北京邮电大学

实验报告

课程名称：数据库系统原理

实验名称：数据库表/视图的创建与维护实验

计算机系2015211312班 姓名刘佳鑫

计算机系2015211312班 姓名万诗婕

教师吴起凡 成绩\_\_\_\_\_\_\_\_\_

2018年4月18日

### 实验目的

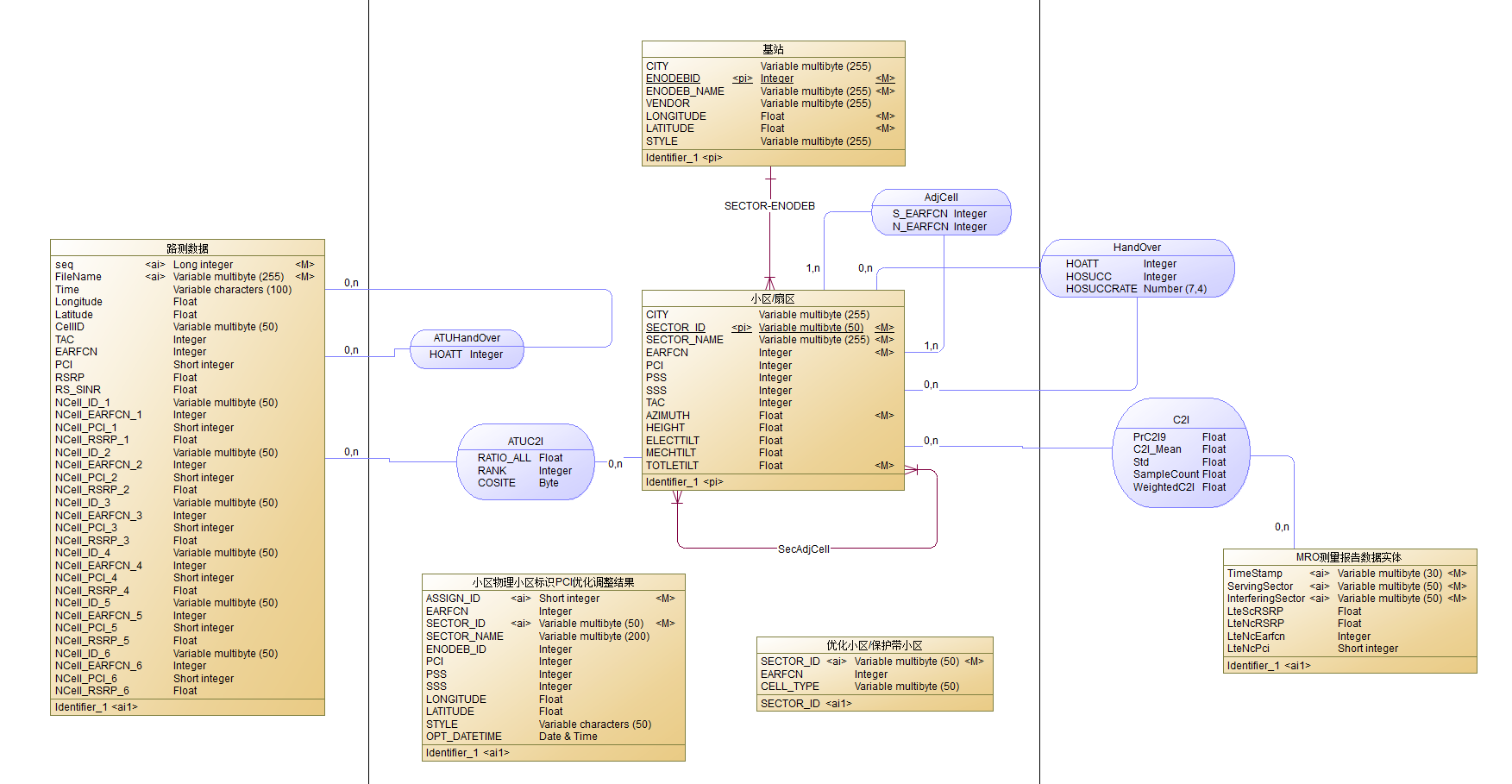
1. 掌握将E-R图转换为数据库逻辑模式（关系表）的方法。
2. 通过进行数据库表的建立操作，熟悉并掌握在Microsoft SQL Server数据库中建立表的方法，理解关系数据库表的结构，巩固SQL标准中关于数据库表的建立语句。
3. 通过对Microsoft SQL Server数据库中建立、维护视图的实验，熟悉Microsoft SQL Server数据库中建立和维护视图的方法，理解和掌握视图的概念。
4. 掌握从Excel表向Microsoft SQL Server关系数据库导入数据的方法，利用实际数据建立TD-LTE网络配置数据库。

