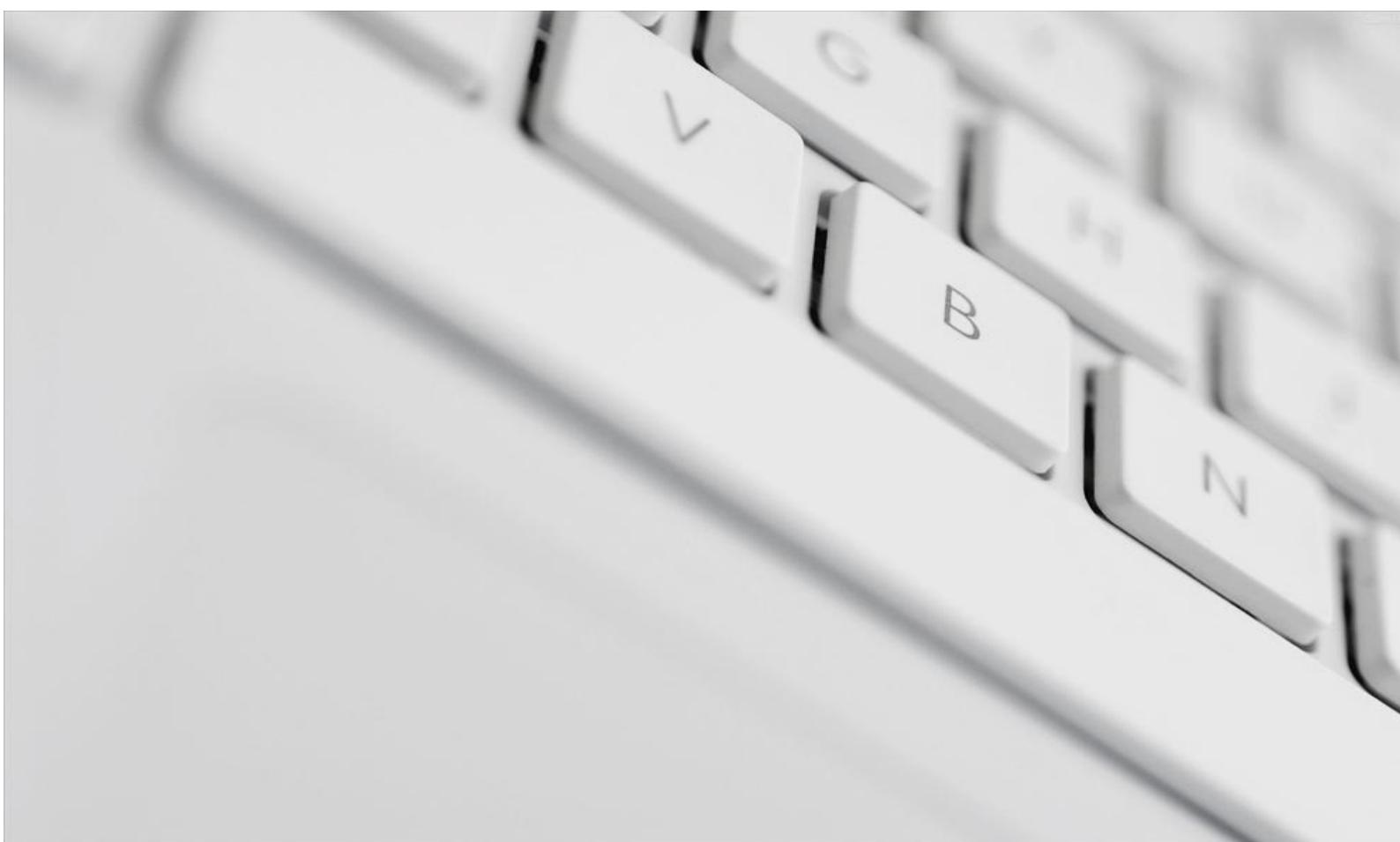


## Oracle 变化数据捕获

变化数据捕获就是我们通常提到的 CDC

（ Change Data Capture ），是用来描述捕  
捉增量变化应用数据到其他数据库或数据  
源。

- *Oracle CDC 基本原理*
- *同步 CDC 模式简介*
- *两种异步模式的创建*
- *附：同步模式在 ODI 中的体现*



# Oracle 变化数据捕获

变化数据捕获就是我们通常提到的 *CDC* (*Change Data Capture*)，是用来描述捕捉增量变化应用数据到其他数据库或数据源。随着数据量的不断增长和数据存储日益变化，变化数据捕获在生产系统中特别重要。——赵宇



化数据捕获就是我们通常提到的 CDC ( Change Data Capture ) , 是用来描述捕捉增量变化应用数据到其他数据库或数据源。随着数据量的不断增长和数据存储日益变化 , 变化数据捕获在生产系统中特别重要 , 比如做多生产中心、报表分离、容灾备份、数据仓库、数据分发等 , 特别是要求实时或近实时的生产系统中。在 Oracle 、 DB2 、 SQL Server 等数据库中 , 都可以通过数据库的日志提取变化的捕捉 , 实现变化数据的提取、传输。而在传统上我们通常都是通过修改源代码的应用 , 在一些表上增加日期列来捕获增量变化。

Oracle 数据库的变化数据捕获可以通过日志的方式实现增量变化捕获而不需要对源程序做任何更改。 Oracle 数据库从 10g R2 介绍异步分布式 CDC , 比原有 9i 数据库 R2 有了更大的提高。但是从 Oracle 11g 版本开始 , Oracle 推荐使用 GoldenGate 来做数据库之间的变化数据捕捉 , 对 CDC 的功能不再增加新特性。所以 Oracle 11g 中的 CDC 功能还是之前 10g 的 CDC 功能版本。

下面介绍如何设置这样一个异步变化数据捕获环境。

- **变化数据捕捉的用户： Publishers and subscribers ( 发布者和用户 )**

publisher 就是发布变化数据捕捉的数据库用户 . 所以需要先创建一个发布者。 subscriber 就是变化数据的订阅者 , 就是利用这个捕捉的变化数据的数据库用户。

