**数据库表的简单介绍**

1. 数据库中与通道相关的表的表名组成介绍：

例如：需要查找7号机2单元3通道，3通道的通道信息如下图：



图一 工步编辑器显示的测试信息

在上图的的工步信息中，红色框选择中的[]是通道的MidNuid为 0B02001A，此Nuid是中位机的唯一标识； #7-2-3即为 7号机 2单元 3通道

上图的工步信息中，可以看到 测试编号为7213 最大数据序号为36441 也即记录条数为36441

还有一个单元nuid，即unit\_nuid为MidNuid （两两一组）高低位调换 （小端变大端）

根据上图，中位机的MidNuid为：0B02001A则 单元NUID为1a00020b

测试数据记录中需要使用到以下表：

1、测试信息表：存储测试编号，记录条数，测试工步文件，测试名称，DF数据存储分表信息（Split\_count），通道镜像(历史接续时使用)等信息；

测试信息表的组成格式为：bts902\_testinfo\_h\_单元号\_通道号，上图中的7号机 2单元3通道的测试信息表即为 即bts902\_testinfo\_h\_1a00020b\_3

2、测试信息内存表：存储测试编号，记录条数，DF数据存储分表信息等

测试信息内存表的组成格式为：bts902\_testinfo\_h\_memory\_单元号\_通道号，上图中的7号机 2单元3通道的测试信息内存表即为 即bts902\_testinfo\_h\_memory\_1a00020b\_3

3、测试辅助信息表（工步跳转信息表）：存储具体测试的工步开始；，工步结束，循环堆栈等信息；

测试辅助信息表的组成格式为：bts902\_data\_comprehensive\_单元号\_通道号，上图中的7号机 2单元3通道的测试辅助信息表即为 即bts902\_data\_comprehensive\_1a00020b\_3

4、DF数据表：存储具体测试的DF数据即Log数据

DF数据表的组成格式为：bts902\_data\_df\_blob \_单元号\_通道号**\_split\_count**，上图中的7号机 2单元3通道的即为 bts902\_data\_df\_blob\_1a00020b\_3\_split\_count split\_count为分表号，即当测试数据过多时会分到不同的数据表中存储；当split\_count为0时，默认不显示即

bts902\_data\_df\_blob\_1a00020b\_3 ； bts902\_data\_df\_blob\_1a00020b\_3\_1； bts902\_data\_df\_blob\_1a00020b\_3\_2 ；

对应关系：

一个通道有

一个 测试信息表 bts902\_testinfo\_h\_

一个 测试信息内存表 bts902\_testinfo\_h\_memory\_

一个 工步跳转表 bts902\_data\_comprehensive\_

多个 DF数据表 bts902\_data\_df\_blob \_单元号\_通道号\_split\_count

**其他表**

通道迁移表 表名 bts902\_chl\_move

存储通道迁移或者历史数据接续关系，当发生通道迁移或者继续时，需要凭借此关系进行数据合并，生成一份完整的测试数据；

1. 各表的详细介绍：

1、测试信息表(bts902\_testinfo\_h\_ xx) 的关键字段如下：

unit\_nuid (单元Nuid 十进制存储)

unit\_id (单元号 十进制存储)

chl\_phyid (通道号 十进制存储)

test\_id (测试编号 十进制存储)

test\_datasn (记录条数/最大数据序号 十进制存储)

end\_time (测试结束时间 十进制存储 存储格式为UInt64MicroSeconds)

batch (批次 )

battery\_model (型号 )

stepname (工步名称 )

bgn\_time (开始时间十进制存储 存储格式为UInt64MicroSeconds)

mid\_nuid (中位机MidNuid 十进制存储)

mid\_dev\_id (设备号/柜号 十进制存储)

chl\_idrow (行号 等同于单元号 十进制存储)

chl\_idcol (列号 等同于通道号 十进制存储)

step\_excutedata (工步文件 二进制存储)

image\_data (镜像文件 二进制存储)

split\_count (DF数据分表号 十进制存储)

data\_type (数据类型 十进制存储)

关于data\_type 一个测试在测试信息表中有2+条记录， 第一条data\_type = 0; 表示该条记录中包含了测试名称，工步信息，镜像信息等数据是主测试信息记录； 第二条到第N条 data\_type = 1，表示该条记录是存储DF数据分表号 split\_count的值不同；

