Prepared by Firoz Ahmad Siddiqui

Training Topic:-

1- BASE24 Transaction flow.
2- Response code.
3- Tran code
4- Message Type
5- BASE24 file (TLF/PTLF/CAF/ATD/DCT/CLF etc.)
6- SAF issue.
7- Replicator/Extractor
8- Process abnormal
9- About interchange interface.
10- How to fetch data and read audit log.
11- How to pak and unpak for BASE24 file
12- Disk space management.
13- How to check queue and how to write
14- How to fup dup BASE24 file
15- ATD hanging issue
16- How to monitor transaction throw macro
17- Interchange Cutover Issue
18-Tandem backup related.
19- EOF checking for all file.
20- ATM configuration
21- SPOOLCOM
22- Architecture of Data File, Which Data Disk holds what files
23- VISA and Master card BIN upload overview and importance of Sproute table
24- Difference between start -stop of station and deliv command with reset/load option
25- Why warmboot is given to DH and how it impact the system
26- Overview on Issuing stations /Acauring stations for interfaces and respective ILF files
27- First level troubleshooting for any incident observed on Base24 Station.
28- Overview of BIN and Branch updation and how it reflects in the system.

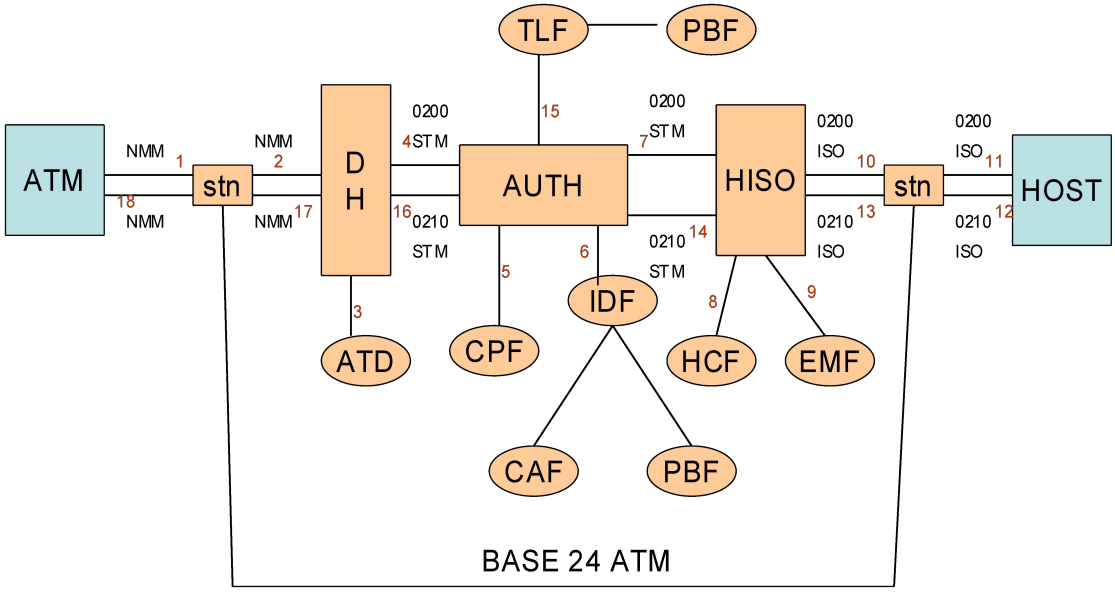
1- BASE24 Transaction flow.

Base24-

Base24 is a switching software application, which routes the requests (transaction) from ATM, POS, and other devices to their appropriate hosts and hosts reply to their appropriate requester device . It is an intelligent switching device. The BASE24 product is a combination of the Base module and device specific module, The BASE24 is divided into 4 basic modules.

1- DH (Device handler Module)
2- AUTH (Authorization Module)
3- H-ISO (Host ISO module)
4- XP-NET

Transaction Flow-



1-At first the customer will insert the ATM card into the machine.
2-After the card is accepted, the magnetic card reader sensors the card in which the data are digitally encoded. It collects the information from the magnetic strip. Which is placed at the back side of the card.
3-After fetching the information the ATM encodes the information in the formats of its own language NMM (Native Message Mode) and sends it to the XPNET.
4-The first request from the ATM is to DH through XPNET to check for CARD, PIN and BANK details.
5-The DH process reformats the NMM message to SIM (Standard Internal Message).
6-After checking with ATD , DH set an authorization timer and send a 0200 request message to the authorization process.
7-Within AUTH the message goes to CPF (Card Prefix File), the CPF defines each card prefix that can be processed within a BASE24 logical network. One record must exist in the CPF for each card prefix to be processed.
8-If within the CPF the prefix doesn't match with the existing prefix, and then the message is routed to some other routing method to channel it towards proper direction.
9-In CPF the message is searched with respect to a primary key which is made of the prefix. With the help of that prefix it then searches for the FIID (Financial Institution ID).
10- Then it checks the IDF. IDF contains one record for each institution participating in the logical network and defines processing for each institution. The IDF contains routing tables for transaction routing within a BASE24 product and each institution's parameters for cards, dates, processing control and sharing. The key to records in the IDF is the FIID for the financial institution. For the help the FIID, it gets the DPC number HISO process where it is to be destined.
11-After IDF check, the CAF and the PBF is also checked (only for offline transaction).
12- The CAF checks for the Expiry date and the card limit.
13-The PBF checks for the offline approval.

