle40”。

(2) 数据库类型: 在源数据库中支持“Oracle”“MySQL”“SQL Server”“DB2”“GBase”“PostgreSQL”“DM”“KES”数据库类型,该例选择“Oracle”。

(3) 数据库版本: 迁移数据的数据库版本,该例选择“11g”。

(4) 服务器地址: 源数据库所在服务器 IP 地址,该例填写“192.168.40.40”。

(5) 服务器端口: 源数据库端口号,该例填写 Oracle 默认的端口号“1521”。

- (6) 用户名: 操作数据库的用户名,该例填写“TPCC”。
- (7) 密码: 与操作数据库用户名相匹配的密码,该例填写用户 TPCC 的密码。
- (8) 数据库: 操作数据库的名称,该例填写 Oracle 实例名“orcl”。
- (9) 驱动: 数据库 JDBC 驱动信息。
- (10) URL: 完成“服务器地址”“端口号”“数据库”的输入自动生成 URL。
- (11) 连接参数: 根据数据库做相应适配,可自行添加或删除参数。

数据库连接信息输入完成后,可以单击“测试”按钮确认连接信息是否正确,从该例中看到测试连接成功,就可以单击“确定”按钮,保存数据库连接信息了。这时在源数据库的列表中就会出现刚才创建的源数据库连接,如图 5-7 所示。



图 5-7 源数据库连接信息

列表中每一行表示一个数据库连接,最后的字段操作中有“编辑”、“测试”和“删除”3 个按钮,可以修改、测试和删除该数据库连接信息。当需要批量删除数据源连接时,可勾选“连接名称”前的复选框选中需要删除的源数据库连接信息,单击“批量删除”即可。在未勾选任何数据库连接时,“批量删除”呈不可选择状态。

## 2. 目标数据库

管理目标数据库信息,在数据迁移时作为存放迁移数据的数据库使用。目标数据库连接信息的管理与源数据库相同,我们新建一个目标数据库 KES 的连接,连接信息如图 5-8 所示。



图 5-8 目标数据库的连接信息



### 5.3.3 迁移任务管理

创建源数据库和目标数据库连接后,就可以创建迁移任务了。迁移任务管理可以操作或查看迁移的全过程。

在 KDTS 工作台界面的左侧单击“迁移任务管理”,继续单击“迁移任务”,则会出现迁移任务的列表,如图 5-9 所示。目前还没有定义迁移任务。



图 5-9 迁移任务列表界面

#### 1. 创建迁移任务

“新建”一个迁移任务,单击“+新建”按钮,出现如图 5-10 所示的界面。

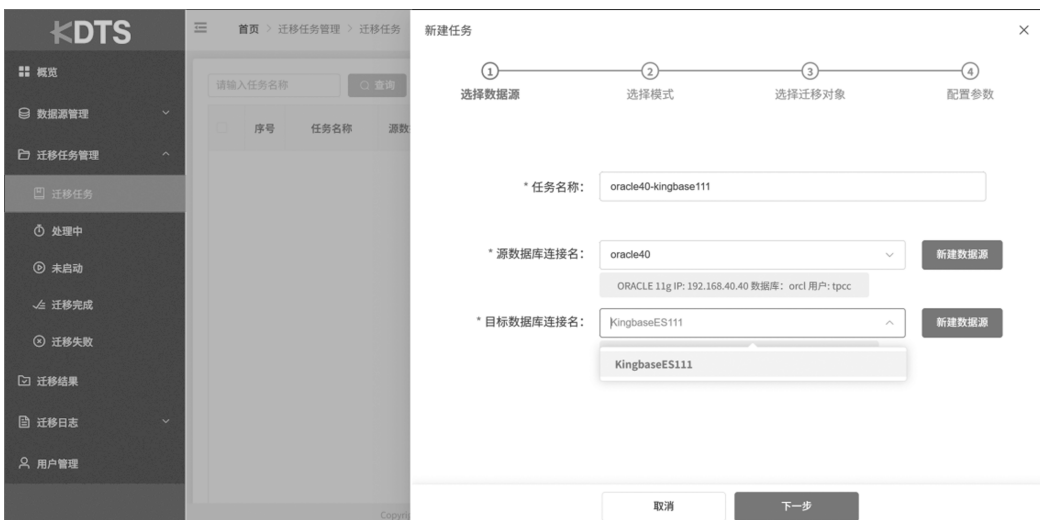


图 5-10 新建迁移任务的选择数据源界面

可以看到,新建任务包含 4 个步骤:“选择数据源”“选择模式”“选择迁移对象”“配置参数”。第一个出现的是“选择数据源”界面,填写自定义的任务名称,选择“源数据库”和“目标数据库”。

这里我们选择刚才创建的源数据库连接 oracle40 和目标数据库连接 kingbase111,然后单击“下一步”按钮,进入如图 5-11 所示的“选择模式”界面。

在该界面中,根据数据迁移所需,选择对应模式(如需选择系统模式,可勾选“包含系统模式”复选框)的表、视图、序列、函数、存储过程、程序包、同义词等。例如,可以选择只迁移表结构,不迁移表数据。还可以选择不迁移函数、存储过程、程序包等非表对象等。