Prepared (also subject responsible if other)		No.		
Brendan Treacy				
Approved	Checked	Date	Rev	Reference
		2008-05-26	PA1	

Test Instruction for CS3.0 CP3 Minsat 5.2.1 ICP07

Prepared (also subject responsible if other)		No.		
Brendan Treacy				
Approved	Checked	Date	Rev	Reference
		2008-05-26	PA1	

Contents

Prepared (also subject responsible if other)		No.			
Brendan Treacy					
Approved	Checked	Date	Rev	Reference	
		2008-05-26	PA1		

1 Introduction

1.1 General

This document specifies the instructions regarding how to verify the corrected TRs in CS3.0 CP3 Minsat 5.2.1 ICP07. It also documents the result of the tests.

2 Test instructions

2.1 HI60971 FailedRechargeDetail table grows indefinitely

Also includes TRs HJ13556 and HI77256.

2.1.1 Description of fault

Functionality to archive FailedRechargeDetail table has been added to Stats Archive Procedure – starc. Additionally command options have been added to starc to allow control of size of batches of RechargeDetail, FailedRechargeDetail

2.1.2 Test environment set up

1. Run DRGEN to create 50,000 Refill Records and 50,000 Voucher Error records. The date on the these records should span 125 days to mimic a real situation where data has accumulated beyond 90 days and the details are as followsThe terms and abbreviations used in this document are explained below.

RechargeDetail

No of rows	AccountID Range	Date	Days
?	42 – 1000	19th March 2008	1
10104	1001 – 2000	18th Feb 2008	30
10824	2001 – 3000	17th Jan 2008	62
10635	3001 – 4000	20th Dec 2007	90
10690	4001 – 5000	15th Nov 2007	125

FailedRecharge Data

No of rows	AccountID Range	Date	Days
?	1 – 1000	19th March 2008	1
11686	1001 – 2000	18th Feb 2008	30
11661	2001 – 3000	17th Jan 2008	62
11760	3001 – 4000	20th Dec 2007	90

Prepared (also subject responsible if other)		No.			
Brendan Treacy					
Approved	Checked	Date	Rev	Reference	
		2008-05-26	PA1		

11505	4001 – 5000	15th Nov 2007	125

In the conditions below **numdays** is set by calculating the number of days from each of the above dates + 1

So for 19th March 2008, select datediff(day, "19 Mar 2008", "01 Apr 2008") = 13 + 1 = 13 which is the value which is used as a paramenter in the conditions below instead of the Actual No of Days in table above. This allows the test conditions to be run at any future date and still maintain the days difference above.

2. After Process DR has been run and stats BCP Files produced, Stats DR Records are processed and loaded into the tables RechargeDetail and FailedRechargeDetail.
3. Copy the contents of these tables to temporary ones to enable them to be restored as specified for each of the tests below
4. Created Test scripts that follows:

```
/* Step 10 - Create RestoreFailedRechargeDetail * RestoreRechargeDetail */

create table RestoreRechargeDetail (
  SubscriberID   numeric(18,0)  not null,
  RechargeDate   datetime  not null,
  CurrencyTypeCode   tinyint  not null,
  UserName   varchar(20)  null,
  RechargeType   varchar(3)  not null,
  RechargeSubType   varchar(3)  null,
  RechargeSource   varchar(2)  not null,
  BalanceAfterRefill   float  not null,
  Amount   float  not null,
  VoucherSerialNumber   varchar(20)  null,
  DeactivationDate   datetime  null
)
create table RestoreFailedRechargeDetail(
  SubscriberID   numeric(18,0)  not null,
  RechargeDate   datetime  not null,
  RefillTypeCode   int  not null,
  RefillErrorCode   int  not null,
  CurrencyTypeCode   tinyint  not null,
  UserName   varchar(20)  null,
  VoucherSerialNumber   varchar(20)  null,
  Amount   numeric(18,9)  null )

/* Step 20 - Age data so that it can be selected for archiving */
```

```
    update RestoreRechargeDetail
    Set RechargeDate = '19 Mar 2008'
    where SubscriberID > 1
     and SubscriberID < 1001

    update RestoreFailedRechargeDetail
```

Prepared (also subject responsible if other)		No.		
Brendan Treacy				
Approved	Checked	Date	Rev	Reference
		2008-05-26	PA1	

```
       Set RechargeDate = '15 Nov 2007'
       where SubscriberID > 4000
        and SubscriberID < 5001
```

/* Step 30 - Copy data from FailedRechargeDetail & RechargeDetail into temporary tables to enable them to be restored quickly after test */
```
       insert into RestoreRechargeDetail
       select  * from RechargeDetail

       select * from RestoreRechargeDetail

       insert into RestoreFailedRechargeDetail
       select  * from FailedRechargeDetail
```