14-From AUTH the message goes to HISO.
15-The Host Interface process reformat the internal message into an external message. In HISO it goes to HCF (Host configuration File) and checks the information of the HOST.EMF (External Message File) checks the bit set.
16-HISO set an outbound request timer and sends the 0200 request message to the HOST.
17-The HOST approve the request and return a 0210 response message to the Host Interface process.
18-The HISO deletes its outbound request timer, reformate the external message into an internal message. And sends the 0210 response message to the authorization process.
19- The AUTH process logs the response to the TLF (Transaction Log File) and send the 0210 response message to the Device Handler process.
20- The DH process deletes its authorization timer and reformats the 0210 response message into a native-mode response that it sends to the ATM.
21-The ATM sends a native-mode message to the DH process to indicate that the transaction completed as authorized.

2- Response code.

Base24 Response Code in TLF:

048	DCC transaction first leg
000	Approved with Balances
001	Approved and No Balance for Display
050	Unauthorized Usage
051	Expired Card
052	Invalid Card
053	Invalid Pin
054	Database Problem
055	Ineligible Transaction
056	Wrong Account Selection
057	Transaction Not Permitted
058	Insufficient Funds and No Amount3
059	Insufficient Funds and with Amount3
061	Withdrawal limit would be Exceeded
062	Pin Tries Exceeded 424
063	Withdrawal Limit already reached
064	Invalid Credit Card Cash Advance Amount
066	Statement Information Not Available
068	External Decline
069	No Sharing Arrangement
070	System Error
071	Contact Card Issuer
072	Destination Not Available
073	Router Problem
074	Bad Track 2
078	Invalid Account
150	Stolen Card
168	External Decline due to Database Error

Base24 Response Code in PTLF:

000	Approved With Balances
001	Approved Without Balances
050	Declined (Destination Not Available)
051	Expired Card
052	Pin Tries Exceeded
055	Invalid Transaction
056	No Support
057	Lost Or Stolen Card Status
058	Invalid Status
059	Restricted Status
060	Account Not in the CAF
061	PBF Record Not Found
064	Bad Track 2
069	Bad Message Edit
070	No IDF
073	Invalid routing to Authorization Service Field
074	Unable To Authorize
076	Insufficient Funds
078	Duplicate Transaction
082	Duplicate Transaction
088	Please Call for Referral
089	Card Status Inactive Or Closed

095	Amount Over Maximum
096	Pin Required
097	Mod 10 Check
099	Bad PBF
100	Unable to Process the Transaction
101	Issue A Call
105	Card Not Supported
107	Daily Usage Limit Reached
113	Timeout
200	Invalid Account
201	Incorrect Pin
203	Transaction Requires Administrative Card
205	Invalid Advance Account
206	Caf Not Found
903	CAF Status 3
909	Capture

3- Tran code

ATM tran code

1000000 W/D GENERIC
1001000 W/D FROM CHK
1011000 W/D FROM SAV
1031000 W/D FROM CC

2000010 DEP TO CHK
2000110 DEP TO SAV

3001000*INQ TO CHK
3011000*INQ TO SAV
3031000*INQ TO CC

4001010*XFER CHK-CHK
4001110*XFER CHK-SAV
4011010*XFER SAV-CHK
4011110*XFER SAV-SAV
4031010*XFER CC-CHK
4031110*XFER CC-SAV
4001600*XFER CHK-OTH
4011600*XFER SAV-OTH
4031600*XFER CC-OTH
4060010*XFER OTH-CHK
4060110*XFER OTH-SAV *
4060600*XFER OTH-OTH

4401000*TXFR CDCI
4401010*TXFR CDCI
4401110*TXFR CDCI *
4411000*TXFR CDCI
4411010*TXFR CDCI
4411110*TXFR CDCI

6101000*LOG ONLY 1
6102000*LOG ONLY 2
6103000*LOG ONLY 3
6104000*LOG ONLY 4

7001000*STMT PRT CHK
7011000*STMT PRT SAV
7031000*STMT PRT CC
7060000*STMT PRT OTH

7511000*PRPD FRM SAV
7501000*PRPD FRM CHK

7600000*PRPD ACK

7700000*PRPD REGN

8100000*PIN CHANGE

9601810*BIL PRES-CHK *
9611810*BIL PRES-SAV
9601830*BIL PAY -CHK
9611830*BIL PAY -SAV

9701810*BIL PRES-CHK *
9711810*BIL PRES-SAV
9701830*BIL PAY -CHK
9711830*BIL PAY -SAV

D100114*CASH-DEP SAV
D100014*CASH-DEP CHK
D100110*CASH-DEP SAV
D100010*CASH-DEP CHK
D188110*CRDL-DEP SAV
D188010*CRDL-DEP CHK

FD11000*FXD DEPOSIT
MR11000*MOB NO REG
N200000*GPIN-ISS
N201000*GPIN-ISS-CHK
N211000*GPIN-ISS-SAV
N231000*GPIN-ISS-CC

PA00010*PAN-CRD(CHK)
PA00110*PAN-CRD(SAV)
PA88110*PAN-CRL

Z100000*GPIN-GEN
Z101000*GPIN-GEN-CHK
Z111000*GPIN-GEN-SAV
Z131000*GPIN-GEN-CC

Z200000*GPIN-VER
Z201000*GPIN-VER-CHK
Z211000*GPIN-VER-SAV
Z231000*GPIN-VER-CC
49 ISO TRANSFER ACCT 2ND LEG
71 AHI PIN VERIFICATION
93 TOPPING LEG 1

4- Message Type

Use of Message types:
- Each message structure is accompanied with one message type.
- Each message type is of 4 bytes.
- Every case there is one message type for request and corresponding one message type for response.