### 实验环境

采用Microsoft SQL Server数据库管理系统作为实验平台。

### 实验内容与步骤

1. 阅读《TD-LTE网络配置数据库》课程实验背景资料及数据建模-17-v3.doc。
2. 将其中数据需求转化为E-R图。

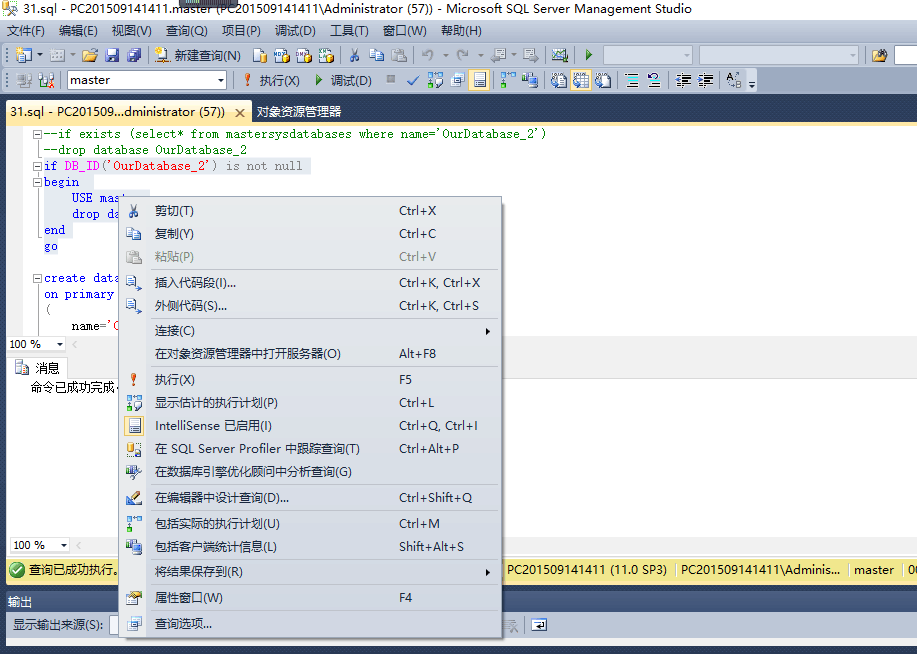


1. 将E-R图转化为逻辑模式（确定表结构及其属性，特别要注意标明其主键、候选键、外键等约束关系）。
2. 进一步明确数据类型等，将逻辑模式转化为物理模式。

方法有两种：

* 1. 写好SQL脚本文件（包括多个**create table、create view**等命令），一次性生成这些表和视图，在Microsoft SQL Server中打开SQL脚本文件并执行。
  2. 一条一条SQL语句执行。

因为这是非常基本的操作，所以一定要了解SQL语句，而不能全部使用图形界面功能来完成。



**5. 视图定义**

1. 创建单表上的视图：
2. 在表“小区/基站工参表”上创建“频点为38400的小区的基本信息”的视图tbcell38400,

属性包括（小区ID，基站ID，小区配置的频点编号，物理小区标识，小区所属基站

的经度，小区所属基站的纬度，基站类型）。

b. 在表“MRO测量报告数据表”上创建“服务小区参考信号接收功率大于40”的视

图MROInfo，属性包括（服务小区/主小区ID，干扰小区ID，服务小区参考信号接

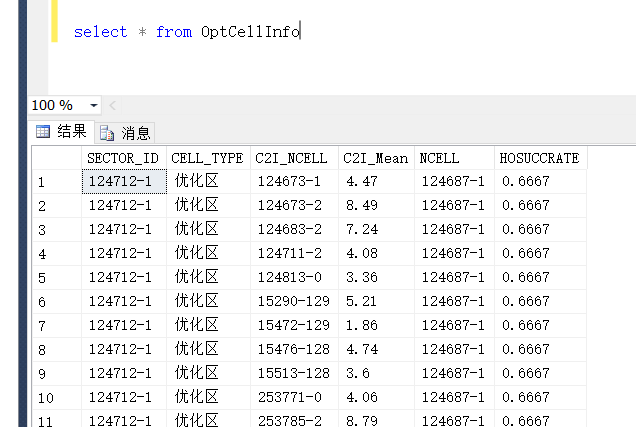
收功率RSRP）。

1. 创建多表上的视图：

根据优化小区表、基于MR测量报告的干扰分析表、小区切换统计性能表，创建“小

区类型为优化区的小区的邻小区ID，C2I干扰的均值、切换目标小区ID、切换成功率”

视图OptCellInfo，属性包括（小区ID，小区类型，C2I干扰邻小区ID，C2I干扰的均

值、切换目标小区ID，切换成功率）。

1. **几个重要的涉及表、视图的SQL语句**
   1. 查看在表或视图中的所有列、在表中找到的该表上的所有索引。
   2. **alter table**此语句用于修改表定义、禁用相关视图或使表能够参与复制服务器复制。
   3. **alter view**此语句用于用修改的版本替换视图定义。
   4. **drop table**此语句用于删除表对象。
   5. **drop view**此语句用于用于删除视图。
2. **数据导入**
3. 将Excel表向Microsoft SQL Server关系数据库导入的方法