/* Step 40 - Run procedure in Unix Window as specified in Test Condidtions Document */

/* Step 50 - Check results by selecting data from Archive tables */

```
       select count(*) from RechargeDetail

       select * from RechargeDetail
       where RechargeDate > '18 Mar 2008'
             order by SubscriberID

       select count(*) from ArchiveRechargeDetail

       select count(*) from ArchiveRechargeDetail
       order by SubscriberID
       where RechargeDate > '18 Mar 2008'

       select count(*) from RestoreRechargeDetail
       where RechargeDate > '18 Mar 2008'

       /* Failed Recharge */

       select count(*) from FailedRechargeDetail
       where RechargeDate > '18 Mar 2008'
       order by SubscriberID

       select * from FailedRechargeDetail
       order by SubscriberID


       select count(*) from ArchiveFailedRechargeDetail
       where RechargeDate > '18 Mar 2008'
       order by SubscriberID
```

/* Step 60 - Save test results */

/* Step 70 - Restore test data from FailedRechargeDetail & RechargeDetail into

Prepared (also subject responsible if other)		No.		
Brendan Treacy				
Approved	Checked	Date	Rev	Reference
		2008-05-26	PA1	

```
main tables to enable tests to be rerun */
    delete RechargeDetail
    go

    insert into RechargeDetail
    select  * from RestoreRechargeDetail
    go

    delete ArchiveRechargeDetail
    go

    /* Restore Part 2 */
    delete FailedRechargeDetail
    go

    insert into FailedRechargeDetail
    select  * from RestoreFailedRechargeDetail
    go

    delete ArchiveFailedRechargeDetail
    go
```

starc usage and description of main parameters referenced in Test Conditions below
usage: [-vAgh] [-U dbusr] [-P dbpwd] [-S dbsrv] [-D dbnam] [-n numdays]\n[-a arcdir] [-b batchsize] [-r rechargebatch] [-f failedrbatch] [-s sscbatch]

-A do the archive to disk
-n numdays	retain number of days of transactions
-w numwrites	number of times per year to unload archive table to disk
-b batchsize	max number of recrords archived in each batch
-r rechargebatch	max number of RechargeDetail batches
-f failedrbatch	max number of failedrecharge batches
-f failedrbatch	max number of failedrecharge batches

Prepared (also subject responsible if other)		No.			
Brendan Treacy					
Approved	Checked	Date	Rev	Reference	
		2008-05-26	PA1		

2.1.3   Test cases

No	Test condition	Test data	Expected result	Actual result
1.	Test when starc is run with **default parameters** that data > 90 days is archived from RechargeDetail but no data is archived from FailedRechargeDetail	Create 125 days of test data as specified above :- Refills = 50,000 Voucher Error =50,000	Data on RechargeDetail table older than 90 days will be archived. No data will be archived from FailedRechargeDetail.	As expected data from RechargeDetail table > 90 days was archived iE.e 21325 rows archived No data was archived from FailedRechargeD etail
2.	Test when starc is run with **numdays set to 180, -b batchsize = 5000, –r rechargebatch set to 10 and -f failedrbatch set to 10** that no rows are archived from the RechargeDetail, FailedRechargeDetail.	As setup in test 1 above.	No data archived from any of the tables as none of the data is older than 180 days.	As expected
3a.	Test when starc is run with **–n numdays set to 2** that all data > 2 day old is archived from RechargeDetail	9958 rows with Recharge Date = 19th March 2008 i.e 1 day old	All data > 2 days old will be archived leaving 1 day old test data	As expected
3b.	Test when starc is run with **–n numdays set to 2, -b batchsize set to 5000, –r rechargebatch set to 8** that only data 1 day old is retained on RechargeDetail	9958 rows with Recharge Date = 19th March 2008 i.e 1 day old	40,000 rows will be archived from RechargeDetail table leaving some day > 2 days old as batchzize * rechargebatch limit in place.	As expected
3c.	Test when starc is run with **–n numdays set to 2, -f failedrbatch set to 10** that all data > 2 day old is archived from FailedRechargeDetail as archive is not limited by small batchsize as default is 200,000.	9871 rows with Recharge Date = 19th March 2008 i.e 1 day old	All data > 2 days old will be archived leaving 1 day old test data	As expected all 9871 rows dated 19th March were left on the FailedRechargeD etail while 46666 rows were archived to Archive table.
3d.	Test when starc is run with **–n numdays set to 2, -b batchsize set to 5000, -f failedrbatch set to 8** that only data 1 day old is retained on	9871 rows with Recharge Date = 19th March 2008 i.e 1 day old	40,000 rows will be archived from FailedRechargeDetail table leaving some data still > 1 day old.	As expected 16537 rows were left on the FailedRechargeD etail which includes the 1 day

Prepared (also subject responsible if other)		No.			
Brendan Treacy					
Approved	Checked	Date 2008-05-26	Rev PA1	Reference	

	FailedRechargeDetail.			old data + other data > 1 day old
4a.	Test when starc is run with **–A parameter i.e Archive to directory and all others parameters set to default**, that data is archived correctly from RechargeDetail into Archive table and then into file in default DBArchive Directory and no data is archived from FailedRechargeDetail	Restore all test data as for test 1.	Refill data older than 90 days will be archived from RechargeDetail.