Different Message Types:
Financial Transaction-
0200-Request
0210-Response

Reversal-
0420-Request
0421-Repeat reversal
0430-Response

Handshaking or log on request / response:-
0800-Request
0810-Response

SAF advice Message :-
0220-From issuer (request)
0221-Reversal request
0230-From Acquirer

0205- Mini statement Request
0215- Mini statement Response

9901-Surcharge (i.e., if it is a not-on-us transaction, then debit the amount from the owner's account)

9906- For multiple account selection
5400 -Adjustment request
0500-Batch / shift / day close request

5- BASE24 file (TLF/ATD/PTLF/CAF/ATD/DCT/CLF etc.)

Transaction Log File (TLF): - The TLF contains one record for each transaction occurring at an ATM in the logical network and each transaction processed by the BASE24 transaction processing system during a business day. TLF records are maintained for both cardholder transactions and terminal balancing transactions. One TLF is maintained for each business day, and each TLF can be saved for the number of days specified in the Logical Network Configuration File (LCONF). The TLF provides an audit trail of ATM activity.
The BASE24-atm Authorization process logs all of the transactions it processes to the Transaction Log File (TLF).
Contains TLF records for every cardholder transaction and for every terminal balancing transaction.

ATM Terminal Data files (ATD): - The Terminal Data files define the characteristics of the terminal, including the transaction security parameters, sharing parameters, transaction totals, terminal owner, and device-dependent data. They are also used to preserve transaction context. Data entered on ATD screens is stored in different files, depending on the type of terminal being defined. For Diebold 10XX/478X and NCR 5XXX terminals, dynamic and static data is stored in the ATM Terminal Data Dynamic File- general data (ATDD1) and ATM Terminal Data Static File-general data (ATDS1), respectively. For all other types of ATM terminals, data is stored in the ATM Terminal Data File (TDF).
Contains the characteristics of the terminal, sharing information, hopper contents, activity performed at the terminal and device specific information.
- ATDD1
- ATDD2
- ATDS1

Positive Balance File (PBF): - The PBF contains one record for each application account whose issuer uses the Positive Balance Authorization method, Parametric Authorization method, or Positive Customer with Balances/History Authorization method. Each PBF record contains available and ledger balances for the account. The PBF is used to authorize transaction requests without having to forward them to a back-end host authorization system. Institutions can maintain up to three different PBFs: one for checking accounts (DDAs), one for savings (SAV) accounts, and one for credit (CCD) accounts.

The Positive Balance Authorization method uses the Cardholder Authorization File (CAF) and the Positive Balance File (PBF).

Cardholder Authorization File (CAF): - The Cardholder Authorization File (CAF) contains one record for each cardholder whose card-issuing institution uses the Positive Authorization method or Positive Balance Authorization method. Each CAF record contains authorization parameters and usage accumulation information for a specific cardholder. The Positive Authorization method uses the Cardholder Authorization File (CAF). The Authorization process authorizes transactions using positive identification.
Contains card usage controls, withdrawal limits, and deposit credit limits, accounts associated with the cardholder and card usage accumulation.

Card Prefix File (CPF)[Card and Cardholder Limits]: -> The CPF contains one record for each card prefix used in a BASE24 logical network. CPF records define the characteristics of each prefix and contain prefix-specific parameters that can be used to control aspects of authorization processing at the prefix level. The CPF also allows institutions to establish authorization parameters for selected prefix groups.

The Card Prefix File (CPF) defines each card prefix that can be processed within the BASE24 logical network. One record exists in the CPF for each card prefix to be processed. Each CPF record contains processing parameters and authorization limits for that specific card prefix. [Deposit Credit Limits]

Contains Track I and II information, processing information, withdrawal limits, PIN information and deposit credit limits.

Institution Definition File (IDF): - The IDF contains one record for each institution participating in the logical network. Each record establishes how processing is handled for that institution and includes routing tables for transaction routing within BASE24 products and parameters for cards, dates, processing control, and sharing.
The authorization level is specified in the AUTH LVL fields on IDF screen 9. The possible authorization levels are online authorization (level 1), offline authorization (level 2), and online/offline authorization (level 3). [Include Screen 9 & Screen 2]
Contains authorization file names, card parameters, processing and withdrawal period parameters, routing table, processing dates and sharing information.

Host Configuration File (HCF): - The HCF contains one record for each unique data processing center (DPC) and Host Interface process pair in the logical network. Each record contains pertinent information about host message formats, timer limits, and request/response sequences. The HCF is used to establish BASE24-to-host communication links for transaction message traffic routed to a host authorization system.
It does this based on an internal table of stations that it builds at initialization from the STATION fields on screen 2 of its records in the Host Configuration File (HCF). There various timers are set in the Host Configuration File (HCF).
Contains timer limits, processing flags and stations.

6- SAF issue.

Store and Forward File (SAF): - Contains approved transactions and reversals when the host communications link to BASE24 is down.

status s P1E^HISO06
delive p P1E^HISO06,/status all/
stop p P1E^HISO06

DPC- 0070
Hiso- P1E^HISO06

```
  ------------------------------------------------------------
station                 Port & IP
S1EHOSTCLI117           16242:10.16.34.102
S1EHOSTCLI118           16243:10.16.34.102
S1EHOSTCLI119           16244:10.16.34.102
S1EHOSTCLI120           16245:10.16.34.102
S1EHOSTCLI121           16331:10.16.34.102
------------------------------------------------------------
Hisp process-- P1E^HISO06
------------------------------------------------------------
DPC- ( DPC 0121 ,DPC 0120,DPC 0119, DPC 0118,DPC 0070,
```

SAF details-

```
   0:  .F..[.5..M...>ISO01600001312210..(..... 1646937536080001370110000
  35:  0000000200000322911412151344481215356200115046420000027469375360800013
  70:  7=24091262971814        SACWF687        SACWF687        79ICICI BANK
 105:          \BRANCH RATANGARH         \RATANGARH    \000331022 NOR3563560
 140:  120000PRO1+0000133608PRO11100P1236080150003800003EFT
-------------------------------------------------------------------------------
```