1> 右击所选数据库，选择“任务”--->导入数据，打开sql server导入导出向导；

2> 根据向导选择数据源为EXCEL文件；

4> 修改目标数据源的数据库为要导入的数据库；

5> 修改表名为要导入的表；

6> 编辑映射，修改数据类型（这一步可忽略，可在导入数据之后修改）；

7> 点击下一步直至完成；

8> 导入成功。

1. 将csv文件向Microsoft SQL Server关系数据库导入的方法

1> 右击所选数据库，选择“任务”--->导入数据，打开sql server导入导出向导；

2> 根据向导选择数据源为平面文件源；

3> 选好要导入的csv文件后，勾选“在第一个数据行中显示列名称”；

4> 修改目标数据源的数据库为要导入的数据库；

5> 修改表名为要导入的表；

6> 编辑映射，修改数据类型；

7> 在“查看数据类型映射”页面中，将“出错时”那一列由“使用全局”改为“忽略”；

8> 点击下一步直至完成；

9> 导入成功。

数据导入过程分析

**6.1 无问题表格数据导入**

tbAdjCell

tbATUHandover

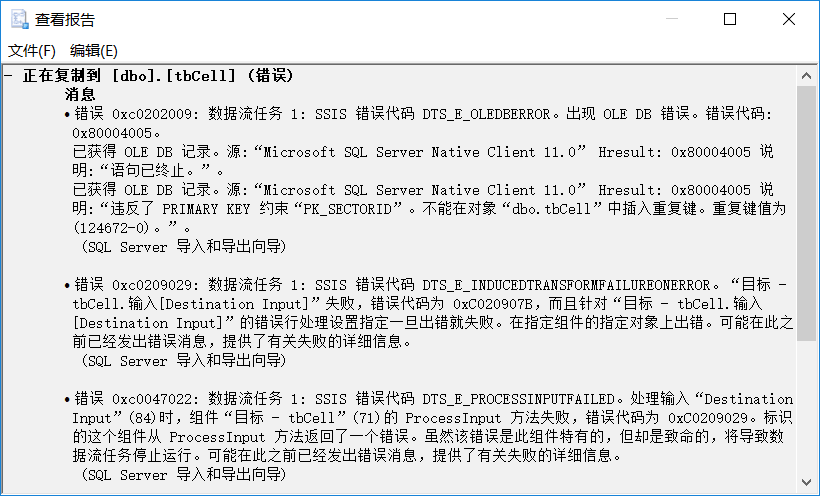
tbATUC2I

tbC2I

**6.2 出现问题的表格数据导入**

**6.2.1 tbCell:**

**报错1：**



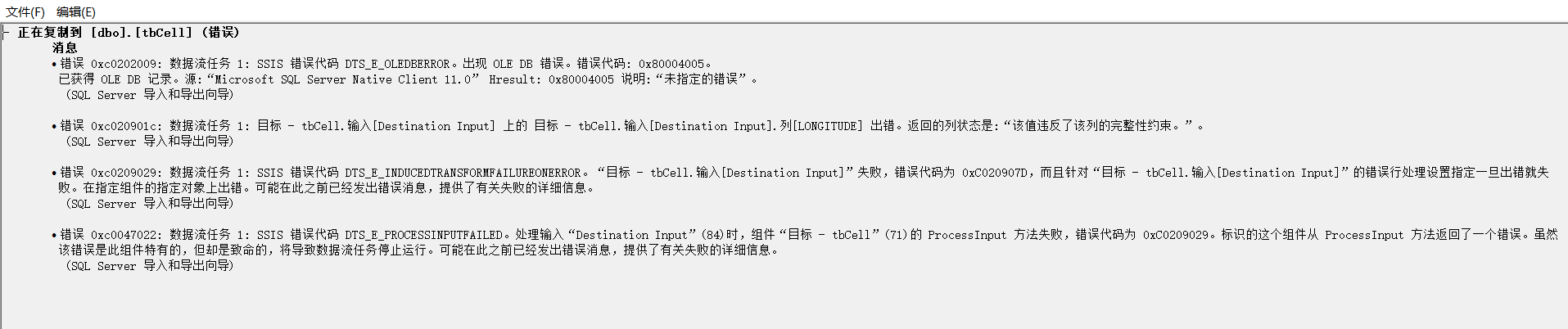
原因分析：

之前导入过一次表，修改之后没有将已经导入部分删除，导致出现重复的值

解决方法：

每一次重新导入数据前加上：truncate table tbCell

**报错2：**





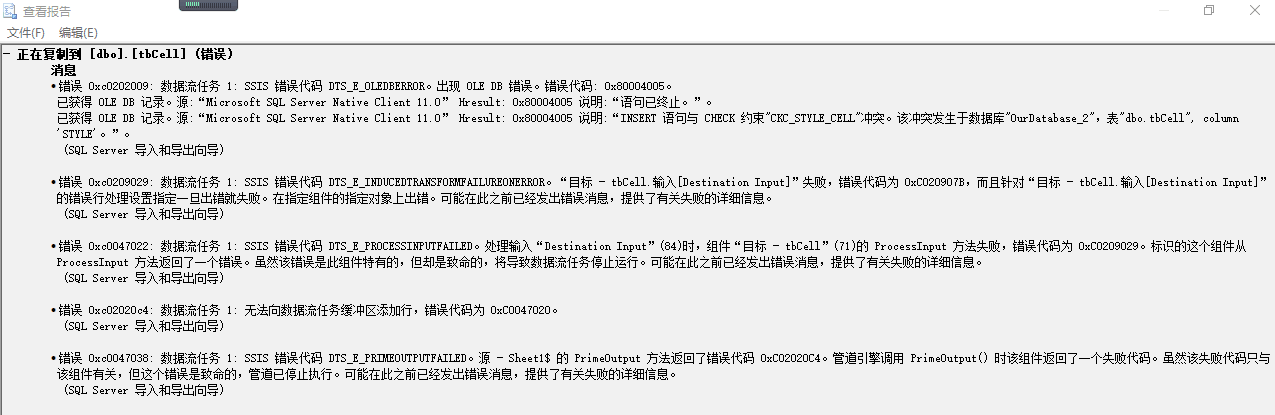
原因分析：

LONGITUDE和LATITUDE两列的格式不对

解决方法：

将这两列的格式改为“文本”