As rechargebatch parameter set to 16 * 200,000 by default it will have no affect on limiting the rows archived in this case.

Zero rows will be archived from SubscriberStatusChange and FailedRechargeDetail. | Data was archived correctly to RechargeDetail table but no file was created in /export/home/min sat/DBArchive TR HJ13556 |
| 4b | Rerun 4a above to see if data is archived to DBArchive directory with .ERR extension | As after test 4a | Data from ArchiveRechargeDetail will be archived to .ERR file in DBArchive | Data from ArchiveRecharge Detail was archived to file :- ArchiveRecharge Detail.ERR. 200803211234 |
| 5. | Test when starc is run with -A parameter, **numdays = 90 days (default) and –r rechargebatch parameter set to 5** that the correct no of rows are archived from RechargeDetail and no rows are archived from FailedRechargeDetail. | Restore data as set-up for Test 1 above. | 1,000,000 oldest Refill rows will be archived i.e (default batchsize of 200,000 * 5 are archived leaving some data older than 90 days.

Zero rows will be archived from SubscriberStatusChange and FailedRechargeDetail. | Not run |
| 6. | Test when starc is run with -A parameter, **numdays default = 30 days, –r rechargebatch set to 0, –f failedrbatch set to 0 (default)** that the correct no of rows are archived from RechargeDetail and no rows are archived from FailedRechargeDetail | Restore data as set-up for Test 1 above. | Data older than 30 days will be archived from RechargeDetail

Zero rows will be archived from FailedRechargeDetail. | As expected. |
| 7. | Test when starc is run with −**A, -n numdays = 30, –r rechargebatch set to 10** that the correct no of rows are archived only from the | Unchanged from previous test 6. | Data older than 30 days will be archived from FailedRechargeDetail table. As failedrbatch set to 10 | As expected. |

Prepared (also subject responsible if other)		No.			
Brendan Treacy					
Approved	Checked	Date	Rev	Reference	
		2008-05-26	PA1		

		FailedRechargeDetail table.		* 200,000, it will have no affect on limiting the rows archived. Zero batches will be archived from RechargeDetail.	
8.		Test when starc is run with **–A parameter i.e Do Archive, numdays = 2 , -b batchsize set to 3,000, –r rechargebatch set to 0, -f failedrbatch set to 10** that the correct no of rows are archived only from the FailedRechargeDetail table.	Restore data as set-up for Test 1 above.	30,000 oldest rows on FailedRechargeDetail will be archived leaving some data possibly older than 2 days.	As expected.
9.		Test when starc is run with **numdays = 3**, **batchsize set to 1,000, –r rechargebatch parameter set to 50** that the correct no. of rows are archived from RechargeDetail and no rows are archived from FailedRechargeDetail	Restore data as set-up for Test 1 above.	50,000 oldest rows on RechargeDetail will be archived leaving some data still older than 3 days All data older than 90 days will be archived from FailedRechargeDetail table.	As expected.
10.		Test when starc is run with numdays = 3, **batchsize set to 50,000**, **–r rechargebatch parameter set to 10, -f set to 10** that the correct amount of rows from RechargeDetail and FailedRechargeDetail	Restore data as set-up for Test 1 above.	Same result as for Test 9 above.	As expected
Run cumulative tests followed by archive to DBArchive directory with –A parameter to simulate real live environment situation where data will be archived to tables regularly and then archived with –A parameter to output data to file System Date: 1st April 2008					
11a.		Test when starc is run with **numdays default = 138 days,** that the correct no of rows are archived from RechargeDetail	Restore data as set-up for Test 1 above.	Data older than numdays will be archived from RechargeDetail Zero rows will be archived from FailedRechargeDetail.	As expected.
11b.		Test when starc is run with **numdays default = 103 days,** that the correct no of rows are archived	As after test 11a	Data older than numdays will be archived from RechargeDetail Zero rows will be archived	As expected.