SAF path - $DATA03.PRO6DATA.SAF28

1- Check saf record and from there find card no
2- Go to caf and from account find fiid (start 4 digit from account)
3- Go to idf on page 9 and find hiso
4- From hiso find dpc no through ems after delive status
5- Go hcf and change max saf retry from 0 to 1 and f5 (MAX SAF RETRY: 0)
6- Take ems with filter hiso
7- Go to ncs and stop start hiso process and give delive status command
8- Check ems now saf file will clear
9- Now come in hcf and change max saf retry from 1 to 0 and f5
10- Then again go to ncs and stop start hiso and give delive status command
11- Send mail for saf details to --
venkata.tadinada@ext.icicibank.com and venkata.tadinada@ext.icicibank.com

7- Replicator/Extracter abbended

E/R Training

Brief Description on E/R:

GGS who has made online data copying possible means in other words replication of data from 1 tandem to another, GG is Golden Gate the company who has created this software and which helps us replicating data from 1 database to any other type of database here database which we are using is enscribe to enscribe on tandem.
Coping of data from one file to another is called Extracting data from production Tandem and Replicating it to Disaster Recovery tandem means extractor and replicator i.e. E/R.

1) What is ER
it is extractor and replicator, means it extracts data from source files i.e. in production tandem and replicat data in target files i.e. in DR Tandem. It is the software by golden gate softwares. Like other softwares have it has its own Prompt its name is ggsci i.e. golden gate software command interface.

2) why we use E/R
Because, by E/R facailty we can copy any database files data to any database file online, it means by the time we have any transaction or any changes in our given file in production tandem will be replicated in our DR tandem in Mumbai within 10 - 15 seconds.
Our second tandem i.e. our DR tandem is called disaster recovery tandem means if any disaster or any thing wrong or any natural calamities happens and our production tandem stops working then even the last data in production we will find it there in DR Tandem and in this way because of E/R facility u will be having all the data at any point of time.

3) **how it works**

There are 4 main parameter files between which E/R works, they are Manager, Logger,Extractor, Replicator. In this we write our suitable parameters, add it in GGS software and run them as required.

Manager - It establishes a connection through network to the remote tandem by tcp-ip process name & port no

Logger – It collects data from source files and collects it in logtrails (if it is non audited - non audited then only logger is used, if it is audited to audited then logger configuring is not required you have to go by only 3 stages i.e. manager, extractor, replicator in this case extractor instead of taking data from logger it takes from tandems inbuilt trail in $AUDIT.ZTMFAT.AA000000). Loggers plays an important role in whole E/R process. Logger always has its eye on the files which names are specified in its parameter, as soon as any Datas / Transactions are inserted, updated or deleted in files specified in Logger parameter, that changes are captured and are send to Log-Trails.

Extractor – It reads the parameter specified in particular Extractor and accordingly takes data from Log-Trails and puts it in Exttrail (if replication is to be done in same tandem) / Rmttrail (if replication is to be done in remote (other) tandem). For connecting it to the other Tandem in Extractor parameter we specify IP Address, Port No and TCPIPPROCESSNAME of the remote Tandem.

Replicator – Replicator is used to retrieves data from Remote trail, the trail in which Extractor writes. Replicator collects all the data from the trail as it is stored by Extractor and stores that particular data in their respective file.

–----------

Commands

To Stop Extractor
GGSCI> stop extract <extractor name>
To Start Extractor
GGSCI> start extract <extractor name>
To Stop Replicat
GGSCI> stop Replicat <replicator name>
To Start Replicator
GGSCI> start Replicat <replicator name>

If Extractor or Replicator Abends then for Trouble shooting we can see reports for that Extractor or Replicator by this Command and solve the problem