出于安全原因，发布者和用户不应该是相同的数据库用户。一个 Publisher 可以支持许多 subscriber 用户，也就是通常我们提到的一对多复制。

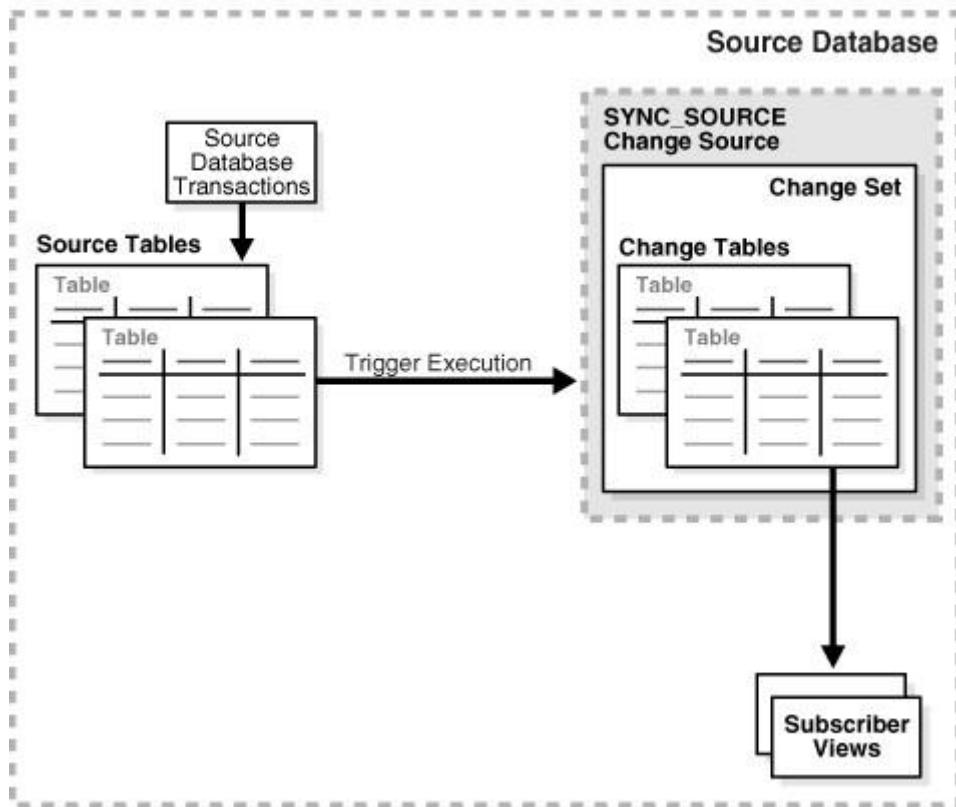
- **CDC 有两种捕捉数据方式：同步 CDC 和异步 CDC**

异步 CDC 又分 3 种模式：异步 HotLog 模式、异步分布式 HotLog 模式、异步 Autolog 模式。

## 同步 CDC 模式创建方式

同步 CDC 的模式通常都是在源数据库上配置触发器进行变化数据捕捉。它是没有任何延迟的，因为数据是连续、实时地在源系统进行捕捉。当源数据库中的 DML 事务执行时，变化数据开始被捕捉。同步 CDC 利用变化表和用户视图来实现。数据变化写入变化表为使用 CDC 提供了一个可扩展的基础架构。用户针对变化表的数据可以获得一致数据变化集的视图。用户可以扩展和清除订阅窗口，通过数据库视图，隐式地改变数据集。如果没有任何用户订阅变化数据，变化表中的数据可以被清除。同步 CDC 通过 PL/SQL 包进行调用。同步 CDC 的优势在于配置简单，不需要配置提取、传输、复制等功能。

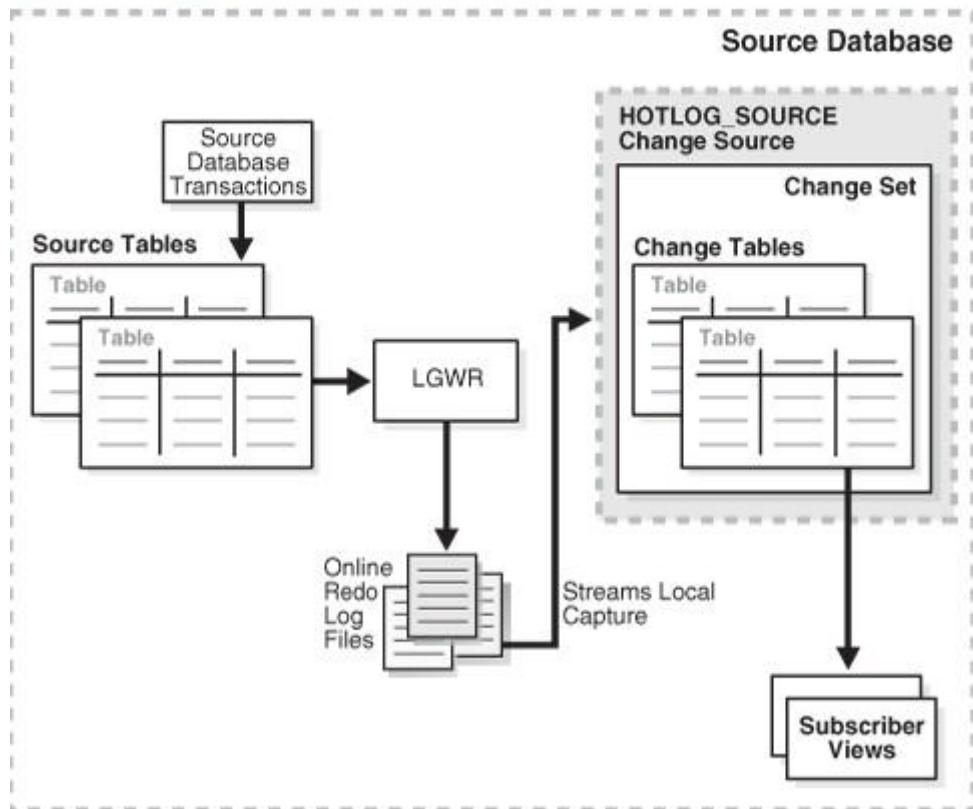
Figure 17-3 Synchronous Change Data Capture Configuration



## 异步 HotLog 模式创建方式

在异步 HotLog 模式中，变化数据捕捉是从源数据库的 online redo log 提取事务数据。这个捕捉有一些延迟在数据提交和写入日志的时间差上。因为我们要设置 Change Table 来存取变化的数据，所以这个 Change Table 一定要在源数据这端。F17-4 图就是一个简单的 HotLog 配置，LGWR 记录提交的事务到 online redo log files 中，用 Oracle Stream 方式自动在 HotLog Source 中填入变化数据到 Change Set 中的 Change Table。