Prepared (also subject responsible if other)		No.			
Brendan Treacy					
Approved	Checked	Date	Rev	Reference	
		2008-05-26	PA1		

		from RechargeDetail		from FailedRechargeDetail.	
11c.	Test when starc is run with **numdays default = 75 days,** that the correct no of rows are archived from RechargeDetail	As after test 11b.	Data older than numdays will be archived from RechargeDetail Zero rows will be archived from FailedRechargeDetail.	As expected.
11d.	Test when starc is run with **numdays default = 43 days,** that the correct no of rows are archived from RechargeDetail	As after test 11c.	Data older than numdays will be archived from RechargeDetail Zero rows will be archived from FailedRechargeDetail.	As expected.
11e.	Test when starc is run with **–A parameters and numdays default = 13 days** that the correct no of rows are archived from RechargeDetail	As after test 11d	Data older than numdays will be archived from RechargeDetail Zero rows will be archived from FailedRechargeDetail.	As expected.
12a.	With System Date set to 1/4/2008 Test when starc is run with **numdays default = 137 days, , –r rechargebatch set to 0, -f failedrbatch set to 10. –w numwrites set to 24** that the correct no of rows are archived from FailedRechargeDetail	Restore data as set-up for Test 1 above.	Data older than numdays will be archived from FailedRechargeDetail and file will be created in DBArchive directory. Zero rows will be archived from RechargeDetail.	As expected.
12b.	With System Date set to 8/4/2008Test when starc is run with **numdays default = 102 days, , –r rechargebatch set to 0, -f failedrbatch set to 10, –w numwrites set to 48** that the correct no of rows are archived from FailedRechargeDetail	As after test 12a	Data older than numdays will be archived from FailedRechargeDetail and file will be created in DBArchive directory. Zero rows will be archived from RechargeDetail.	As expected.
12c.	With System Date set to 8/4/2008Test when starc is run with **numdays default = 74 days, , –r rechargebatch set to 0, -f failedrbatch set to 10**, **–w numwrites**	As after test 12b.	Data older than numdays will be archived from FailedRechargeDetail but data on ArchiveFailedRechargeDetail will not be written to a file as System Date is set to 8 of month whereas numwrites is	As expected.

Prepared (also subject responsible if other)		No.			
Brendan Treacy					
Approved	Checked	Date 2008-05-26	Rev PA1	Reference	

	set to 24 that the correct no of rows are archived from FailedRechargeDetail		set to 24. Zero rows will be archived from RechargeDetail.		
12d.	With System Date set to 8/4/2008Test Test when starc is run with **numdays default = 42 days, , –r rechargebatch set to 0, -f failedrbatch set to 10**, **-–w numwrites set to 24** that the correct no of rows are archived from FailedRechargeDetail	As after test 12c.	Data older than numdays will be archived from FailedRechargeDetail but data on ArchiveFailedRechargeDetail will not be written to a file as System Date is set to 8 of month whereas numwrites is set to 24. Zero rows will be archived from RechargeDetail.	As expected.
12e.	With System Date set to 16/4/2008 Test when starc is run with **–A parameters and numdays default = 12 days, –r rechargebatch set to 0, -f failedrbatch set to 10, -–w numwrites set to 24** that the correct no of rows are archived from FailedRechargeDetail	As after test 12d	Data older than numdays will be archived from FailedRechargeDetail and file will be created in DBArchive with data for 17th Jan, 18th Feb and 18th March. Zero rows will be archived from RechargeDetail.	As expected.
13.	TR HJ13556 - starc writes archived data to disk in .ERR files	Restore data as set-up for Test 1 above and carry out initial runs to archive data to both tables before running this test. Then change the system date to correspond to the various date intervals so	The unload to disk of the ArchiveRechargeDetail and the ArchiveFailedRechargeDetail will only occur on the dates specified.	As expected.

Prepared (also subject responsible if other)		No.			
Brendan Treacy					
Approved	Checked	Date	Rev	Reference	
		2008-05-26	PA1		

		that the unload to disk is performed and not performed when the date does not match.		
14	TR HI77256 - starc creates deadlocks in sybase	Restore data as set-up for Test 1 above. Run starc in one Unix window and in another window carry out a ps-ef|grep stats to monitor the closure of stats dr processes.	Statsmgr and child stats DR processes will closedown cleanly meanwhile starc will wait until statsmgr fully down before starting the archive. When the archive is complete, statsmgr will restart.	As expected.

Prepared (also subject responsible if other)		No.			
Brendan Treacy					
Approved	Checked	Date	Rev	Reference	
		2008-05-26	PA1		

2.2 HI43072 - **Adjustments DR for DA 7-10 not processed**

2.2.1 Description of fault

The fault occurs when an AIR's Adjustment record contains Dedicated Account info for Dedicated Account 7,8,9 or 10 so that it is effectively been ignored and is not displayed on the Transaction Details Tab of the Account History Screen.

2.2.2 Test environment set up

An AIR's Adjustment DR containing 6 records has been supplied with TR and will be used to verify this TR.