To View Report
GGSCI>view report <extractor/replicator name>

–----------

8- Process abnormal :-

1- Check EMS with filter
2- Check through inspect procedure

Inspect procedure:-

```
If process going abnormal , then troubleshooting procedure------
step1--

stop p P1M^C10003
cont p P1M^C10003,purge
start p P1M^C10003
status p P1M^C10003

step2---
ncs
o n
4 > info p P1M^C10003,prog
   PROCESS                                  PROGRAM
   ---------------------------------------- --------------
 E \SBI01.P1M^NODE.P1M^C10003               $DATA26.PROXOBJ.C1000Q


step3-
$DATA26.PROXOBJ > fup info * where modtime > 12:20

----------------
$DATA26 PROXOBJ 4> fi *zzs*

$DATA26.PROXOBJ

              CODE             EOF   LAST MODIFIED  OWNER  RWEP    PExt    SExt
ZZSA0128       130         20094976 12FEB2013 12:24 242,255 NGNG     28      28
ZZSA0447       130         20094976 12FEB2013 12:32 242,255 NGNG     28      28
```

```
ZZSA0946       130         20094976 12FEB2013 12:25 242,255 NGNG     28     28
ZZSA1139       130         20094976 11FEB2013 14:07 242,255 NGNG     28     28



$DATA01 P610CNTL 3> v $DATA26.PROXOBJ
$DATA26 PROXOBJ 4> fi *zzs*


$DATA26 PROXOBJ 6> inspect                                (command)
INSPECT - Symbolic Debugger - T9673H01 - (17FEB2012)   System \SBI01
Copyright Hewlett Packard Company 1983,1985-1999,2000-2011
--add program ZZSA4077                                      // (command - add program ZZSA4077)
              Program
Num Program ID Name       Type         State Location
 *1   04,00915 $SMC03     TNS SAVE     STOP  #GGS_OS_STOP.#834(GUARD) + %121I
-$SMC03-d stm                                              // (type d stm)    (some time type t )
STM =
  TYP =
    BYTE[0] = "0210"
  PROD_ID =
    BYTE[0] = "01"
  DPC_NUM = 0
  TRAN_CDE =
    BYTE[0] = "86"
  FROM_ACCT_TYP =
    BYTE[0] = "00"
  TO_ACCT_TYP =
    BYTE[0] = "00"
  RTE =
    STAT =
      BYTE[0] = "00"
    NOT_ON_US_FLG = "0"
    ORIG_PRO_NAME =
      BYTE[0] = "P1M^C10003      "
    HLD_ORIG_PRO_NAME =
      BYTE[0] = "0000000000000000"
    ORIGINATOR = "1"
    RESPONDER = "5"
    EXCPT_FLG = "0"
  CRD_LN =
    BYTE[0] = "PRO1"
  RCV_INST_ID_NUM =
    BYTE[0] = "45915000000"
  CRD_FIID =
    BYTE[0] = "C012"
  SHRG_GRP[0] = "A                       "
  TRAN_DAT =
    YY =
      BYTE[0] = "13"
    MM =
      BYTE[0] = "02"
    DD =
      BYTE[0] = "11"
  TRAN_TIM =
    HH =
      BYTE[0] = "13"
    MM =
      BYTE[0] = "59"
    SS =
      BYTE[0] = "29"
    TT =
      BYTE[0] = "54"
  USER_FLD1 = "0"
  TIM_OFST = 0
  POST_DAT =
    YY =
      BYTE[0] = "13"
    MM =
      BYTE[0] = "02"
    DD =
      BYTE[0] = "12"
  ACQ_ICHG_SETL_DAT =
    YY =
```

```
    BYTE[0] = "00"
  MM =
    BYTE[0] = "00"
  DD =
    BYTE[0] = "00"
ISS_ICHG_SETL_DAT =
  YY =
    BYTE[0] = "00"
  MM =
    BYTE[0] = "00"
  DD =
    BYTE[0] = "00"
TERM_OWNER_FIID =
  BYTE[0] = "C021"
TERM_ID =
  BYTE[0] = "S10I100026035   "
TERM_LN =
  BYTE[0] = "PRO1"
BRCH_ID =
  BYTE[0] = "D   "
REGN_ID =
  BYTE[0] = "SBBJ"
SEQ_NUM =
  BYTE[0] = "2330        "
DEP_SETL_IMP_FLG = "0"
ADJ_SETL_IMP_FLG = "0"
PROC_ACCT_IND = "0"
TERM_CAPTURE_FLG = "0"
RQST =
  TRACK2 =
    BYTE[0] = ";4591510079296237=171012687900000000?0000"
  PIN_OFST =
    BYTE[0] = "687900000000?0000"
  MBR_NUM =
    BYTE[0] = "000"
  TERM_TYP =
    BYTE[0] = "30"
  ORIG_CRNCY_CDE =
    BYTE[0] = "356"
  MULT_CRNCY =
    AUTH_CRNCY_CDE =
      BYTE[0] = "000"
    SETL_CRNCY_CDE =
      BYTE[0] = "000"
    AUTH_CONV_RATE =
      BYTE[0] = "00000000"
    SETL_CONV_RATE =
      BYTE[0] = "00000000"
    CONV_DAT_TIM = 3472328296227680304
  ACQ_INST_ID_NUM =
    BYTE[0] = "60020610026"
  MULT_ACCT = "0"
  RVSL_CDE =
    BYTE[0] = "00"
  FROM_ACCT =
    ACCT_NUM =
      BYTE[0] = "19/10       531.00-"
  USER_FLD4 = "0"
  TO_ACCT =
    ACCT_NUM =
      BYTE[0] = "O TRANSFER     20/1"
  PIN =
    BYTE[0] = "0000000000000000"
  PIN_SIZE =
    BYTE[0] = "00"
  PIN_TRIES = "0"
  PIN_KEY =
    BYTE[0] = "0000000000000000"
  PIN_FRMT = "0"
  PINPAD_CHAR = "0"
  ANSI_OFST = 0
  USER_KEY =
    BYTE[0] = "0000000000000000"
  USER_FLD4A =
```

```
    BYTE[0] = "00"
  CUST_BAL_INFO = "0"
  CUST_BAL_DSPY = "0"
  AMT_1 = 6072350129424179248
  AMT_2 = 4048507577579872288
  AMT_3 = 0
  DEP_BAL_CR = 808141633
  AVAIL_DEP_TYP = "0"
  DEP_TYP = "0"
  COMPL_REQ = " "
  RESP =
    BYTE[0] = "DEP"
  TERM_NAME_LOC =
    BYTE[0] = "2NDATM @KALWAR RD, JHOTWA"
  TERM_OWNER_NAME =
    BYTE[0] = "SBBJ 2NDATM @KALWAR R "
  TERM_CITY =
    BYTE[0] = "JAIPUR       "
  TERM_ST_X =
    BYTE[0] = "RJ "
  TERM_CNTRY_X =
    BYTE[0] = "IN"
  ORIG =
    SEQ_NUM =
      BYTE[0] = " WDL        "
    TRAN_DAT =
      BYTE[0] = "    "
    TRAN_TIM =
      HH =
        BYTE[0] = "  "
      MM =
        BYTE[0] = "  "
      SS =
        BYTE[0] = "  "
      TT =
        BYTE[0] = "  "
    B24_POST_DAT =
      BYTE[0] = "    "
  TERM_TRAN_ALLOWED = "0"
  TRACE_NUM =
    BYTE[0] = "      "
  TERM_ST =
    BYTE[0] = "00"
  TERM_CNTY =
    BYTE[0] = "000"
  TERM_CNTRY =
    BYTE[0] = "356"
  USER_FLD5 =
    BYTE[0] = "0000000000000000"
  PRI_DEST =
    BYTE[0] = "                "
  ALT_DEST =
    BYTE[0] = "                "
  AUTH_DEST =
    BYTE[0] = "                "
  DEST_ORDER = " "
  RTE_GRP =
    BYTE[0] = "00000000000"
  SAVE_ACCT =
    FROM_ACCT_TYP =
      BYTE[0] = "00"
    TO_ACCT_TYP =
      BYTE[0] = "00"
  POSTAL_CDE =
    BYTE[0] = "302003    "
  USER_FLD6 =
    BYTE[0] = "00000000000000000000000000000000000000000000"
  FRMT = "0"
  PRNT_SIZ =
    BYTE[0] = "000"
ENTRY_TIM = 212227412762439977
EXIT_TIM = 0
RE_ENTRY_TIM = 0
CRD_ACCPT_ID_NUM =
```

```
    BYTE[0] = "00000000000"
  FRWD_INST_ID_NUM =
    BYTE[0] = "00000622018"
  CRD_ISS_ID_NUM =
    BYTE[0] = "00000000000"
  AUTH_ID_RESP =
    BYTE[0] = "716026"
  TRACK3 =
    BYTE[0] = "000000000000000000000000000000000000000000000000000000000000000000000
    000000000000000000000000000000000000000000000"
  MICR_DATA =
    BYTE[0] = "0000000000000000000000000000000000000000000000000"
  USER_FLD10 = "0"
-$SMC03-exit
$DATA26 PROXOBJ 7>
```