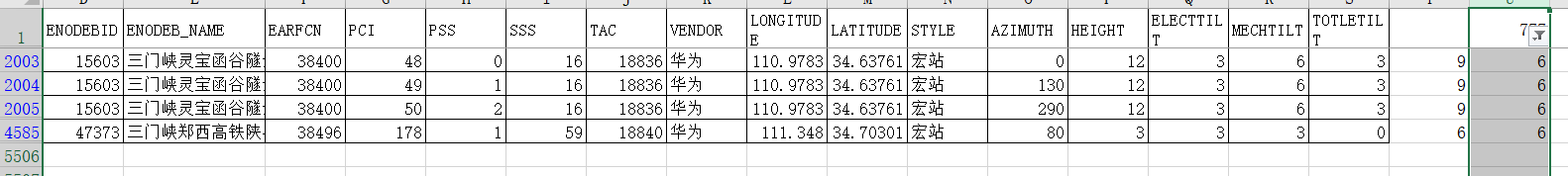
**报错3：**



原因分析：

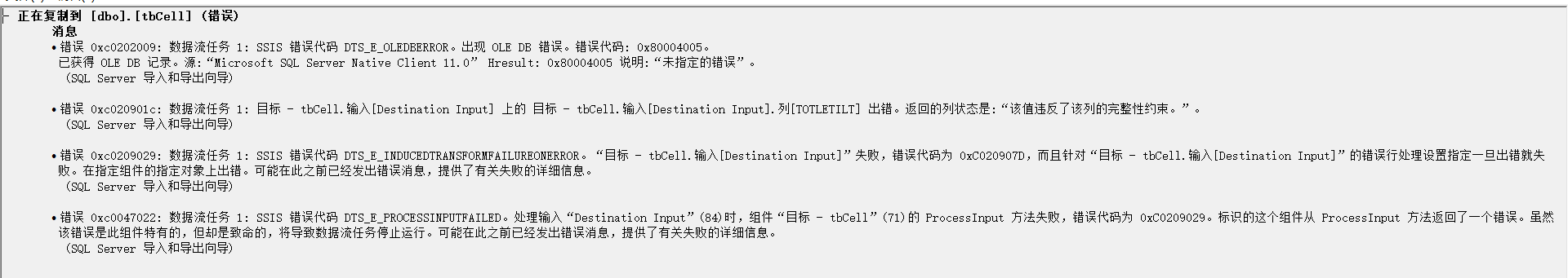
有四列数据不满足约束：

constraint CKC\_TOTLETILT\_CELL check(TOTLETILT = ELECTTILT + MECHTILT)



解决方法：

将错误的列值手动改成符合约束的即可

**报错4：**

原因分析：

表格后面有空值，而约束中要求“not null”

解决方法：

将TOTLETITL的属性改为null，因为表格中的数据后面是空值

**报错5**



原因分析：

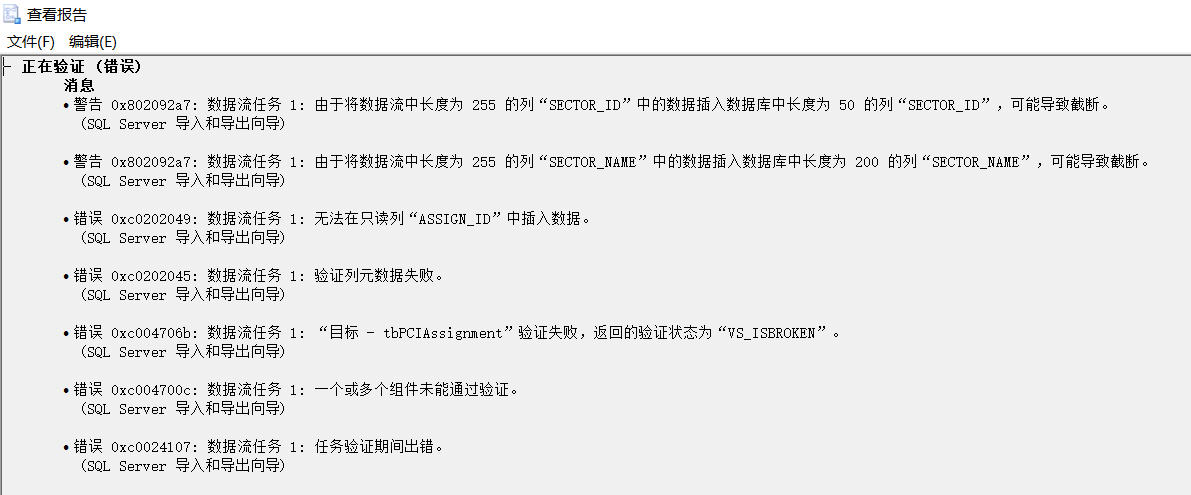
数据中STYLE类型为“宏站”，“室分”，而给的约束是“宏站”，“室内”，“室外”