Additionally the tests performed for HH90386 below will further verify this TR also.

2.2.3 Test cases

Ref.	Test Condition	Test Data	Expected Result	Actual Result
1.	Test that when a AIR DR Adjustment is processed with data contained in Dedicated Account 7, 8, 9 and 10 fields that the Dedicated Accounts can be viewed on the Transaction Details Tab in Account History.	The sample AIR's Adjustment DR file should be copied into the ~/DRS/AIR directory.	Check the Transaction Details Tab in Account History and it should be observed that these Dedicated Account Fields in question do indeed contain the data that I set-up on the AIR Adjustment Record.	As Expected See Files below - \HI43072 – 353871000004 – AC HI43072 – 353871000005 – AC HI43072 – 353871000006 – AC

Prepared (also subject responsible if other)		No.		
Brendan Treacy				
Approved	Checked	Date	Rev	Reference
		2008-05-26	PA1	

2.3 HI60821 Malfunctioning Statistics DR Processing / HI51838 **Statsdr deadlock error: -1205**

2.3.1 Description of fault

Deadlocks occurred when processing high-volumes or a backlog by Stats DR Processing so the 'noholdlock' condition has been added to certain sql 'select' statements in the Process Stats Stored Procedures.

2.3.2 Test environment set up

1. Provision 5,000 subscribers using the MGU and MPU Command Line Utilities.

2. Generate 210,000 SDP Adjustment DRs and 40,000 various other types of DRs for subscribers in the range provisioned above using the DRGen utility.

This involves editing the DRGen.cfg file in directory /users/donalo/proj/R521/minsatah/src/DRGen

and then running command as follows :-

sh /users/donalo/proj/R521/minsatah/src/DRGen/DRGen.sh -b /users/brendant/DRGenFiles

3. There were two test runs of Stats DR where an accumulated backlog of DRs were released and for each of these runs I have attached the output to the Sybase Log :-

Test Run 1 consisted of 5,000 Subscribers with 1.1 M DR's of various types

Test Run 2 consisted of 15,000 Subscribers with 1.1 M DR's of various types

2.3.3 Test cases

Ref.	Test Condition	Test Data/Activity	Expected Result	Actual Result
1.	Test that when a backlog of Stats DR are released to Stats DR Processing that no deadlock errors occur	1. Stop statistics processing in the crontab and then kill the statsmgr process 2. Load DRs by putting them into	There will be no instances of deadlock error 1205 in /var/minsat/logs/minsat.errors	For Test Run 1 there were 15 deadlock errors encountered (as can be seen in Run1.txt above) while for Test Run 2 there were just 5 deadlock

Prepared (also subject responsible if other)		No.			
Brendan Treacy					
Approved	Checked	Date 2008-05-26	Rev PA1	Reference	

| | | the appropriate directory in /export/home/minsat/DRS.
3. Start statistics DR processing when the Processdrs are finished. | | errors (as can be seen in Run2.txt above) so there was a 66% reduction in deadlocking errors when subscribers were increased by 10,000 (reducing the DR Per Subscriber ratio from 1:220 to 1:73). It is expected that if the no of Subscribers was increased by a further 10,000 that no deadlocking would occur at all.

HI51838 - Run2.txt

HI51838 - Run1.txt

I referred decision to David S and he replied :-
I agree, this seems to be a significant reduction of the deadlock problem, and to completely eliminate any possibility of deadlock to occur in the database is too big a task yet. This is at least |

Prepared (also subject responsible if other)		No.			
Brendan Treacy					
Approved	Checked	Date	Rev	Reference	
		2008-05-26	PA1		

				much better than before, and I think that's good.

2.4　HH90386 - Most CSSTAT tables with no indexes, causing TableScans/Locks

Also includes TRs HI79190

2.4.1　Description of fault

Within Stats DR main procedures listed below, the select from both the history and error tables after processing batch, were causing the tablescans. Also the the main querys in these procedures were holding read locks for the duration of the process.

processstatsdrairsadj.sql
processstatsdrairspromo.sql
processstatsdrairsrefill.sql
processstatsdrairsvoucherr.sql
processstatsdrairsvvoucher.sql
processstatsdrbillableminutes.sql
processstatsdrsdpadj.sql
processstatsdrsdpbonus.sql
processstatsdrsdplifecycle.sql
processstatsdrsdpnegbal.sql
processstatsdrsdpnegbalbarred.sql
processstatsdrsdpsfeededuct.sql
processstatsdrsdpvvexpiry.sql

Additionally the error handling within the main procedure and sub-procedure calls below were not working correctly and would have heretofore resulted in Stats DR records in error not being marked as such.

saveactivationdetail.sql
saveclearedcredit.sql
savedeductionhistory.sql
savefailedrecharge.sql
saverechargedetail.sql
updateserviceclasschange.sql
updateservicefeededresult.sql

Prepared (also subject responsible if other)		No.		
Brendan Treacy				
Approved	Checked	Date	Rev	Reference
		2008-05-26	PA1	

2.4.2 Test environment set up

The DRGen utility was used to generate 100 records for each DR Type under test and the following DRGen help lists the parameter required to generate each file which is then copied into the relevant directory within ~/DRS/ which can be identified by referring to the DRStream table in the csacch database.