9- About interchange interface.

Station name	Port No.	IP Address	Port name	Description
S1FGPINHOST	9005	10.24.187.164	$PCLI	**GREENPIN**
S1FHM	5570	10.75.3.36	$FHMV	DCMS
S1FHM1	5570	10.16.12.142	$FHMV	
S1FHM2	5570	10.16.12.41	$FHMV	
S1HCMSFHM1	7790	10.50.37.148	$CFHM1	
S1HCMSFHM3	7992	10.50.37.148	$CFHM3	
S1HCMSFHM2	7791	10.50.37.151	$CFHM2	
S1HCMSFHM4	7993	10.50.37.151	$CFHM4	
S1APHOSTCLI01	1722	10.16.58.51	$PCLI	**CREDIT CARD**
S1FHOSTCLT02	4568	10.50.37.173	$PGRV1	
S1FHOSTCLT01	4568	10.50.37.249	$PGRV1	
S1GBICIST1	5580	10.50.42.139	$ISRV1	IST
S1GBICIST2	5581	10.50.42.139	$ISRV2	
S1GBICIST3	5580	10.50.46.56	$ISRV1	
S1GBICIST4	5581	10.50.46.56	$ISRV2	
S1DNPCS11	7691	10.50.46.157	$PCL23	**PCS**
S1DNPCS12	7691	10.50.46.157	$PCL23	
S1DNPCS13	7696	10.50.46.157	$PCL23	
S1DNPCS14	7696	10.50.46.157	$PCL23	
S1ENPCS11	7692	10.50.46.157	$PCL23	
S1ENPCS12	7692	10.50.46.157	$PCL23	
S1ENPCS13	7697	10.50.46.157	$PCL23	
S1ENPCS14	7697	10.50.46.157	$PCL23	
S1FNPCS11	7693	10.50.46.157	$PCL23	
S1FNPCS12	7693	10.50.46.157	$PCL23	
S1FNPCS13	7698	10.50.46.157	$PCL23	
S1FNPCS14	7698	10.50.46.157	$PCL23	
S1GNPCS11	7694	10.50.46.157	$PCL23	
S1GNPCS12	7694	10.50.46.157	$PCL23	
S1GNPCS13	7699	10.50.46.157	$PCL23	
S1GNPCS14	7699	10.50.46.157	$PCL23	
S1HNPCS11	7695	10.50.46.157	$PCL23	
S1HNPCS12	7695	10.50.46.157	$PCL23	
S1HNPCS13	7700	10.50.46.157	$PCL23	
S1HNPCS14	7700	10.50.46.157	$PCL23	