Figure 17-4 Asynchronous HotLog Configuration



## 异步分布 HotLog 和 AutoLog 模式

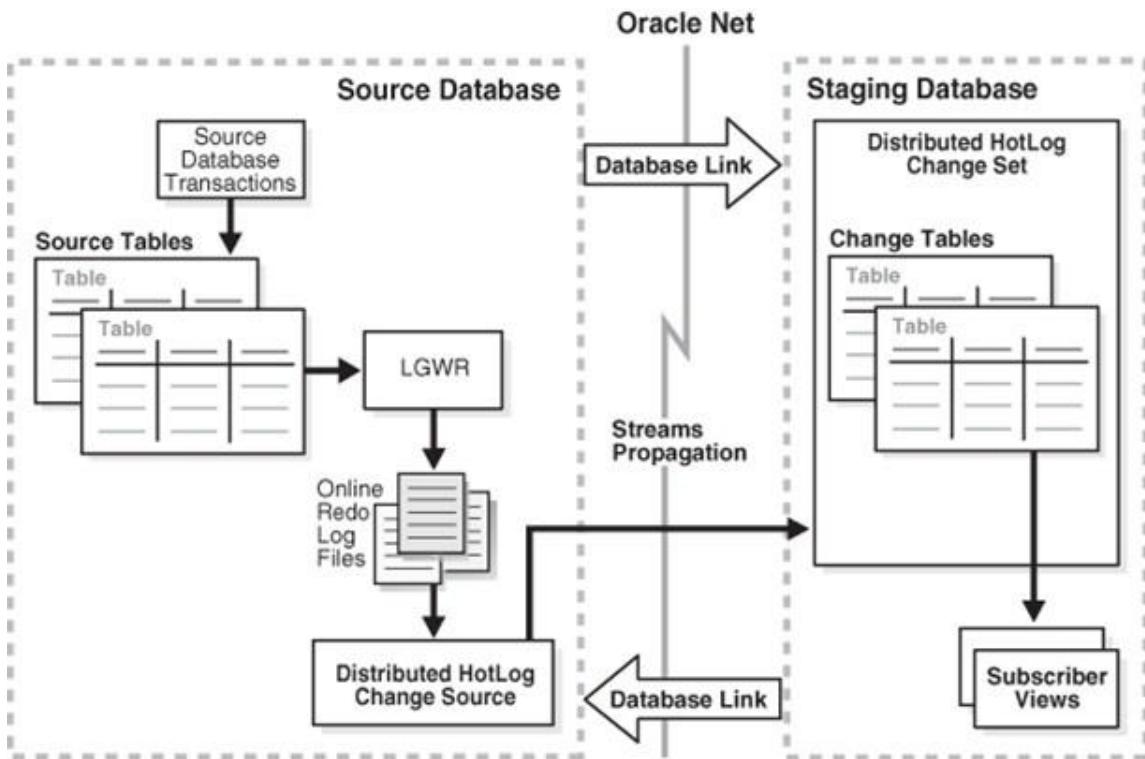
在一个异步分布 HotLog 模式下，变化数据捕捉是从源数据库的 online redo log file 中。不像之前介绍的模式，异步分布 HotLog 模式中变化数据捕捉的动作和对象跨了源数据库（Source Database）和中间库（Staging Database）。变化源（Change sources）在源数据库定义，在中间数据库发布。F17-5 图说明异步分布式配置，源数据库的 Change Source 从 online redo log files 捕捉变化的数据，用 Stream 方式填充到中间库的变化集中。这个变化

集 ( Change Set ) 会把数据存到变化集中的变化表 ( Change Table )。

这个配置中，有两个发布者，一个是在源数据库、一个是在中间数据库。源数据库发布者定义一个 DBLINK 连接到中间数据库作为中间数据库的发布者，同时在中间数据也定义一个反向的 DBLINK 连接到源数据库作为源数据库的发布者。所有发布操作都是通过中间数据库发布者执行的，也就是源数据库操作引起的。

为什么变化数据捕捉单向复制，还要配置一个反向的 DBLINK 呢？这个 DBLINK 是必须创建的，主要是为了 Create、Alter、or Drop Distributed HotLog Change sources，Change sets 和 Change Tables 使用。但是，这个 DBLINK 在变化捕捉的时候并不起作用。一旦 Distributed HotLog change sources，change sets 和 change tables 启用以后，这个 DBLINK 是可以被 drop 掉。如果要重建 Distributed HotLog change sources，change sets 和 change tables，那需要重新启用这个 DBLINK。

Figure 17-5 Asynchronous Distributed HotLog Configuration



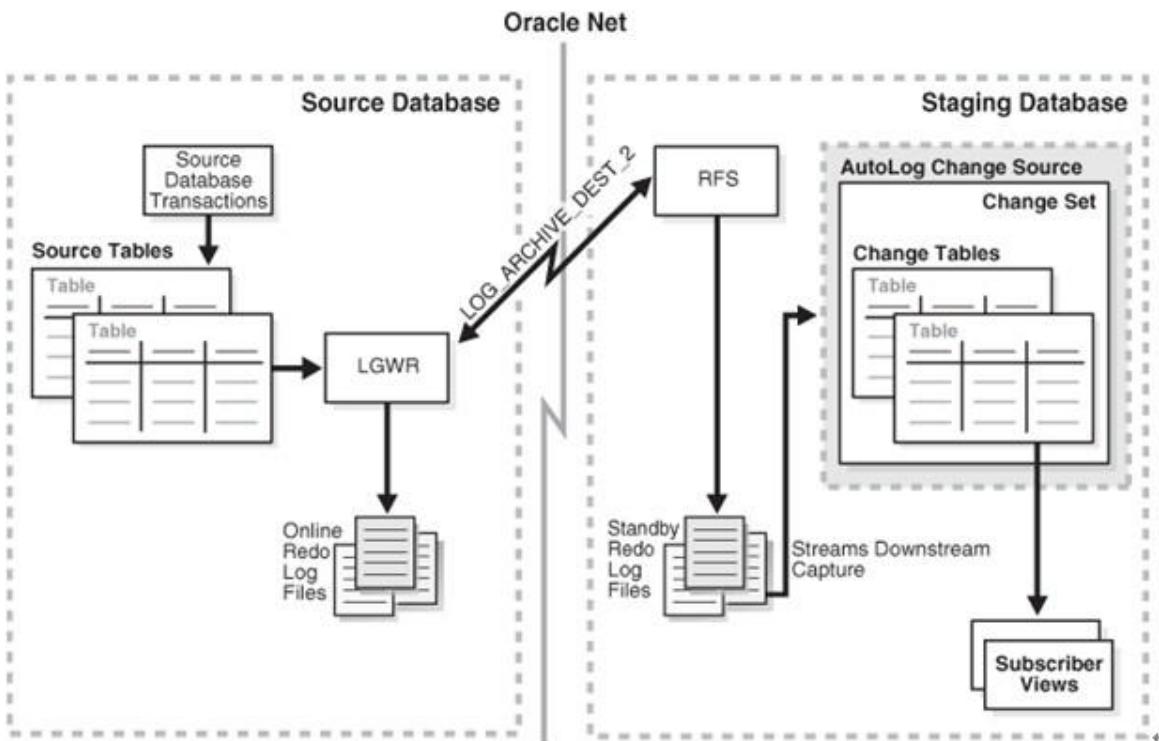
在异步 AutoLog 模式下，数据变化通过 redo transport services 服务从 redo log files 或者 archived log files 进行捕捉。Redo Transport Services 自动捕捉日志从源库到中间库。配置这种模式，我们需要设置数据库参数 LOG\_ARCHIVE\_DEST\_2 , LOG\_ARCHIVE\_DEST\_1 做本地日志的归档目录。源数据库的发布者 ( Publisher ) 配置日志传输服务，然后从源数据库拷贝 Redo Log Files 到中间库，在中间库自动地注册 Redo log Files 进行数据同步。AutoLog 模式从 redo log files 捕捉就是异步 AutoLog Online 模式，如果从 Archived Log 捕捉就是异步 AutoLog Archive 模式。如果设置了 AutoLog Online 模式，日志传输服务会从源库拷贝 Redo Log 到中间库或者目标库的