-A generate AIRS Adjustment DRs.
-P generate AIRS Promotion DRs.
-R generate AIRS Refill DRs.
-V generate AIRS Value Voucher DRs.
-E generate AIRS Error DRs.
-L generate SDP LifeCycle DRs.
-D generate SDP Adjustment DRs.
-B generate SDP Negative Balance Barred DRs.
-N generate SDP Negative Balance DRs.
-S generate SDP Service Fee Deduction DRs.
-X generate SDP Value Voucher Expiry DRs.
-O generate SDP Bonus Adjustment DRs.

For the test of ProcessStatsDRBillableMinutes the following additional set-up was required :-

Added the following OperatorPrefix records:

DO:AddOperatorPrefix:Country,44:Prefix,101;
DO:AddOperatorPrefix:Country,48:Prefix,99;
DO:AddOperatorPrefix:Country,48:Prefix,101;
DO:AddOperatorPrefix:Country,353:Prefix,99;

2.4.3 Test cases

Ref.	Test Condition	Test Data/Activity	Expected Result	Actual Result
1.	Test that the procedure ProcessStatsDRSDPAdj can process 100 SDP Adjustment records which will include a Rebate or Debit etc, resulting in the sub-procedures affected by TR also being executed for the relevant record types and records also written to history and	Use DRGen utility to generate 100 SDP Adjustment Records.	Procedure UpdateServiceClassChange will be executed where the Service Class has changed resulting in rows written to ServiceClassChange table Procedure SaveDeductionHistory will be executed for Deductions	Records were written to ServiceClassChange table :- ServiceClassChange from SDPAdj.xls Records were

Prepared (also subject responsible if other)		No.		
Brendan Treacy				
Approved	Checked	Date	Rev	Reference
		2008-05-26	PA1	

		error tables.		+ SaveRechargeDetails will also be executed.	written to DeductionHistory –
					DeductionHistory from SDPAdj.xls
				Procedure SaveRechargeDetail will be executed for Rebate or Debit resulting in records being written to RechargeDetail.	
				Procedure SaveClearedCredit will be executed resulting in records written to AccountCreditClearance table.	Records were written to Recharge Detail table - RechargeDetail from SDPAdj.xls
					Records were written to AccountCredit Clearance - AccountCreditCleara nce from SDPAdj.xls
				Successfully processed records will be written to DRSDPAccountAdjustment History while those where subscriber not found will be written to DRSDPAccountAdjustment Error.	
					Correct no. of records were written to DRSDPAccou ntAdjustmentHi story and DRSDPAccou ntAdjustmentE rror. C:\Documents and Settings\brendan.tre C:\Documents and Settings\brendan.tre

Prepared (also subject responsible if other)		No.		
Brendan Treacy				
Approved	Checked	Date 2008-05-26	Rev PA1	Reference

2.	Test that the procedure ProcessStatsDRAIRS Adj can process 100 AIRS Adjustment Records resulting in the sub-procedures affected by TR also being executed and records also written to history and error tables.	Use DRGen utility to generate 100 AIR Adjustment Records.	Procedure SaveClearedCredit will be executed where Dedicated Account Balances not zero resulting in records written to AccountCreditClearance table. Successfully processed records will be written to DRAIRSAdjustmentHistory while those where subscriber not found will be written to DRAIRSAdjustmentError.	Records were written to AccountCredit Clearance :- AccountCreditCleara nce from AIRAdj.xls Correct no. of records written to DRAIRSAdjust mentHistory and DRAIRSAdjust mentError. DRAIRSAdjustmentHi story.xls DRAIRSAdjustmentE rror.xls
3.	Test that the procedure ProcessStatsDRAIRS Promo can process 100 AIRS Promotion DR Records resulting in the sub-procedures affected by TR also being executed and records also written to history and error tables.	Use DRGen utility to generate 100 AIRS Promotion DR Records.	Procedure SaveClearedCredit will be executed where Dedicated Account Balances not zero resulting in records written to AccountCreditClearance table. Successfully processed records will be written to DRAIRSPromotionHistory while those where subscriber not found will be written to DRAIRSPromotionError.	Correct no of records were written to AccountCredit Clearance :- AccountCreditCleara nce from Promo.xls Correct no. of records written to DRAIRSPromo tionHistory and DRAIRSPromo tionError