2、测试信息内存表(bts902\_testinfo\_h\_memory \_ xx) 的关键字段如下：

unit\_nuid (单元Nuid 十进制存储)

unit\_id (单元号 十进制存储)

chl\_phyid (通道号 十进制存储)

test\_id (测试编号 十进制存储)

test\_datasn (记录条数/最大数据序号 十进制存储)

end\_time (测试结束时间 十进制存储 存储格式为UInt64MicroSeconds )

bgn\_time (开始时间 十进制存储 存储格式为UInt64MicroSeconds )

mid\_nuid (中位机MidNuid 十进制存储)

mid\_dev\_id (设备号/柜号 十进制存储)

chl\_idrow (行号 等同于单元号 十进制存储)

chl\_idcol (列号 等同于通道号 十进制存储)

split\_count (DF数据分表号 十进制存储)

data\_type (数据类型 十进制存储)

关于data\_type 一个测试在测试信息表中有2+条记录， 第一条data\_type = 0; 表示该条记录是主测试信息记录； 第二条到第N条 data\_type = 1，表示该条记录是存储DF数据分表号 split\_count的值不同；

3、工步跳转表(bts902\_data\_comprehensive\_ xx) 的关键字段如下：

mid\_nuid (中位机MidNuid 十进制存储)

chltest\_uid (测试编号 十进制存储 等同于测试信息表中的测试编号test\_id)

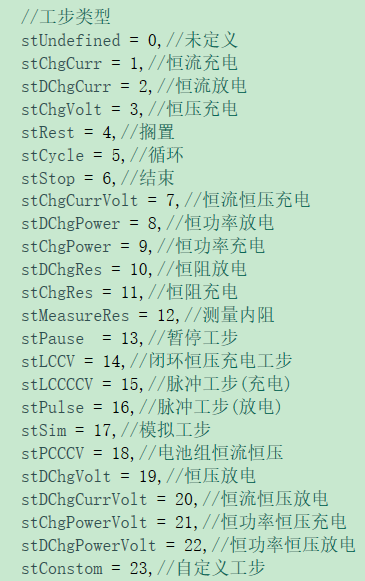
test\_datasn (数据序号 十进制存储)

step\_id (当前记录的工步号 十进制存储)

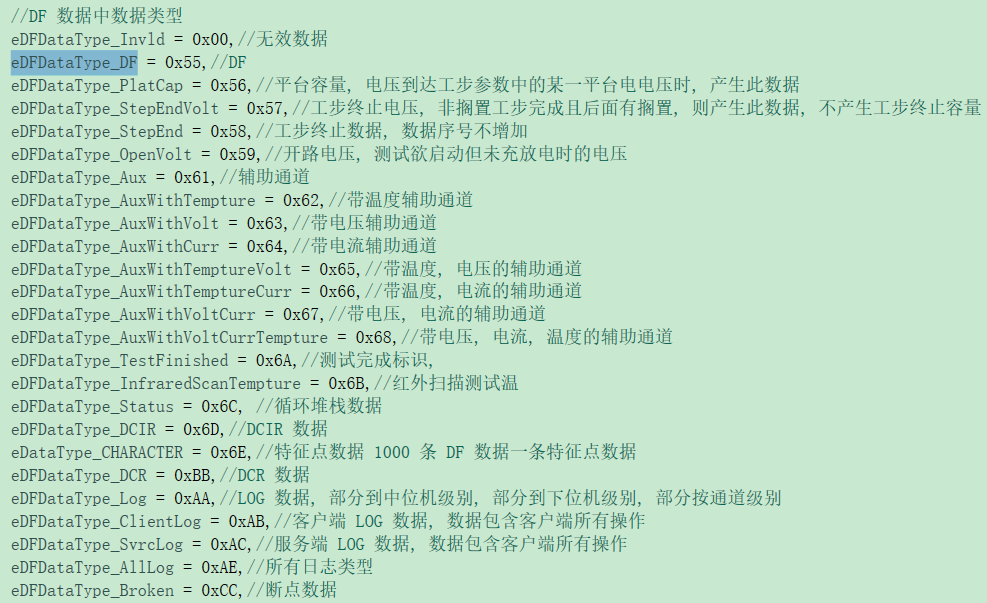
step\_type (当前记录的工步类型 十进制存储)

data\_type (当前记录的类型 十进制存储)

step\_type : 定义如下



data\_type : 定义如下



4、DF数据表 (bts902\_data\_df\_blob \_xx) 的关键字段如下：

chltest\_uid (测试编号 十进制存储 等同于测试信息表中的测试编号test\_id)

test\_datasn\_min (本包数据的最小数据序号 十进制存储)

test\_datasn\_max (本包数据的最大数据序号 十进制存储)

test\_len (本包数据中一条DF数据的长度 十进制存储)

test\_record\_count (本包数据中DF数据一共有多少条 十进制存储)

test\_data (这一包DF数据的具体内容 二进制存储)