standby redo log。源库事务提交后，中间库的变化集被填充数据。在中间库，只能有一个 AutoLog Online Change Source，也只能包含一个 Change set. 如果设置了 AutoLog Archive 模式，日志传输服务是拷贝归档日志到中间库，等中间库接受到归档日志后，Change set 才被填充。

F17-6 图显示的是 AutoLog Online 配置模式，LGWR 进程拷贝 redo data 到中间库的 Standby Redo Log。LGWR 进程是用 Oracle Net 网络发送数据给中间库的 RFS 进程。中间库的 RFS 进程会写 redo data 到 standby redo log. 变化数据再通过 Oracle Stream downstream 的方式填充变化数据到 change tables 中。

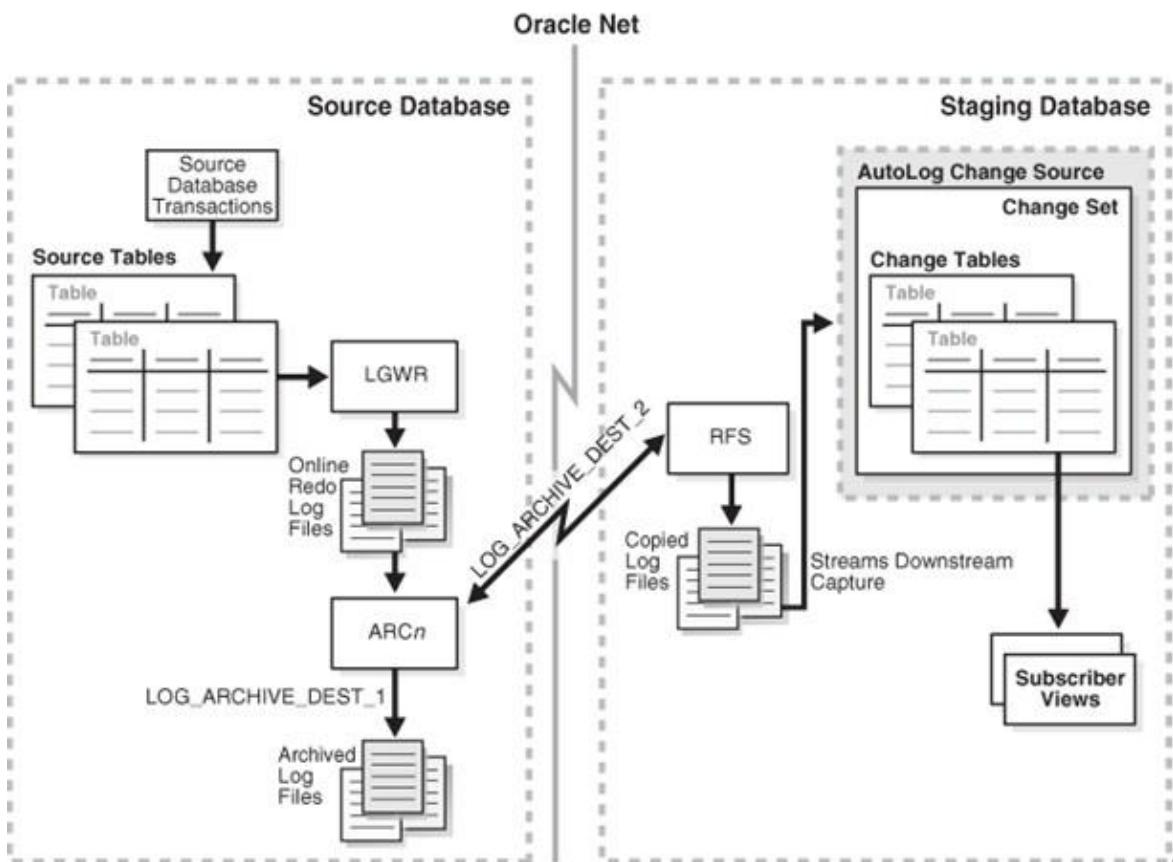
其实，我们可以发现这个日志传输和应用的模式，和我们的 Data Guard 配置非常相似，只是多了一步 Stream 的配置过程。

Figure 17-6 Asynchronous Autolog Online Change Data Capture Configuration



F17-7 图显示 Redo Log Files 切换后，归档进程归档 Redo Log 到 LOG\_ARCHIVE\_DEST\_1，然后再拷贝数据到 LOG\_ARCHIVE\_DEST\_2 指定的中间库。归档进程也是通过 Oracle Net 发送 redo data 给 RFS 进程。在中间库，RFS 进程写 Redo data 到被拷贝的日志文件。再通过 Oracle Streams downstream 传输变化到 change tables。

Figure 17-7 Asynchronous AutoLog Archive Change Data Capture Configuration



附：CDC 不同模式在 ODI 中体现系列之同步模式

在本次的 Oracle 电子书中，我们将介绍 Oracle 变化数据捕捉中，同步模式下的 ODI 体现。如果想要了解异步模式的体现方式，TT 数据库将在下一期的电子书中为您奉上，敬请关注！

## 两种模式对比

同步 CDC 模式 ( Synchronous Mode ) 通过在源数据上建立 trigger 的方式来捕获增量数据 , 因此可以做到实时抽取增量数据 , 如果在晚上定义作业来完成 , 对源系统的影响不大。