解决方法：

将STYLE属性的约束中加上“室分”。

**6.2.2 tbPCIAssignment**

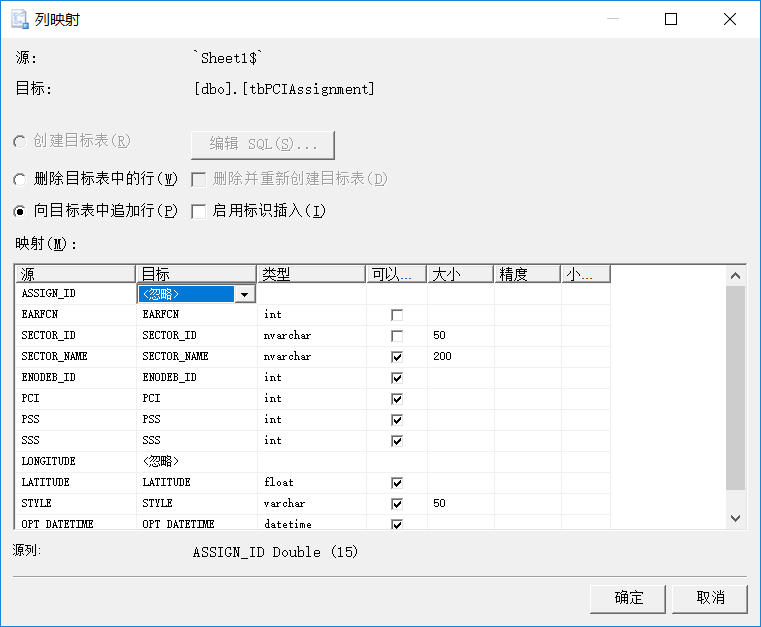
**报错1：**



原因分析：

ASSIGN\_ID约束设为自增，为只读属性，不可写

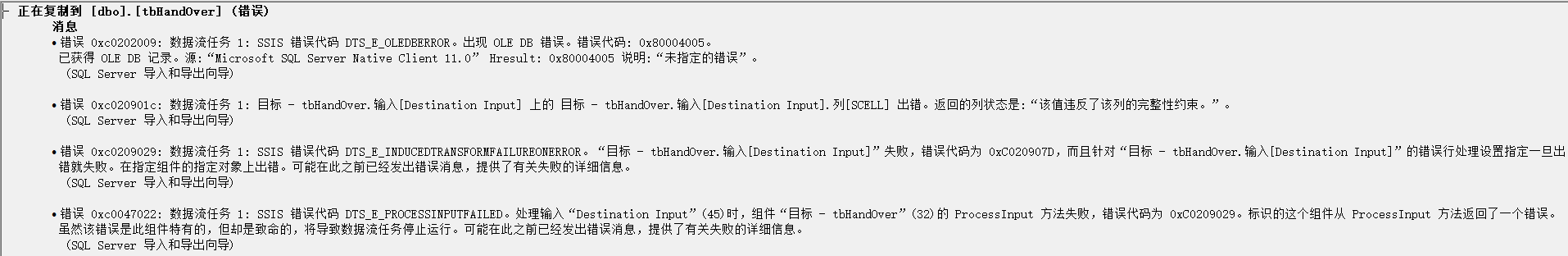
解决方法：



将ASSIGN\_ID的目标改为【忽略】，不导入这一列即可

**6.2.3tbHandOver**

**报错1：**



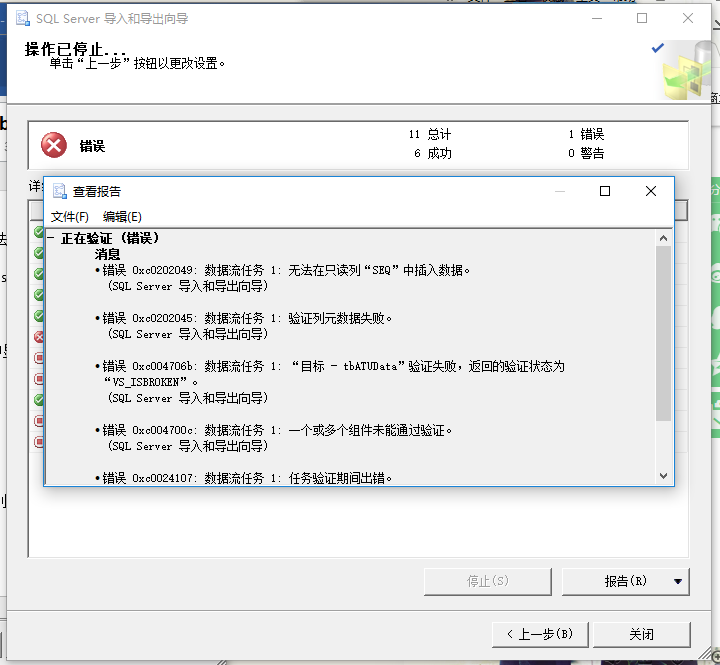
原因分析：  
 文件最后一行多了一行不完整的数据

解决方法：

删除最后一行数据即可

**6.2.4tbATUData**

**报错1：**

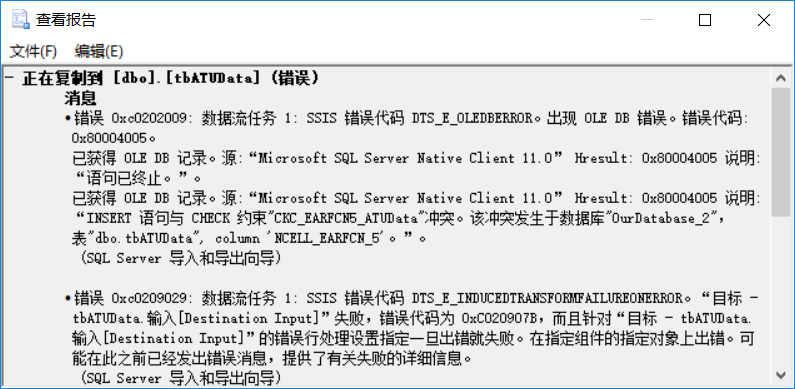


原因分析：

seq属性约束设为自增，为只读属性。不可写入

解决方法：

将seq的目标改为【忽略】，不导入这一列即可

**报错2：**

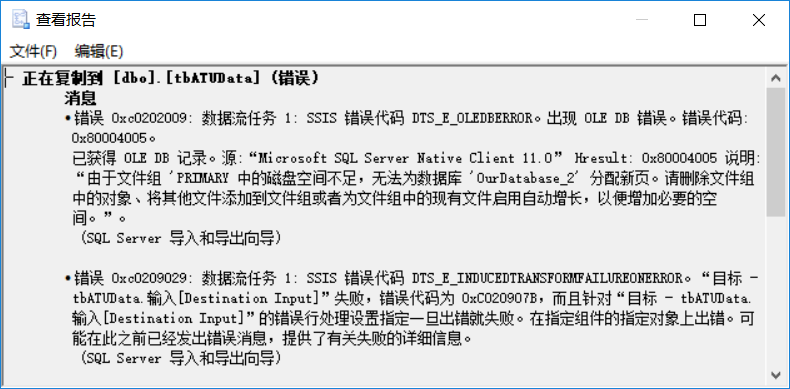
原因分析：

数据中'NCELL\_EARFCN\_5'列数值不仅限于所设置的约束值中

解决方法：

删除这个约束

**报错3：**



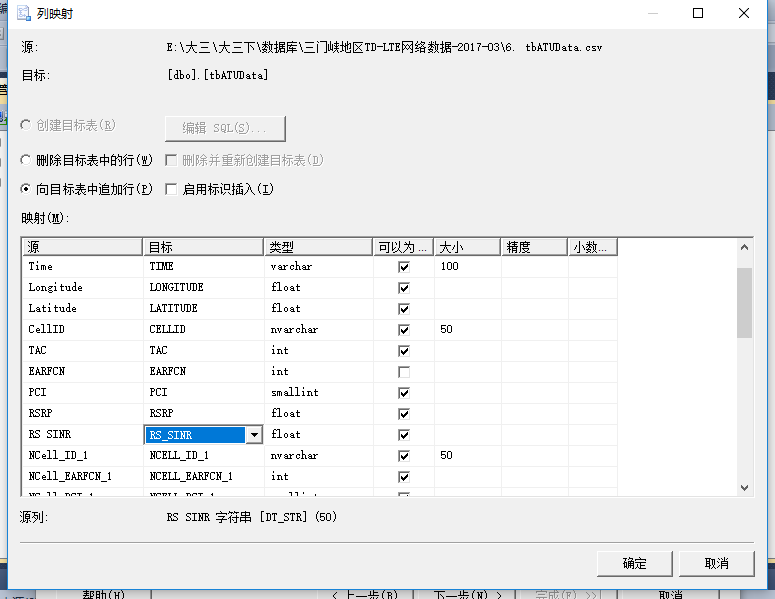
原因分析：

数据库初设大小不足

解决方法：

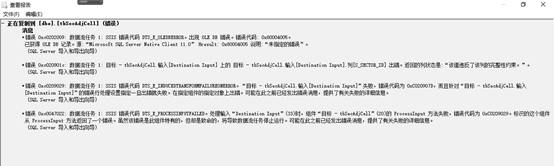
增大数据库大小即可

另外，在导入时，编辑映射选项中RS SINR中目标默认为忽略，发现目标名称与源名称不一致，需要手动选一下。



**6.2.5 SecAdjCell**

**报错1：**



原因分析：

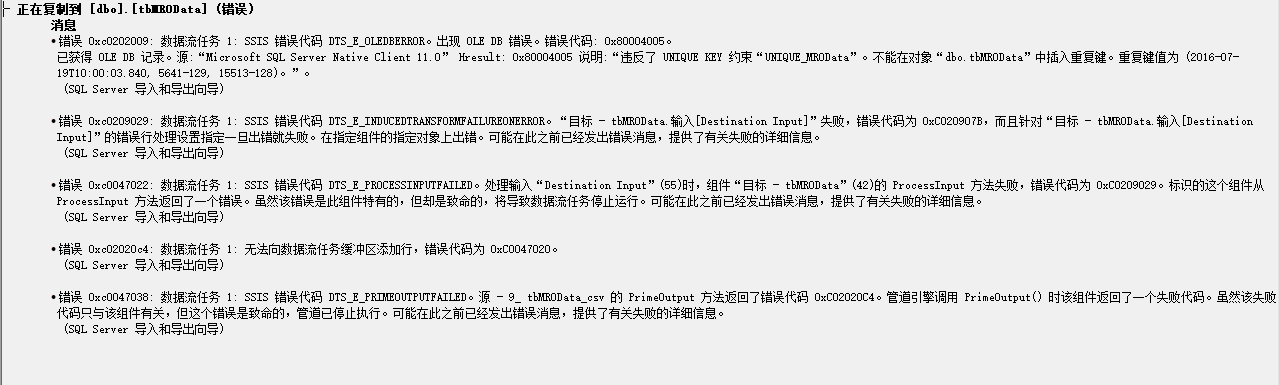
原数据表格中最后一行之后还多了一行空格

解决方法：

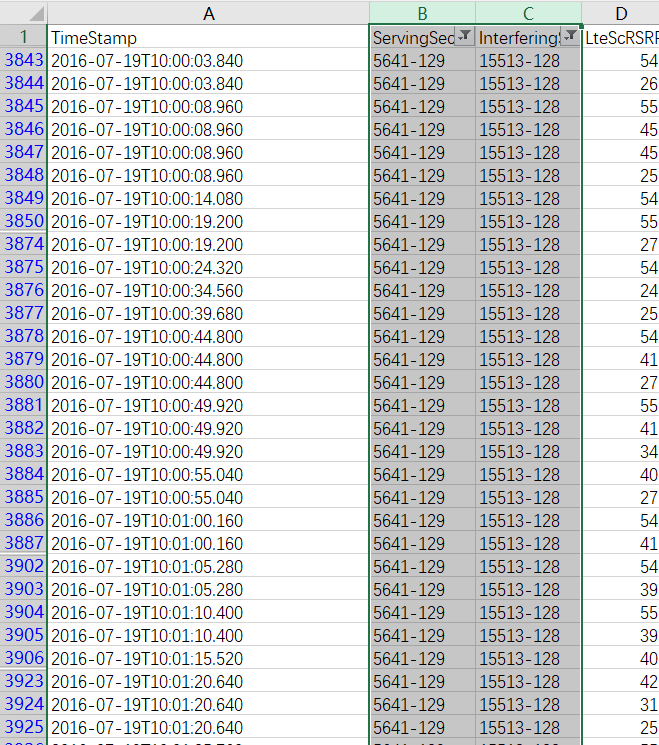
删掉那个空格即可

**6.2.6 tbMROData**

**报错1：**



原因分析：

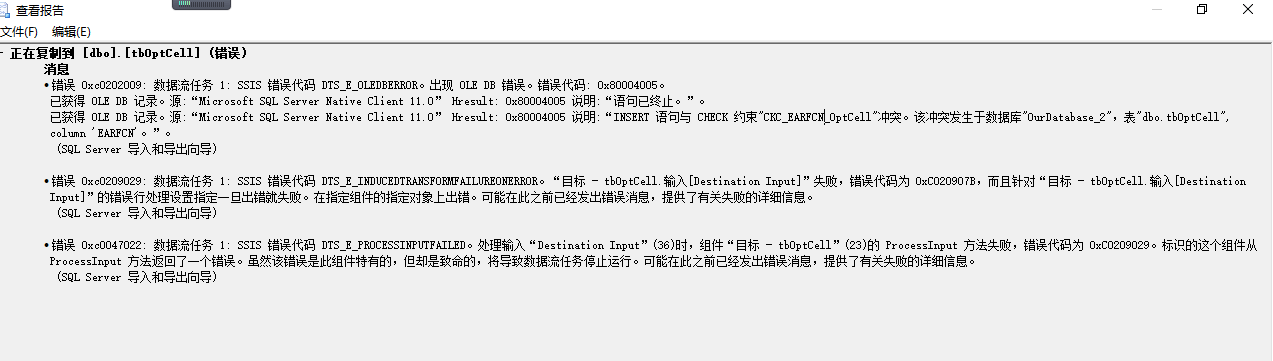
···· 数据文件中前三列主键中有很多重复的。

解决方法：

删掉重复的数据或在脚本中删掉主键约束，因为数据中重复的行过多，所以我们暂时选择删掉主键约束

**6.2.7 tbOptCell**

**报错1：**



原因分析：

是数据表格中有制定约束之外的值

解决方法：

EARFCN约束的值要加38496和38544,

### 实验总结

本次实验过程中我们遇到了很多问题，实验数据和数据库中设置的约束等不一致，造成导数据时出现各种报错。具体见上面的步骤，通过修改数据或是减少不必要的约束等方式解决了报错。在本次实验中，对于创建数据库以及表和视图，还有对于其内部属性添加约束的SQL语句有了进一步的了解。在解决问题的过程中，对于各项约束所起的作用，以及约束条件有了进一步的认识，比如主键不为空，或是不能重复等。这次实验提醒了我们数据库中的一些细节问题，因为不小心删除数据库或是重复添加表，出现了很多浪费时间的操作，所以每一步都要想好再做。

### 【脚本源代码】

if DB\_ID('OurDatabase\_2') is not null

USE master

go

drop database OurDatabase\_2

go

create database OurDatabase\_2

on primary

(

name='OurDatabase2',

filename='E:\data\LTE\OurDatabase2.mdf',

size=200MB,