5、通道迁移表 (bts902\_chl\_move ) 的关键字段如下：

src\_mid\_nuid (本次迁移的源测试mid\_nuid 十进制存储)

src\_unit\_nuid (本次迁移的源测试unit\_nuid 十进制存储)

src\_unit\_id (本次迁移的源测试单元号 unit\_id 十进制存储)

src\_chl\_phyid (本次迁移的源测试通道号 chl\_phyid 十进制存储)

src\_test\_id (本次迁移的源测试测试编号 test\_id 十进制存储)

src\_test\_datasn (本次迁移的源测试unit\_nuid 十进制存储)

desc\_mid\_nuid (本次迁移的目标测试mid\_nuid 十进制存储)

desc\_unit\_nuid (本次迁移的目标测试unit\_nuid 十进制存储)

desc\_unit\_id (本次迁移的目标测试单元号 unit\_id 十进制存储)

desc\_chl\_phyid (本次迁移的目标测试通道号 chl\_phyid 十进制存储)

desc\_test\_id (本次迁移的目标 测试编号 test\_id 十进制存储)

mid\_nuid (第一次启动的测试的mid\_nuid 十进制存储)

unit\_nuid (第一次启动的测试的unit\_nuid 十进制存储)

unit\_id (第一次启动的测试的单元号 unit\_id 十进制存储)

chl\_phyid (第一次启动的测试的通道号 chl\_phyid 十进制存储)

test\_id (第一次启动的测试的测试编号 test\_id 十进制存储)

**BTS9\_Manager\_Tool 维护工具使用指南**

1. Repair Testinfo 界面功能介绍

主要是测试信息相关的维护 对应数据库中测试信息表(bts902\_testinfo\_h\_xx)、测试信息内存表(bts902\_testinfo\_h\_memory \_ xx)表