S1AFLCN01	50051	10.50.77.146	$PCLI	**FOLCAN**
S1AFLCN02	50052	10.50.77.146	$PCLI	
S1BFLCN01	50051	10.50.77.146	$PCLI	
S1BFLCN02	50052	10.50.77.146	$PCLI	
S1CFLCN01	50051	10.50.77.146	$PCLI	
S1CFLCN02	50052	10.50.77.146	$PCLI	
S1DFLCN01	50051	10.50.77.146	$PCLI	
S1DFLCN02	50052	10.50.77.146	$PCLI	
S1EFLCN01	50053	10.50.77.146	$PCLI	
S1EFLCN02	50054	10.50.77.146	$PCLI	
S1FFLCN01	50053	10.50.77.146	$PCLI	
S1FFLCN02	50054	10.50.77.146	$PCLI	
S1GFLCN01	50053	10.50.77.146	$PCLI	
S1GFLCN02	50054	10.50.77.146	$PCLI	
S1HFLCN01	50053	10.50.77.146	$PCLI	
S1HFLCN02	50054	10.50.77.146	$PCLI	
S1BMDS	6004	172.16.118.226	$PCLI	**MASTERCARD**
S1BMDSP	7004	172.16.118.226	$PCLI	
S1ABNET1	7003	172.16.118.226	$PCLI1	
S1EBNET2	6043	172.16.118.227	$PCLI1	
S1EMDS2	8004	172.16.118.227	$PCLI	
S1HBICINI2	9568	172.16.219.1	$PCLI	**NFS ISSUER**
S1HBICINI3	9569	172.16.219.1	$PCLI	
S1HBICINI4	4614	172.16.219.1	$PCLI	
S1HBICINI5	4615	172.16.219.1	$PCLI	
S1EBICINI1	4611	172.16.219.1	$PCLI	
S1EBICINI2	4612	172.16.219.1	$PCLI	
S1EBICINI3	4613	172.16.219.1	$PCLI	
S1EBICINI4	9567	172.16.219.1	$PCLI	
S1ABICINA1	4601	172.16.219.1	$PCLI	NFS ACQUIRER
S1ABICINA2	4602	172.16.219.1	$PCLI	
S1FBICINA3	4603	172.16.219.1	$PCLI	
S1FBICINA4	4604	172.16.219.1	$PCLI	
S1GBICINA5	4605	172.16.219.1	$PCLI	
S1GBICINA6	9764	172.16.219.1	$PCLI	
S1HBICINAT	9527	172.16.219.1	$PCLI	
S1HBICINAT1	9765	172.16.219.1	$PCLI	

S1GBICRPAYI	6188	172.16.219.19	$PCLI	**RUPAY**
S1AELEC	8002	172.16.224.109	$PCLIV	VISA
S1CUSDELEC	8015	172.16.224.109	$PCLI	
S1DBHDELEC	8004	172.16.224.109	$PCLI	
S1AELEC1	1713	172.24.1.5	$PCLI	
S1HPLUSVMT	8022	172.16.224.109	$PCLI	
S1DBHDPLUS	8006	172.16.224.109	$PCLI	
S1AVISA	8001	172.16.224.109	$PCLIV	
S1APLUS1	8007	172.16.224.109	$PCLI	
S1APLUS	8003	172.16.224.109	$PCLI	
S1ABICB	4219	172.24.10.242	$PCLI	BAHRAIN
S1GFDCBIC11	5584	192.168.113.30	$NFDS1	FIRST DATA
S1GFDCBIC12	5585	192.168.113.30	$NFDS2	
S1HFDCBIC11	5590	192.168.113.30	$NFDS3	
S1HFDCBIC12	5591	192.168.113.30	$NFDS4	
S1EURONET	5465	172.16.215.169	$PCLI	EURONET
S1AHI2	7780	10.16.120.230	$PGAF	AHI
S1AHIC2	7777	172.16.18.58	$PGRV	
S1AHIC	7777	172.16.18.6	$PGRV	
S1HAHIAT2	4504	10.16.13.211	$ESRV3	
S1GAHIAT1	4504	10.16.120.230	$ESRV3	
S1FBICIE1	4502	10.16.120.230	$ESRV1	ELECTRA
S1FBICIE2	4503	10.16.120.230	$ESRV2	
S1FBICIE3	4502	10.16.13.211	$ESRV1	
S1FBICIE4	4503	10.16.13.211	$ESRV2	
S1CBATCH	7128	10.16.139.25	$PNAC6	WEB SI
S1FBATCH	7128	172.16.40.179	$PNAC6	

S1HBICIFSS	5666	172.16.237.40	$PFSS	FINAGIRI

FIID	SWI-PRO	STA1
BFDC	P1G^FDCBIC1	S1GFDCBIC11
BFDC	P1H^FDCBIC1	S1HFDCBIC11
BHDE	P1D^BHDELEC	S1DBHDELEC
BHDP	P1D^BHDPLUS	S1DBHDPLUS
BICB	P1A^BICIB	S1ABICB
BICF	P1H^BICIFSS	S1HBICIFSS
BICN	P1A^BICIN	S1ABICN
BICN	P1F^BICIN1	S1ABICN
BICN	P1G^BICIN2	S1ABICN
BICN	P1H^BICINT	S1HBICINAT
BICS	P1E^BICINIS	S1EBICINI1
BICS	P1E^BICINIS2	S1EBICINI2
BICS	P1E^BICINIS3	S1EBICINI3
BICS	P1E^BICINIS4	S1EBICINI4
BICS	P1H^BICINIS2	S1HBICINI2
BICS	P1H^BICINIS3	S1HBICINI3
BICS	P1H^BICINIS4	S1HBICINI4
BICS	P1H^BICINIS5	S1HBICINI5
BNE1	P1E^BNET2	S1EBNET2
BNEM	P1C^BNETM	S1CBNETM
BNET	P1A^BNET	S1ABNET1
BNTP	P1D^BNETPP	S1DBNETPP
BRPI	P1G^BICRPAYI	S1ABICRPAYI
ECOM	P1A^VISA	S1AVISA
ELT	P1F^BICIE	S1FBICIE1
IST	P1G^BICIST	S1GBICIST1
MDS	P1B^MDS	S1BMDS
MDS1	P1E^MDS2	S1EMDS2
MDSP	P1B^MDSP	S1BMDS
MTAP	P1C^BICMTAP	S1CBICMTAP
PLUS	P1A^PLUS	S1APLUS

```
SMS   P1A^PLUS1         S1APLUS1
UK01  P1A^ELEC1         S1AELEC1
USDE  P1C^USDELEC       S1CUSDELEC
VISA  P1A^ELEC          S1AELEC
VISP  P1D^ELECPP        S1DELECPP
VMTP  P1H^PLUSVMT       S1HPLUSVMT
```

10- How to fetch data and read audit log.

Audit Procedure

First find out required station/Process on which need to enable audit, Second find out respective node.