异步 HotLog 模式 ( Asynchronous HotLog Mode ) 直接从 source database 的 online archive log file 中抽取增量数据 , 由于需要解析日志文件 , 会有一定的时间延迟。变化表 ( change table ) 也必须在源库中生成。该模式由于是在源数据库中解析日志 , 对源数据库也会造成一定的压力 , 但是可以选择在非业务高峰期完成。

异步分布式 HotLog 模式 ( Asynchronous Distributed HotLog Mode ) 和异步 HotLog 模式相比 , 主要是将多个 source database 的 archive log 解析出增量数据 , 然后传递一个 staging database 中处理 , 便于集中式数据管理。 Staging Database 实际上起到一个接受数据源 , 然后做数据传递。

异步 AutoLog 模式 ( Asynchronous AutoLog Mode ) 则是直接把 source database 的 online redo log 传递到 staging database , 然后在 staging database 的 online standby redo log 执行日志分析 , 这样的架构对源库性能影响最小、数据传递性能最大。

## CDC 同步模式需要配置两个用户

Publishers and Subscribers (发布用户和订阅用户)

publisher 就是发布变化数据捕捉的数据库用户.所以在 ODI 需要先创建一个发布者。subscriber 就是通过所谓的订购，利用这个捕捉的变化数据的数据库用户，也可以理解为订阅者。出于安全原因，发布者和订阅用户不应该是相同的数据库用户。一个 Publisher 可以支持许多 subscriber 用户，也就是可以把数据传给多个订阅用户。

CDC 利用变化表和用户视图。数据变化写入变化表为使用 CDC 提供了一个可扩展的基础架构.用户针对变化表的数据可以获得一致数据变化集的视图。用户可以扩展和清除订阅窗户，通过数据库视图，隐式地改变数据集。如果没有任何用户订阅变化数据，变化表中的数据可以被清除。CDC 也通过 PL/SQL 包进行调用。

## 开始同步 CDC 模式创建

1、创建 Publisher 用户，在 sys 用户中创建变化表集（ change set ），然后在变化集中创建相应的变化表（ change table ），一个 change set 可以对应多个 change table。

2、创建 Subscriber 用户， publisher 用户把 change table 访问权限授予 subscriber 用户。

3、Subscriber 可以创建 subscription ( 订阅 ) ，通过创建 subscribe 可以订阅 change set 中多个 source table ，这个 table 是 change table 的源表。

4、激活订阅 activate subscription。

5、扩展窗口 extend\_windows ，扩展窗口以后才能看到数据变化.如果想每次都看到数据的变化，必须每次扩展窗口。

上面的 5 个步骤，是在 Oracle Database 中实施的步骤，ODI 实际上也是调用数据库的功能，做成 KM ( 知识模块 ) 来实现，那么在 ODI 中如何配置呢？下面我们来具体的实施方法。

因为同步模式 CDC 是通过触发器的方式来实现，所以数据库可以运行在非归档的模式下，如果是其他的方式，最好配置归档模式，但是如果调用 Goldengate KM，也可以直接挖掘 online redo log。

## ● 步骤一



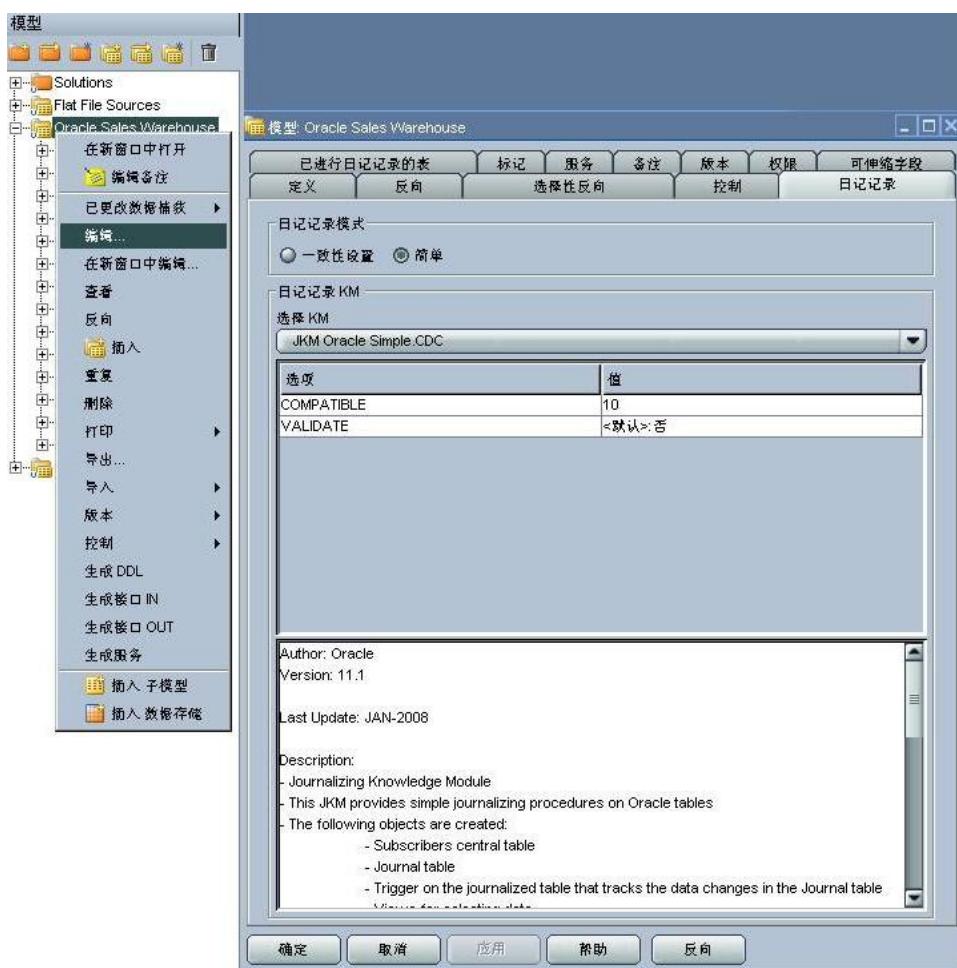
在 ODI 中 CDC 是通过日志知识模块来实施的,所以我们首先要引入知识模块。

在 ODI 项目中选择日记 JKM ( Journal Knowledge Management ) , 然后选择右键,导入知识模块,因为我们是用同步模式 CDC , 选择"JKM Oracle Simple"知