1. 维护测试信息，不显示测试信息等

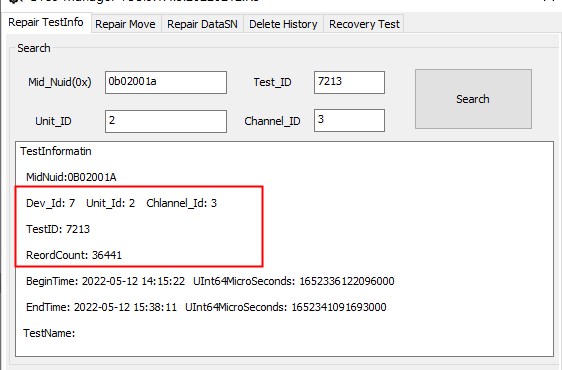
输入待查询的测试的 通道相关信息

Mid\_Nuid(0x) 即为单元MidNuid 可从 图一 工步编辑器显示的测试信息 中获取（方括号中的内容即为Mid\_Nuid）

Test\_ID 即为 测试编号对应的测试信息/内存表中为 test\_id 字段

Unit\_id即为 单元号对应的测试信息/内存表中为 unit\_id 字段

Channel\_id 即为 通道号 对应的测试信息/内存表中为 chl\_phyid 字段



查询结果即为上图框中显示

Dev\_ID 即为设备号

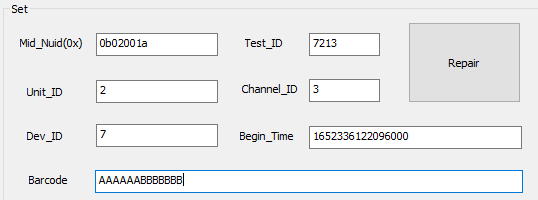
对应testInfo等测试信息表中为 mid\_dev\_id 字段

BeginTime 即为测试开始时间 UInt64MicroSeconds即为微秒时间

对应testInfo等测试信息/内存表中为 bgn\_time字段 均为UInt64MicroSeconds 转换的时间

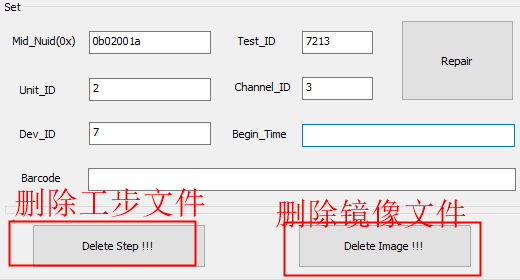
EndTime 即为测试结束时间/最后一次接收DF数据的时间 UInt64MicroSeconds为微秒时间

对应testInfo等测试信息/内存表中为 end\_time字段 均为UInt64MicroSeconds 转换的时间



当Dev\_ID, Unit\_id, Channel\_id, Bgn\_Time值异常时 可在设置界面填写正确的值,也可补录条码等 然后点击 Repair 即可

1. 删除测试信息，删除通道镜像等；

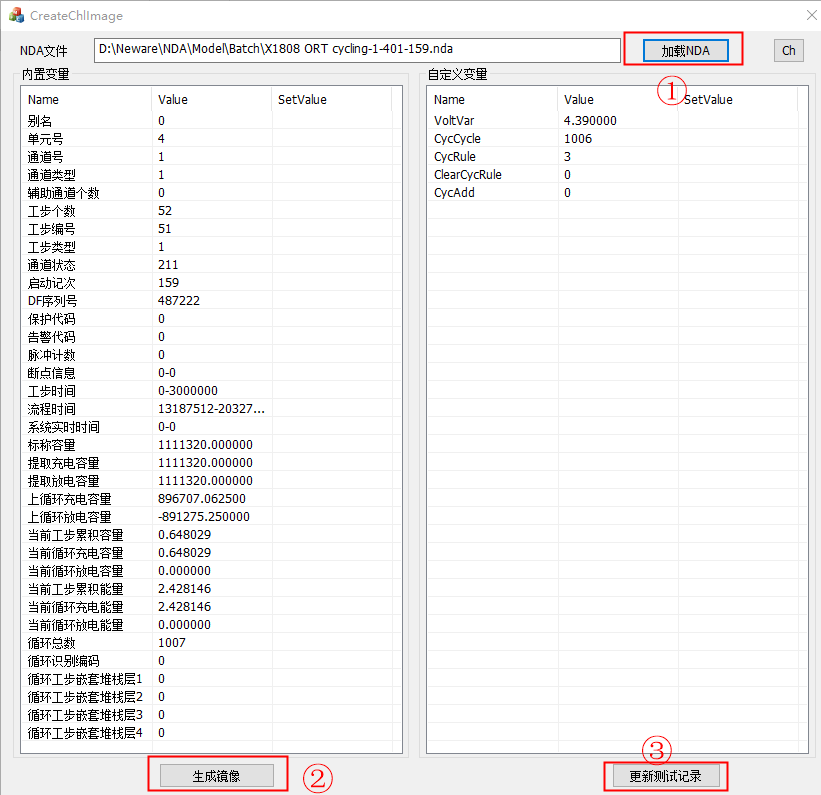


当打开数据报记录条数长度与工步信息中不匹配时，填写上述正确的通道信息以及测试编号后 (Begin\_Time, Barcode等字段不需要填写) ，点击Delete Step 按钮即可

然后在通道历史数据界面上选择对应的测试 右键 插入测试信息 选择对应正确的工步文件，点击确定/启动即可

当需要重新生成镜像时，填写上述正确的通道信息以及测试编号后 (Begin\_Time, Barcode等字段不需要填写) ，点击Delete Image 按钮即可

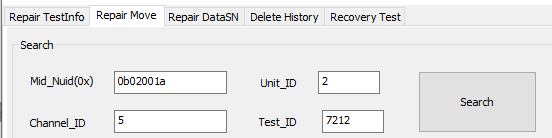
然后使用镜像工具加载NDA 生成正确的镜像即可



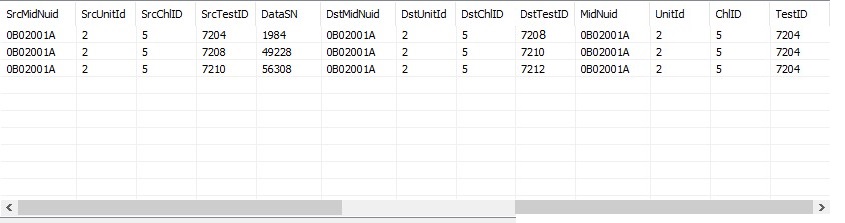
1. Repair Move 界面功能介绍

主要是历史数据接续/通道迁移相关的维护 对应数据库中通道迁移表(bts902\_chl\_move)