maxsize=500MB,

filegrowth=20%

)

log on

(

name='MyDatabase2\_log',

filename='E:\data\LTE\OurDatabase2\_log.ldf',

size=50MB,

filegrowth=30MB

)

go

use OurDatabase\_2

create table tbCell(

CITY nvarchar(255) null,

SECTOR\_ID nvarchar(50) not null,

SECTOR\_NAME nvarchar(255) not null,

EARFCN int not null,

PCI int null,

PSS int null,

SSS int null,

TAC int null,

AZIMUTH float not null,

HEIGHT float null,

ELECTTILT float null,

MECHTILT float null,

TOTLETILT float null default 0,

ENODEBID int not null,

ENODEB\_NAME nvarchar(255) not null,

VENDOR nvarchar(255) null,

LONGITUDE float not null,

LATITUDE float not null,

STYLE nvarchar(255) null,

--默认聚集型索引，聚集性索引每个值只能出现一次

constraint PK\_SECTORID primary key clustered(SECTOR\_ID),--作为主键的变量不能为空值

constraint CKC\_EARFCN\_CELL check(EARFCN in (38350,38400,38098,38100,37900,37902,40936,40938,40940,38950,39052,39148,39250,38496,38544)),

constraint CKC\_PCI\_CELL check(PCI is null or (PCI between 0 and 503)),

constraint CKC\_PSS\_CELL check(PSS is null or (PSS in(0,1,2))),

constraint CKC\_SSS\_CELL check(SSS is null or (SSS between 0 and 167)),

constraint CKC\_VENDOR\_CELL check(VENDOR is null or (VENDOR in ('华为','中兴','诺西','爱立信','贝尔','大唐'))),

constraint CKC\_STYLE\_CELL check(STYLE is null or (STYLE in ('宏站','室内','室外','室分'))),

constraint CKC\_LONGITUDE\_CELL check(LONGITUDE between -180.00000 and 180.00000),

constraint CKC\_LATITUDE\_CELL check(LATITUDE between -90.0000 and 90.0000),

constraint CKC\_PCI2\_CELL check(PCI = 3\*SSS + PSS),

constraint CKC\_TOTLETILT\_CELL check(TOTLETILT = ELECTTILT + MECHTILT)

)

create table tbAdjCell(

S\_SECTOR\_ID nvarchar(50) not null,