识 KM 就可以，我们还可以看到 JKM Oracle Consistent，再后面的异步 CDC，我们会引用进来。

## ● 步骤二

在模型属性中选择日志记录 TAB 页，然后选择简单，选择日志记录 KM 时，可以看到刚才我们添加的 KM，JKM Oracle Simple.CDC，这个 KM 下面有说明，不适合频繁的操作，不会对生产系统带来价值，反而影响性能。因为这个 JKM 是触发器的形式。



这个时候，我们在模型中也已经定义好了 JKM，下面就可以在模型的 Data Store 中进行添加到 CDC，我们也可以针对整个模式做 CDC，这里我们选择单独的 Data Store。

我们先查看系统已有的触发器：

```
SQL> conn odi_tmp/odi_tmp
```

已连接。

```
SQL> select * from user_triggers;
```

未选定行

```
SQL>
```

可以看到目前开发用户下面是没有触发器的。

选择 JKM 以后，点击应用。

后面，我们可以通过数据库确认 ODI 在数据库中建立了触发器。

### ● 步骤三

开始添加 Data Store 到数据源



添加数据源到 CDC 以后，会看到 CDC 左上角出现一个表的橙色图标，表示已经加到 CDC，但是还没有启动。

## ● 步骤四

### 启动日志



启动日志以后，我们需要定义 ODI 的订阅者，也就是需要捕捉这个日志数据的用户，通常就是我们数据仓库数据库。



启动日志会弹出执行对话框，选择相应的上下文，如果切换到生产库，切换上下文就可以（在 ODI 架构文章中已经介绍到上下文，就是这个用途），代理的话，可以选择本地代理，如果不是在本机器上运行，选择远端代理。

## ● 步骤五

开始订阅日志，选择订阅用户



订阅时，因为我们用 SUPERVISOR 用户登录，所以系统会默认用 SUNOPSIS 用户，这个用户是 ODI 管理的超级用户，如果想用其他用户执行，需要在 Security Model 做用户分配和权限配置。



点击确定以后，可以看到 Data Store 的左上角有个绿色时钟的图标，表示已经成功启动 CDC。

## ● 步骤六

查看 CDC 的订阅状态



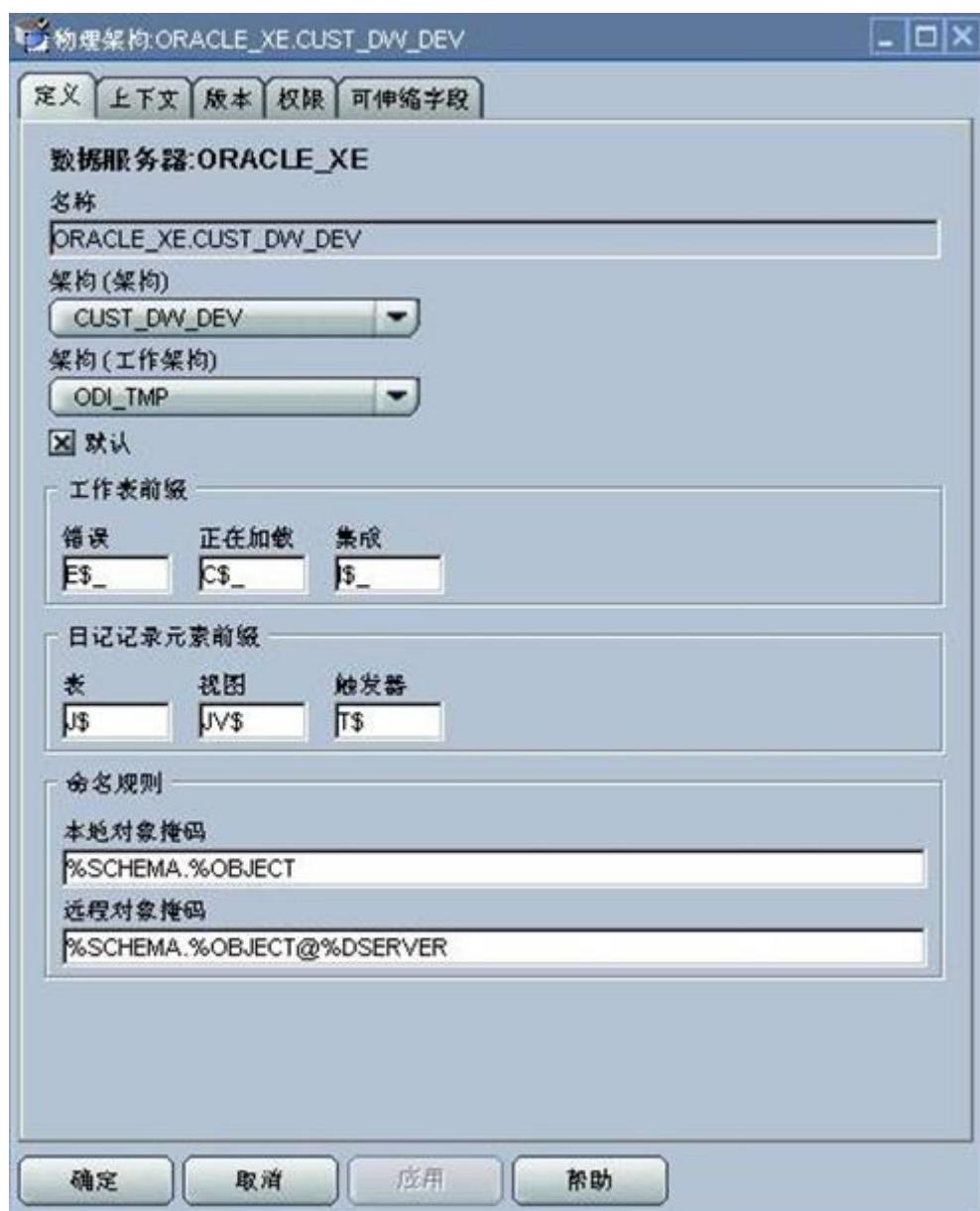
绿色图标表示已经成功订阅。

## ● 步骤七

确认数据库中是否也用相应的触发器。

这个时候，再次查看 ODI\_TMP 下面的触发器，为什么要查看 ODI\_TMP 下面：

因为我们在 Topology Manager 当中定义 ORACLE 连接的时候，就已定义了工作资料库的用户，默认是和架构用户一样，所以我们可以看到选择工作资料库用户的好处。下面是我们在 ODI Topology Manager 中的配置。



这时可以看到触发器已经成功被 ODI 创建：

```
SQL>select trigger_name,trigger_type,table_name,status,action_type from user_triggers;
```