1. 查询通道历史接续/迁移关系 输入对应的通道信息以及测试ID 点击查询



结果如下：



上述信息描述了与要查询的测试相关联的历史接续/迁移记录

Src测试通道信息 测试ID 记录条数 Dst测试通道信息+测试ID 首次测试通道信息 测试ID

如上图所示：

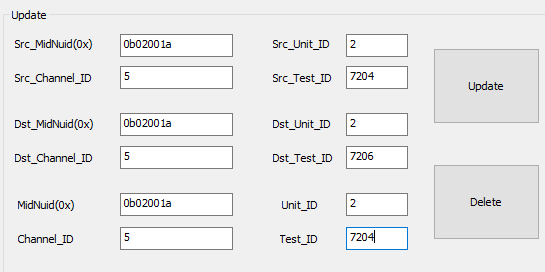
首次测试是 MidNuid(0b02001a) Unit\_id(2)的5通道的7204号测试；

第一次迁移/接续 是从2-5通道 (7204测试)到 2-5通道(7208测试)

第二次迁移/接续 是从2-5通道 (7208测试)到 2-5通道(7210测试)

第三次迁移/接续 是从2-5通道 (7210测试)到 2-5通道(7212测试)

1. 修改测试迁移或者历史接续关系



修改源测试信息或者目标测试信息，首次测试信息不可以修改，需填入查询到的首次测试启动信息，然后点击Update 按钮即可

若需要删除某个记录，则需要填入对应的源测试信息，目标测试信息，首次测试启动信息，然后点击 Delete 按钮即可

1. Repair DataSN界面功能介绍

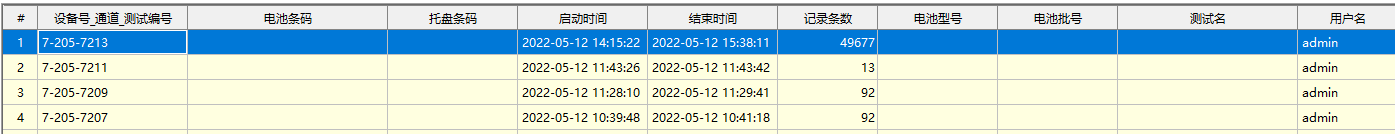
主要是删除某一段测试DF数据，重新进行历史数据接续；(例如测试异常导致完成，删除异常的那一段数据，重新进行历史数据接续)