N\_SECTOR\_ID nvarchar(50) not null,

S\_EARFCN int not null,

N\_EARFCN int not null,

constraint CKC\_SEARFCN\_AdjCell check(S\_EARFCN in (38350,38400,38098,38100,37900,37902,40936,40938,40940,38950,39052,39148,39250,38496,38544)),

constraint CKC\_NEARFCN\_AdjCell check(N\_EARFCN in (38350,38400,38098,38100,37900,37902,40936,40938,40940,38950,39052,39148,39250,38496,38544)),

primary key(S\_SECTOR\_ID,N\_SECTOR\_ID)

)

create table tbSecAdjCell(

S\_SECTOR\_ID varchar(50) not null,

N\_SECTOR\_ID varchar(50) not null,

primary key(S\_SECTOR\_ID,N\_SECTOR\_ID)

)

create table tbOptCell(

SECTOR\_ID nvarchar(50) not null,

EARFCN int not null,

CELL\_TYPE nvarchar(50) null,

primary key (SECTOR\_ID),

constraint UNIQUE\_OptCell unique(SECTOR\_ID),

constraint CKC\_EARFCN\_OptCell check(EARFCN in (38350,38400,38098,38100,37900,37902,40936,40938,40940,38950,39052,39148,39250,38496,38544)),

constraint CKC\_CELLTYPE\_OptCell check(CELL\_TYPE is null or (CELL\_TYPE in ('优化区','保护带')))

)

create table tbPCIAssignment(

ASSIGN\_ID smallint identity(1,1),

EARFCN int not null,

SECTOR\_ID nvarchar(50) not null,

SECTOR\_NAME nvarchar(200) null,

ENODEB\_ID int null,

PCI int null,

PSS int null,

SSS int null,

LONGITUDE float null,

LATITUDE float null,

STYLE varchar(50) null,

OPT\_DATETIME datetime null default getdate(),

primary key(ASSIGN\_ID,SECTOR\_ID),

constraint UNIQUE\_PCIAssignment unique(ASSIGN\_ID,SECTOR\_ID),

constraint CKC\_EARFCN\_PCIAssignment check(EARFCN in (38350,38400,38098,38100,37900,37902,40936,40938,40940,38950,39052,39148,39250,38496,38544)),

constraint CKC\_STYLE\_PCIAssignment check(STYLE is null or (STYLE in ('宏站','室内','室外'))),