Prepared (also subject responsible if other)		No.		
Brendan Treacy				
Approved	Checked	Date	Rev	Reference
		2008-05-26	PA1	

				DRAIRSPromotionHis tory.xls DRAIRSPromotionErr or.xls .
4.	Test that the procedure ProcessStatsDRAIRS Refillcan process 100 AIRS Promotion DR Records resulting in the sub-procedures affected by TR also being executed and records also written to history and error tables.	Use DRGen utility to generate 100 AIR Refill DR Records.	Procedure SaveRechargeDetail will be executed for all records. Procedure SaveActivationDetail will be executed where Account Status after Refill = 'Active'. Procedure SaveClearedCredit will be executed where Dedicated Account Balances not zero resulting in records written to AccountCreditClearance table. Successfully processed records will be written to DRAIRSRefillHistory while those where subscriber not found will be written to DRAIRSRefillError.	Records were written to RechargeDetai l C:\Documents and Settings\brendan.trea Records written to ActivationDetai l table. C:\Documents and Settings\brendan.trea Records were written to AccountCredit Clearance. C:\Documents and Settings\brendan.trea Correct no. of records written to DRAIRSRefillH istory and DRAIRSRefillE rror

Prepared (also subject responsible if other)		No.		
Brendan Treacy				
Approved	Checked	Date	Rev	Reference
		2008-05-26	PA1	

				C:\Documents and Settings\brendan.tre: C:\Documents and Settings\brendan.tre:
5.	Test that the procedure ProcessStatsDRAIRS VoucherErr process 100 Value Voucher DR Records resulting in the sub-procedures affected by TR also being executed and records also written to history and error tables.	Use DRGen utility to generate 100 Records.	Procedure SaveFailedRecharge will be exececuted where Operation field on input table set to 1,2 or 3 Successfully processed records will be written to DRAIRSVoucherErrHistory while those where subscriber not found will be written to DRAIRSVoucherErrError.	Records were written to SaveFailedRec harge table C:\Documents and Settings\brendan.tre: Correct no. of records written to DRAIRSVouch erErrHistorywh ile and DRAIRSVouch erErrError C:\Documents and Settings\brendan.tre: C:\Documents and Settings\brendan.tre:
6.	Test that the procedure ProcessStatsDRAIRS VVoucher process 100 Voucher Error DR Records resulting in the sub-procedures affected by TR also being executed and records also written to	Use DRGen utility to generate 100 Records.	Procedure SaveClearedCredit will be executed where Dedicated Account Balances not zero resulting in records written to AccountCreditClearance table.	Correct no. of records were written to AccountCredit Clearance :- AccountCreditCleara nce from VVoucher.x

Prepared (also subject responsible if other)		No.		
Brendan Treacy				
Approved	Checked	Date	Rev	Reference
		2008-05-26	PA1	

		history and error tables.		Successfully processed records will be written to DRAIRSValueVoucherHistory and those where subscriber not found will be written to DRAIRSValueVoucherError.	Correct no. of records written to DRAIRSValue VoucherHistor yand DRAIRSValue VoucherError DRAIRSValueVouche rHistory.xls DRAIRSValueVouche rError.xls
7.		Test that the procedure ProcessStatsDRBillabl eMinutes process 100 Billable Minutes DR Records resulting in the sub-procedures affected by TR also being executed and records also written to history and error tables.	Loaded CDR file TTFILE-1902. This is basically a repeat of the test in regression test script 11_DRS_1 902.sh. I then created TTFILE-0202 containing 4 of the records from TTFILE-1902 and loaded that file.	Records will be written to BillableMinute table. Successfully processed records will be written to DRBillableMinutesHistory and those in error will be written to DRBillableMinutesError.	As expected, there were 8 records written to DRBillableMin utesHistory and 8 records in BillableMinute BillableMinute.xls I checked that there were 4 additional records in DRBillableMin utesHistory and that 4 of the records in BillableMinute were updated (Value and Number field values doubled).

Prepared (also subject responsible if other)		No.		
Brendan Treacy				
Approved	Checked	Date 2008-05-26	Rev PA1	Reference