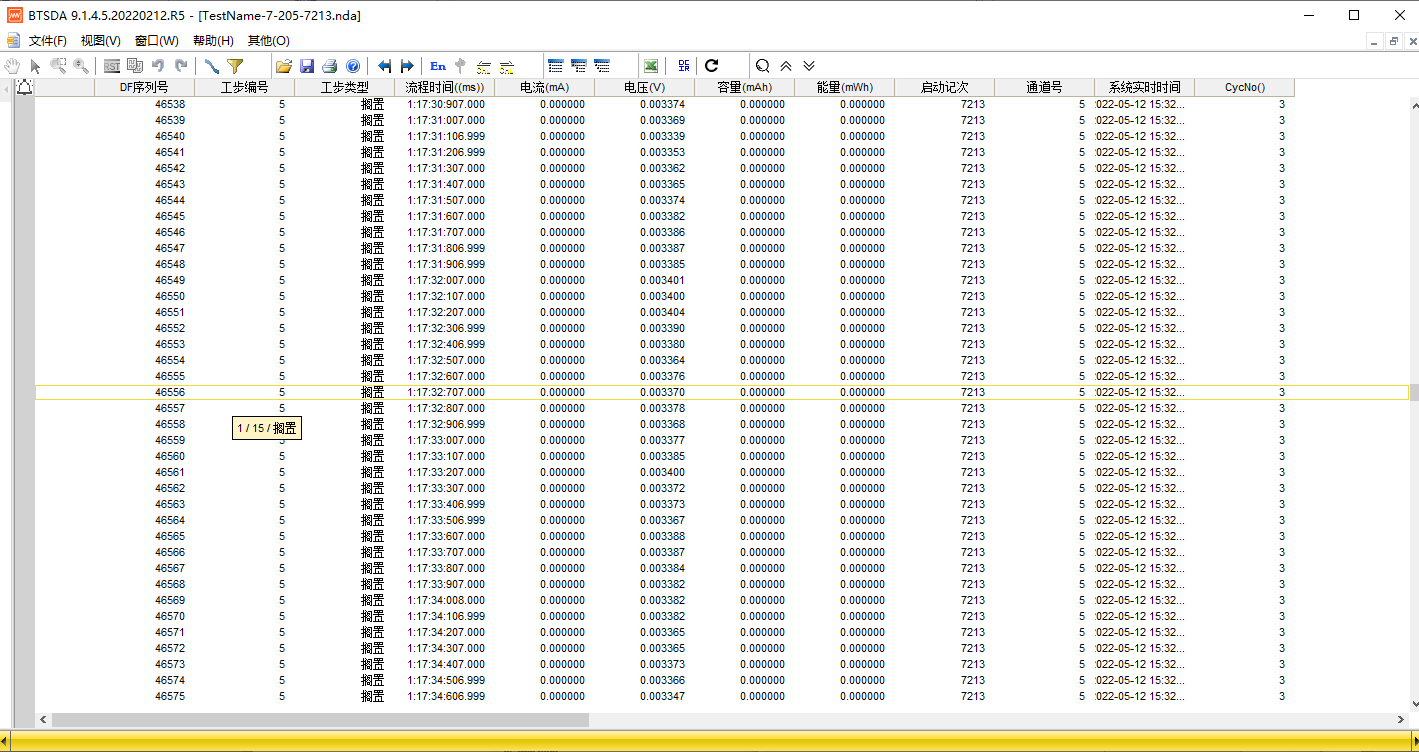
对应操作如下：

1. 根据测试NDA数据，找到大概需要的删除的数据范围；
2. 查询DF数据表，根据DF数据包确定真实的需要删除的某一段数据并进行删除，然后记住删除后的数据序号 test\_datasn\_max；
3. 找到需要删除的工步跳转信息表，删除跳转信息表中属于已经删除的DF数据中的工步跳转记录；
4. 修改测试信息表，测试信息内存表中的数据记录test\_datansn；
5. 重新下载NDA文件；生成镜像，进行历史数据接续；

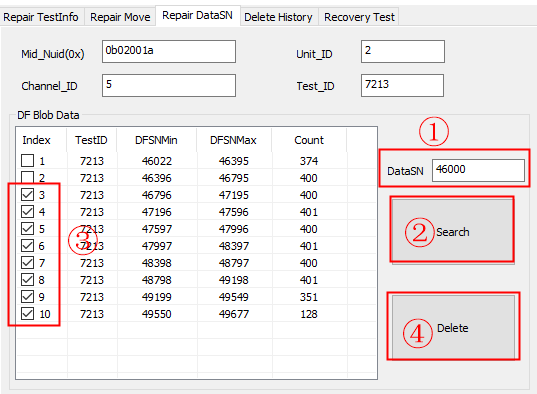
下面以 205通道7213号测试为例，删除大概46556之后的数据；

步骤一、下载NDA文件，在NDA文件中找到大概需要删除的数据块

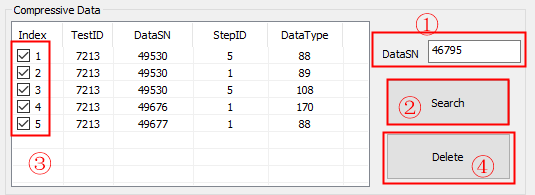




步骤二、在维护工具中 DF Blob Data 数据块区域，输入要检索的数据块本次选择46000（这个值根据需要可以比需要删除的数据要小一些 (小1000以内)），检索的结果是46000所在的那一包数据之后的数据包；勾选需要删除的数据块；根据实际需要确定删除的数据包，我们这里选择删除46795这包之后的数据，然后点击 Delete 按钮即可，删除完毕后记住最后一包最大的数据序号DFSNMax 为 46795；