constraint CKC\_PSS\_PCIAssignment check(PSS = PCI % 3),

constraint CKC\_SSS\_PCIAssignment check(SSS = PCI / 3),

)

create table tbATUData(

SEQ bigint identity(1,1),

FILENAME nvarchar(255) not null,

TIME varchar(100) null,

LONGITUDE float null,

LATITUDE float null,

CELLID nvarchar(50) null,

TAC int null,

EARFCN int not null,

PCI smallint null,

RSRP float null,

RS\_SINR float null,

NCELL\_ID\_1 nvarchar(50) null,

NCELL\_EARFCN\_1 int null,

NCELL\_PCI\_1 smallint null,

NCELL\_RSRP\_1 float null,

NCELL\_ID\_2 nvarchar(50) null,

NCELL\_EARFCN\_2 int null,

NCELL\_PCI\_2 smallint null,

NCELL\_RSRP\_2 float null,

NCELL\_ID\_3 nvarchar(50) null,

NCELL\_EARFCN\_3 int null,

NCELL\_PCI\_3 smallint null,

NCELL\_RSRP\_3 float null,

NCELL\_ID\_4 nvarchar(50) null,

NCELL\_EARFCN\_4 int null,

NCELL\_PCI\_4 smallint null,

NCELL\_RSRP\_4 float null,

NCELL\_ID\_5 nvarchar(50) null,

NCELL\_EARFCN\_5 int null,

NCELL\_PCI\_5 smallint null,

NCELL\_RSRP\_5 float null,

NCELL\_ID\_6 nvarchar(50) null,

NCELL\_EARFCN\_6 int null,

NCELL\_PCI\_6 smallint null,

NCELL\_RSRP\_6 float null,

constraint CKC\_EARFCN\_ATUData check(EARFCN in (38350,38400,38098,38100,37900,37902,40936,40938,40940,38950,39052,39148,39250)),

constraint UNIQUE\_ATUDate unique (seq,FileName)

)

create table tbATUC2I(

SECTOR\_ID nvarchar(50) not null,

NCELL\_ID nvarchar(50) not null,

RATIO\_ALL float null,

RANK int null,

COSITE tinyint null,

constraint CKC\_COSITE\_ATUC2I check (COSITE is null or (COSITE in (0,1))),

)

create table tbATUHandOver(

SSECTOR\_ID nvarchar(50) not null,

NSECTOR\_ID varchar(50) not null,

HOATT int null,

)

create table tbMROData(

TimeStamp nvarchar(30) not null,

ServingSector nvarchar(50) not null,

InterFeringSector nvarchar(50) not null,

LteScRSRP float null,

LteNcRSRP float null,

LteNcEarfcn int null,

LteNcPci smallint null,

constraint CKC\_EARFCN\_MROData check(LteNcEarfcn in (38350,38400,38098,38100,37900,37902,40936,40938,40940,38950,39052,39148,39250)),

)

create table tbC2I(

CITY nvarchar(255) null,

SCELL nvarchar(255) not null,

NCELL nvarchar(255) not null,

PrC2I9 float null,

C2I\_Mean float null,

Std float null,

SampleCount float null,

WeightedC2I float null,

primary key (SCELL,NCELL),

constraint CKC\_WeightedC2I\_C2I check(WeightedC2I = PrC2I9 \* SampleCount),

)

create table tbHandOver(

CITY nvarchar(255) null,

SCELL varchar(50) not null,

NCELL varchar(50) not null,

HOATT int null,

HOSUCC int null,

HOSUCCRATE numeric(7,4) null default null,

primary key (SCELL,NCELL)

)

go

create trigger pss\_pci on tbCELL for insert as

begin

declare

@pci int,

@numrows int

select @numrows = @@rowcount

if @numrows = 0

return

select @pci=PCI from inserted

update tbCELL set PSS=@pci%3 where PCI=@pci

return

/\* Errors handling \*/

end

go

create trigger pci\_sss on tbCELL for insert as

begin

declare

@pci int,

@numrows int

select @numrows = @@rowcount

if @numrows = 0

return

select @pci = PCI from inserted

update tbCELL set SSS = @pci/3 where PCI=@pci

return

/\* Errors handling \*/

end

go

create trigger HOSUCC on tbHandOver for insert as

begin

declare

@hoatt int,

@hosucc int,

@numrows int

select @numrows = @@rowcount

if @numrows = 0

return

select @hoatt=HOATT from inserted

select @hosucc=HOSUCC from inserted

if @hoatt is not null and @hosucc is not null and @hoatt>0

update tbHandOver set HOSUCCRATE=(@hosucc\*1.0000)/@hoatt where HOATT=@hoatt and HOSUCC=@hosucc

return

end

go

use OurDatabase\_2

go

create view tbcell38400 as

(

select dbo.[tbCell].SECTOR\_ID,dbo.[tbCell].ENODEBID,dbo.[tbCell].EARFCN,dbo.[tbCell].PCI,dbo.[tbCell].LATITUDE,dbo.[tbCell].LONGITUDE,dbo.[tbCell].STYLE

from dbo.[tbCell]

where dbo.[tbCell].EARFCN=38400

)

go

create view MROInfo as

(

select dbo.[tbMROData].ServingSector,dbo.[tbMROData].InterFeringSector,dbo.[tbMROData].LteScRSRP

from dbo.[tbMROData]

where dbo.[tbMROData].LteScRSRP>40

)

go

create view OptCellInfo as

(

select dbo.[tbOptCell].SECTOR\_ID,dbo.[tbOptCell].CELL\_TYPE,dbo.[tbC2I].NCELL as C2I\_NCELL,dbo.[tbC2I].C2I\_Mean,dbo.[tbHandOver].NCELL,dbo.[tbHandOver].HOSUCCRATE

from dbo.tbOptCell,dbo.tbC2I,dbo.tbHandOver

where dbo.tbOptCell.CELL\_TYPE='优化区'

)

go