TRIGGER_NAME	TRIGGER_TYPE	TABLE_NAME	STATUS	ACTION_TYPE
T\$CUSTOMER	AFTER EACH ROW	CUSTOMER	ENABLED	PL/SQL

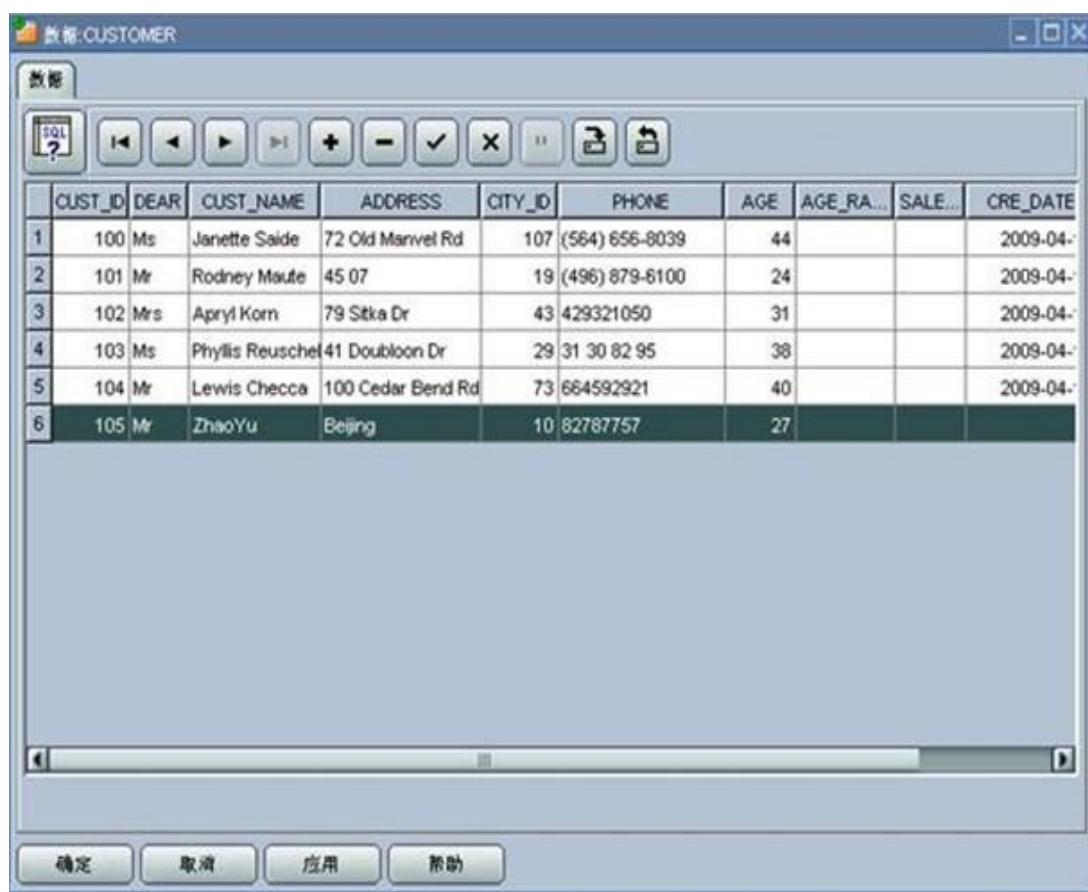
## ● 步骤八

测试 CDC 同步模式的数据

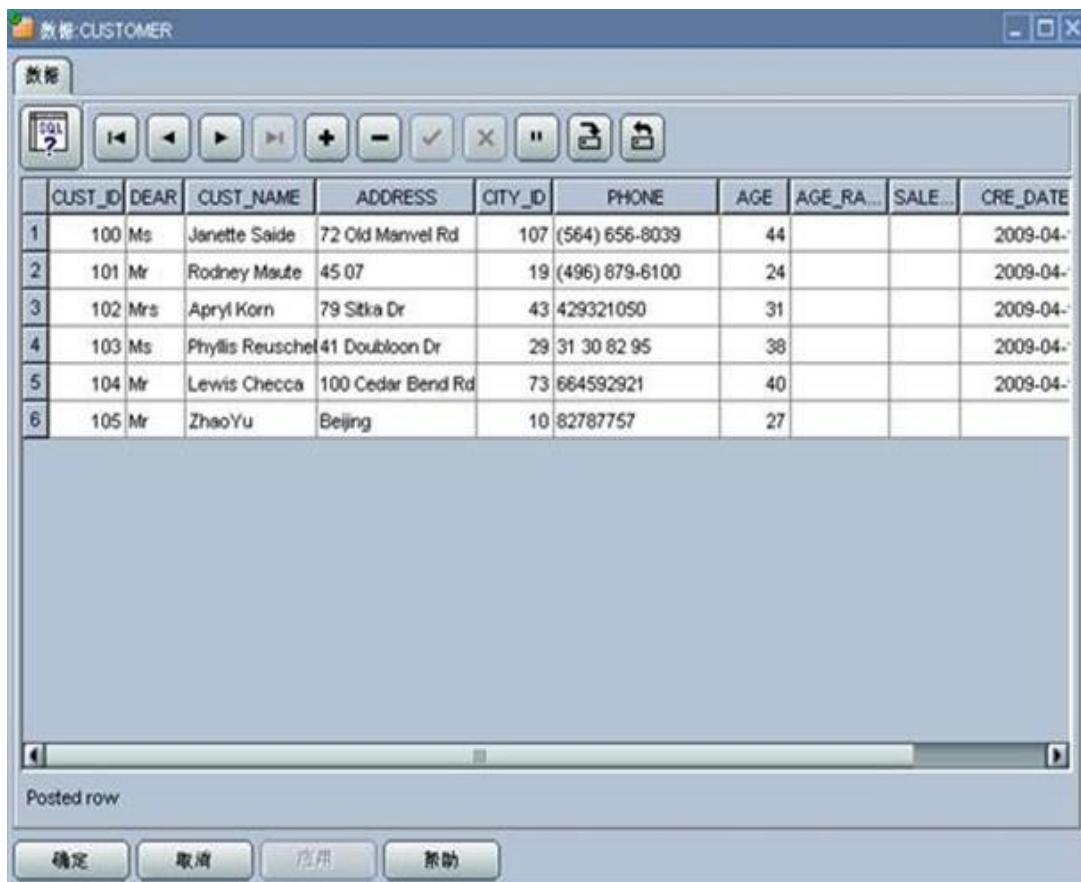
我们右键选择数据模型以后，点击数据，进行数据的编辑。



点击+号，我们添入一条数据



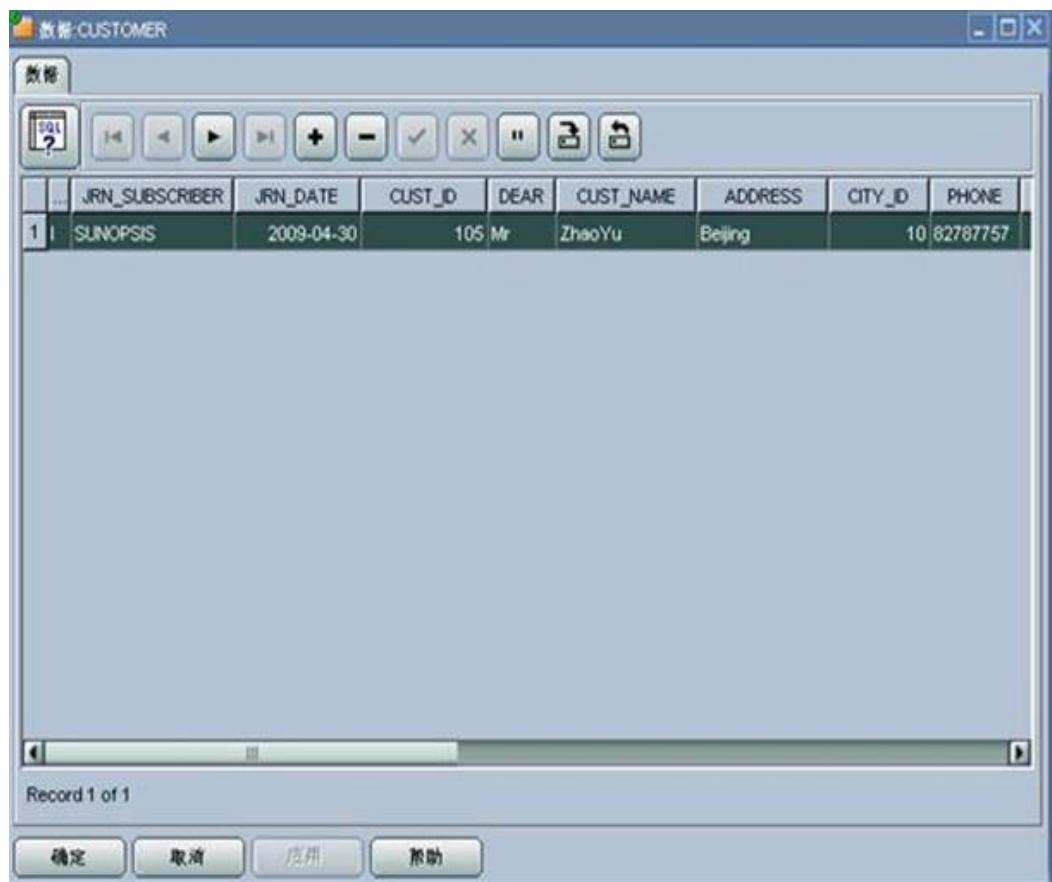
点击应用以后，我们可以看到数据已经加入到源表。



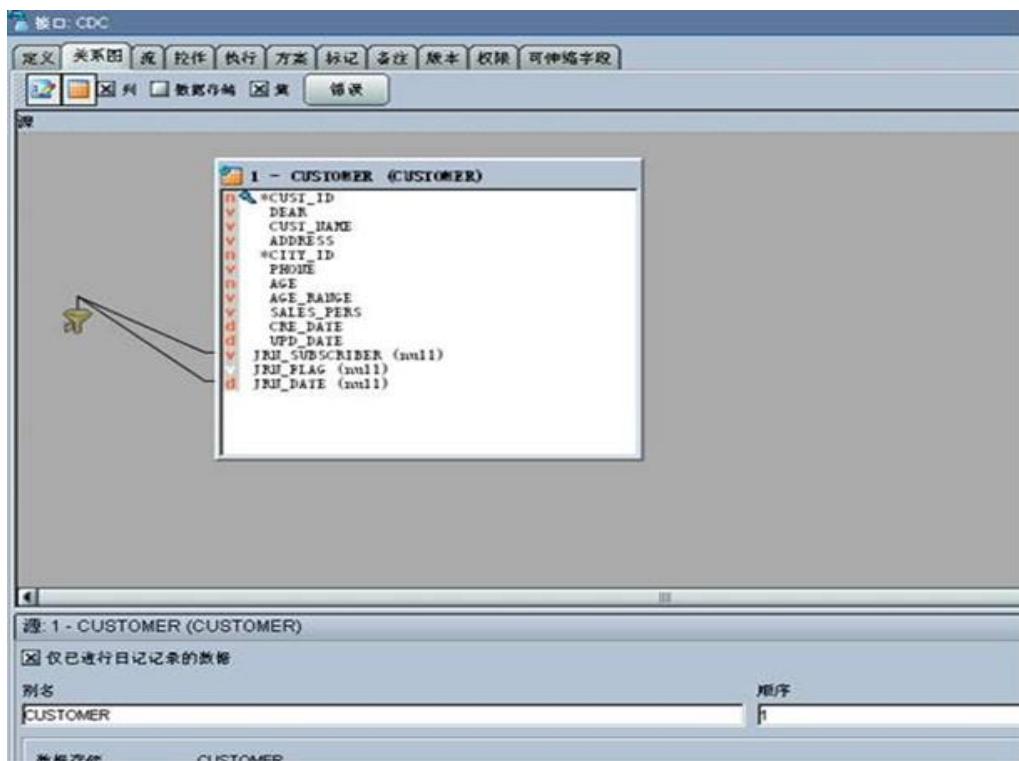
点击日志数据，查看 CDC 是否已经成功捕捉数据库表的数据。

## ● 步骤九

查看 ODI 是否捕捉数据到变化表。



我们在接口把同步 CDC 加入到接口中，接口是配置源数据库和目标数据库的列匹配和数据同步的设计。



我们点击源数据源下面有个“仅已记录日记记录的数据”前面的方框，点击以后，就把 CDC 成功添加到接口了，执行这个接口时，就会看到数据的变化。CDC 启动以后，只要源端的数据有变化，通过代理调度接口，数据就会实时同步到目标数据库。

## 我们的编辑团队

您若有何意见与建议，欢迎[与我们的编辑联系](#)。

诚挚感谢以下人员热情参与 TechTarget 中国《Oracle 系列电子书》的内容编辑工作！

诚邀更多的数据库专业人士加入我们的内容建设团队！



赵宇

TechTarget中国特邀技术专家。05年开始研究Oracle, 从事过电信级的数据库开发和管理工作。目前专注于ODI、GoldenGate等新产品技术，长于为客户做性能优化和问题诊断。