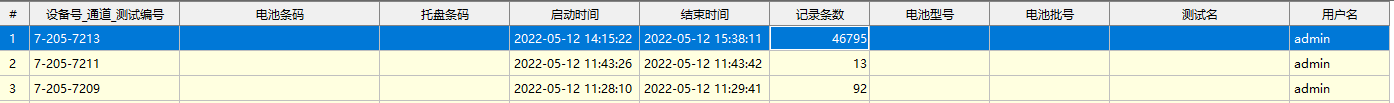
步骤三、在维护工具中 Compressive Data 数据块区域，输入步骤二中记录的最大数据序号46795 （即原来的46795的数据都是不再需要的），勾选这些数据并删除；

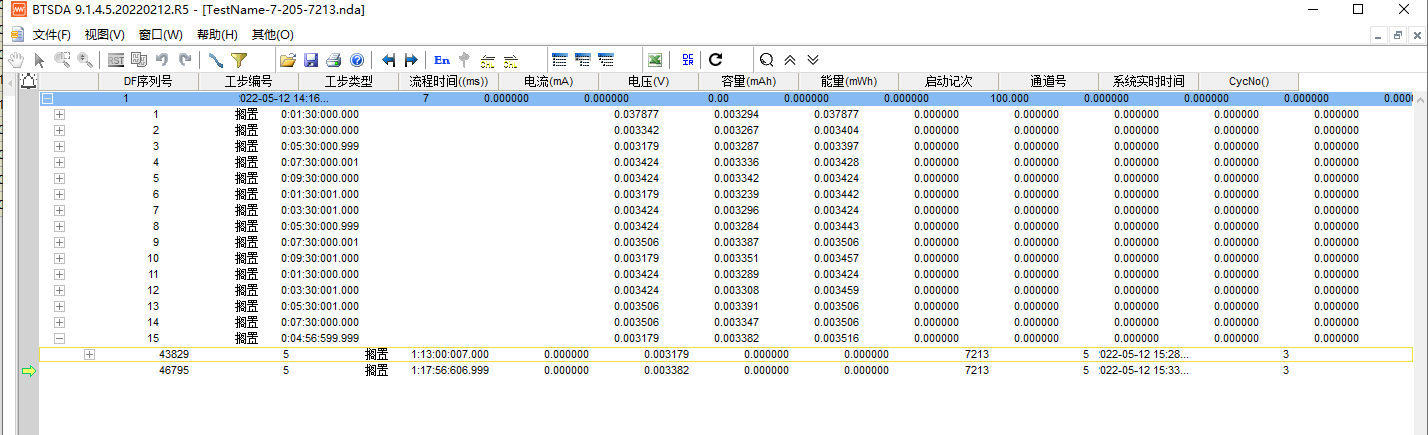


步骤四、在维护工具中DataSN区域中，输入步骤二中记录的最大数据序号46795 （即原来的46795的数据都是不再需要的），然后点击Update 按钮；



步骤五、删除之前下载的NDA文件，在客户端重新刷新历史数据，记录条数已经变更为46795，然后重新下载NDA文件，即可看到数据中最后只有46795条了；然后再重新生成镜像，进行历史数据接续即可；