				 DRBillableMinutesHist ory.xls
8.	Test that the procedure ProcessStatsDRSDPLi feCycle process 100 Life Cycle DR Records resulting in the sub-procedures affected by TR also being executed and records also written to history and error tables.	Use DRGen utility to generate 100 Records.	Procedure SaveActivationDetail will be executed where Account Status becomes 'Active'. Procedure SaveClearedCredit will be executed where Dedicated Account Balances not zero resulting in records written to AccountCreditClearance table. Successfully processed records will be written to DRSDPLifeCycleChangeHi story those where subscriber not found will be written to DRSDPLifeCycleChangeEr ror.	Records written to ActivationDetai l table. ActivationDetail from LC.xls Records were written to AccountCredit Clearance :- AccountCreditCleara nce from LC.xls Correct no. of records written to DRSDPLifeCy cleChangeHist ory DRSDPLifeCy cleChangeErro r DRSDPLifeCycleChan geHistory.xls DRSDPLifeCycleChan geError.xls
9.	Test that the procedure ProcessStatsDRSDPS FeeDeduct process	Use DRGen utility to generate	Procedure ServiceFeeDeductionHistor y will be executed for all input records.	Recprds written to ServiceFeeDe ductionHistory

Prepared (also subject responsible if other)		No.			
Brendan Treacy					
Approved	Checked	Date	Rev	Reference	
		2008-05-26	PA1		

	100 Service Fee Deduction DR Records resulting in the sub-procedures affected by TR also being executed and records also written to history and error tables.	100 Records.	Procedure UpdateServiceClassChange will be executed where the Service Class has changed resulting in rows written to ServiceClassChange table. Procedure SaveClearedCredit will be executed where Dedicated Account Balances not zero resulting in records written to AccountCreditClearance table. Successfully processed records will be written to DRSDPServiceFeeDeductionHist those where subscriber not found will be written to DRSDPServiceFeeDeductionError.	:- ServiceFeeDeduction History.xls Records written to ServiceClassChange :- ServiceClassChange. xls Records were written to AccountCredit Clearance :- AccountCreditCleara nce from SFeeDeduct Correct no. of records written to DRSDPServiceFeeDeduction Hist DRSDPServiceFeeDeduction Error ServiceFeeDeduction History.xls DRSDPServiceFeeDe ductionError.xls

Prepared (also subject responsible if other)		No.		
Brendan Treacy				
Approved	Checked	Date 2008-05-26	Rev PA1	Reference

| 10. | Test that the procedure ProcessStatsDRSDPVVExpiry process 100 Value Voucher Expiry DR Records resulting in the sub-procedures affected by TR also being executed and records also written to history and error tables. | Use DRGen utility to generate 100 Records. | Procedure UpdateServiceClassChange will be executed where the Service Class has changed resulting in rows written to ServiceClassChange table.

Procedure SaveClearedCredit will be executed where Dedicated Account Balances not zero resulting in records written to AccountCreditClearance table.

Successfully processed records will be written to DRSDPValueVoucherExpiryHistory those where subscriber not found will be written to DRSDPValueVoucherExpiryError. | Records written to ServiceClassChange :-

ServiceClassChange from VV Expiry.xls

Above file from previous run so will not match up with re-run files below.

AccountCreditClearance from vvexpiry.xls

Correct no. of records written to DRSDPValueVoucherExpiryHistory DRSDPValueVoucherExpiryError

DRSDPValueVoucherExpiryHistory.xls

DRSDPValueVoucherExpiryError.xls |
| 11 | TR **HI79190** - ServiceClassChange DRs fail with SQL error 559 | As for Test 1 above. | As for Test 1 above.

This is quite simple to test as a successful run of the procedure will validate that problem has been fixed | As expected. |

Prepared (also subject responsible if other)		No.			
Brendan Treacy					
Approved	Checked	Date	Rev	Reference	
		2008-05-26	PA1		

			hence reference above to Test 1 above.	

Prepared (also subject responsible if other)		No.			
Brendan Treacy					
Approved	Checked	Date	Rev		Reference
		2008-05-26	PA1		

2.5 HI36483 - Timeout error on Minsat Report. smms goes into continuous loop

2.5.1 Description of fault

Smms goes into continuous loop and drains resources for stats processing.

2.5.2 Test environment set up

This TR is very similar to TR 4144 which suggests that to recreate fault it is necessary to reduce the number of locks in the server to quite low, open a cursor on a large table and lock lots of rows and then run the smms and see does the smms handle the error.

So, I wrote an SQL procedure which applies 50,000 updates to the RechargeDetail table and thus applying exclusive locks, I then set locks to 1000 using sp_configure and run the SQL procedure. I then executed smms in -vvgg mode.

RechargeDetailRead SubStatusChangeRe
Lock.sql   adLock.sql

2.5.3 Test cases

Ref.	Test Condition	Test Data	Expected Result	Actual Result
1.	Recreate Sybase error -16843189 as described above.	Use the test data created for HI60821 and run SQL procedure to lock 50,000 rows, reducing Sybase locks to 1000 using sp_configure before running smms	It is expected that by following the above method that the Sybase error -16843189 will occur.	Fault cannot be recreated so will have to be passed as being expected to resolve reported problem but not verifed. This decision was arrived at by consultation with Duala O'Brien in Telcotec.