1. RepairDB界面功能介绍

主要是维护数据库表异常的情况；

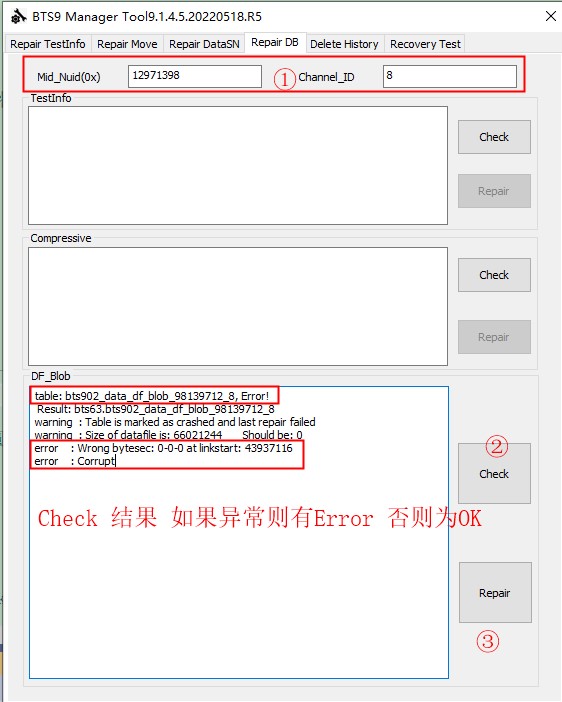
涉及到数据库表有测试信息表，DF数据表，工步跳转表等

当下载数据异常时，可尝试检查DF\_BLOB表，当测试信息异常时可检查TestInfo；

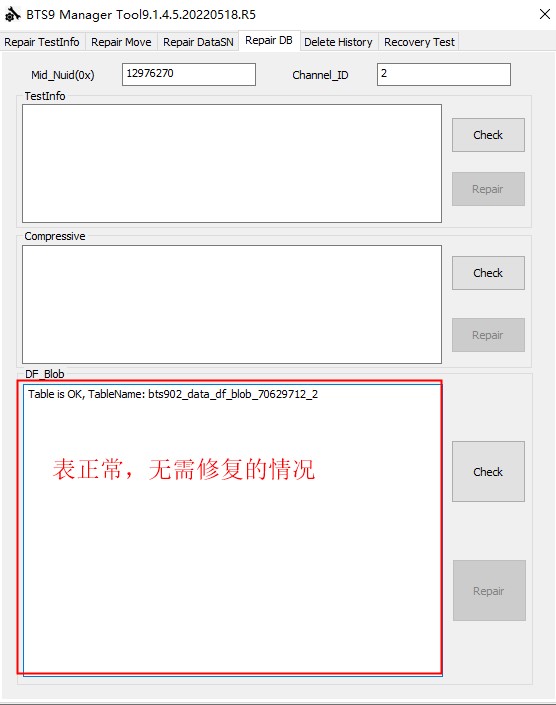
默认 Repair按钮为禁用状态 当检查出错误时，Repair恢复正常可操作状态;

注意： Check或者Repair前 当前操作通道需要停止五分钟以上！！！！！！！

下图为需要修复的情况



下图为正常无需修复的情况，Check 完毕 显示正常 Repair依然为灰色禁用状态

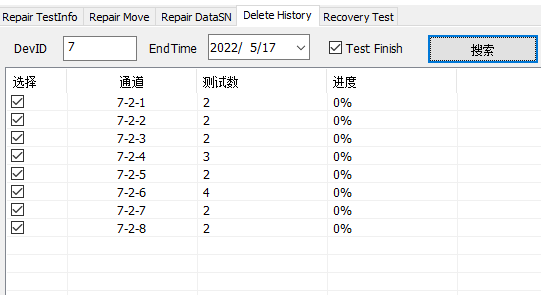


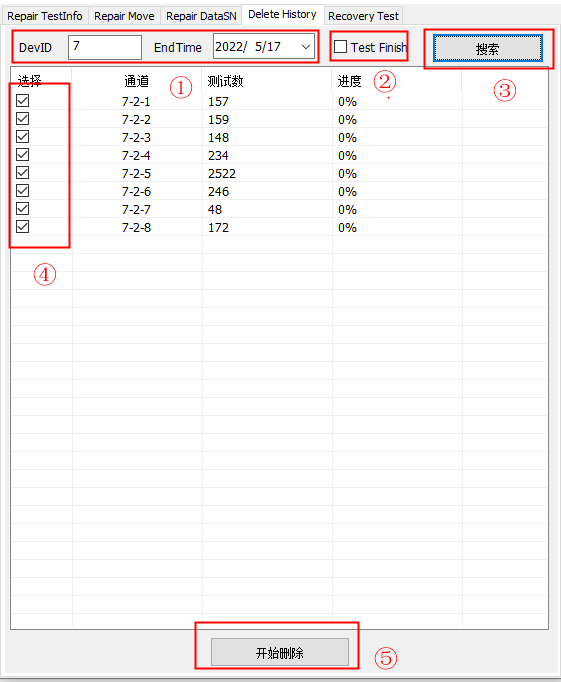
1. Delete History界面功能介绍

主要是删除历史的测试；

涉及到数据库表有测试信息表，DF数据表，工步跳转表，通道迁移表等；

下述界面中的Test Finish 可选框 选中时表示只删除已经完成的测试，不选中时则表示删除在这个时间之前的所有测试；





1. Recovery Test界面功能介绍

主要是将NDA文件恢复至数据库中；

涉及到数据库表有测试信息表，DF数据表，工步跳转表，通道迁移表等；