Command for enable audit :-

```
        ALTER S <STATION NAME>,IAU ON                  (input audit)
        ALTER S <STATION NAME>,OAU ON                 (output audit)
        ALTER P <PROCESS NAME>,IAU ON                  (input audit)
        ALTER P < PROCESS NAME>,OAU ON               (output audit)
        ALTER NODE <NODE NAME>,AUDIT ON
```

Command for disable audit :-

```
        ALTER S <STATION NAME>,IAU OFF
        ALTER S <STATION NAME>,OAU OFF
        ALTER P <PROCESS NAME>,IAU OFF
        ALTER P < PROCESS NAME>,OAU OFF
        ALTER NODE <NODE NAME>,AUDIT OFF
```

How to explode audit log:-

1:- Find out audit file path.

```
        INFO NODE <NODE NAME>,AUDINFO
```

Example:-

```
    info node P1D^GATE,audinfo
    NODE: \IMAIN.P1D^GATE
    AUDITA $DATA05.AUDITD.A
    AUDITB $DATA05.AUDITD.B
    AUDITC $DATA05.AUDITD.C
```

2:- Find out which audit file is showing open mode.

Example :-

```
    $DATA05.AUDITD 8> v $DATA05.AUDITD
    $DATA05.AUDITD 9> fi ?

    $DATA05.AUDITD

                  CODE             EOF   LAST MODIFIED  OWNER  RWEP   PExt   SExt
    A             1300       184003356 21SEP2015 21:39 251,255 NNNU    210    210
    B             1300        30419940 29SEP2015  4:38 251,255 NNNU    210    210
    C         O   1300        23943862 29SEP2015 16:18 251,255 NNNU    210    210
$DATA05.AUDITD 10>
```

3:- Explode command.

Go in audit file path and take a incoming log and fire below mentioned command.

```
nexplode c -real -detail -src <source station/process name> -dest <destination station/process name>
-from -1
```

Example:-

```
            V $DATA05.AUDITD
            nexplode a -real -detail -src S1ABICINA1 P1A^BICIN -dest S1ABICINA1 P1A^BICIN -from -1
```

Note:- Nexplode file path - $DATA03.ROSEPRAC

5:- Once required transactions done , after that open log in edit plus and search by card no.

Example of audit log:-

```
R  12696 P1C^PCSHISO2     -> S1CPCS22         Len 355  f--- i 01/04 15:18:33.81
                                                                        59382
ISO8583 message
Header: SoM [ISO] Prod Ind [01] Rel Nr [60] Msg Stat [000] Orig [1] Resp [0]
    Message Type            : 0200 - Financial transaction request
    Primary Bit Map         : F238 8001 A8E0 8018
                      Bits : 1,2,3,4,7,11,12,13,17,32,33,35,37,41,42,43,49,60,
                             61
  1 Secondary Bit Map       : 0000 0000 1400 000C
                      Bits : 100,102,125,126
  2 PAN                     : 4667 3002 3400 2061
  3 Processing Code         : P2 10 00 - (not recognised)
  4 Tran Amount             : 0.00
  7 Transmit Date and Time  : 01/04/yy 09:48:33
 11 Trace Number            : 002326
 12 Local Time              : 15:18:33
 13 Local Date              : 01/04/yy
 17 Capture Date            : 01/04/yy
 32 Acquiring Inst ID       : 50464200000
 33 Forwarding Inst ID      : (length zero)
 35 Track 2                 : 4667300234002061=1708126045
 37 RRN                     : 9788
 41 Card Accptr Term ID     : S1AN2012
 42 Card Accptr ID          : S1AN2012
 43 Card Accptr Name/Locn   : ICICI BANK            MUMBAI       MH IN
 49 Tran Currency Code      : 356 - Indian rupee
 60 Advice Reason Code      : 0000REL6+000
 61 POS Data                : 9998REL61100P
100 Rcv Inst ID             : 5046429997
102 Account ID 1            : 000401140433
125 Reserved (Euro new PIN) : 0200P201019788        Y0000
126 Reserved - Private      : 0200P202019788        Y0000

(6.12)            --- End of Message ---

R  12697 S1CPCS2          -> P1C^PCSHISO2     Len 359  f--- i 01/04 15:18:33.87
                                                                        59383
ISO8583 message
Header: SoM [ISO] Prod Ind [01] Rel Nr [60] Msg Stat [000] Orig [1] Resp [5]
    Message Type            : 0210 - Financial transaction response
    Primary Bit Map         : F238 8001 AAE0 8018
                      Bits : 1,2,3,4,7,11,12,13,17,32,33,35,37,39,41,42,43,49,
                             60,61
  1 Secondary Bit Map       : 0000 0000 1400 000C
                      Bits : 100,102,125,126
  2 PAN                     : 4667 3002 3400 2061
  3 Processing Code         : P2 10 00 - (not recognised)
  4 Tran Amount             : 0.00
  7 Transmit Date and Time  : 01/04/yy 09:48:33
 11 Trace Number            : 002326
 12 Local Time              : 15:18:33
 13 Local Date              : 01/04/yy
 17 Capture Date            : 01/04/yy
 32 Acquiring Inst ID       : 50464200000
 33 Forwarding Inst ID      : (length zero)
 35 Track 2                 : 4667300234002061=1708126045
 37 RRN                     : 9788
 39 Response Code           : 00
 41 Card Accptr Term ID     : S1AN2012
 42 Card Accptr ID          : S1AN2012
 43 Card Accptr Name/Locn   : ICICI BANK            MUMBAI       MH IN
 49 Tran Currency Code      : 356 - Indian rupee
 60 Advice Reason Code      : 0000REL6+000
 61 POS Data                : 9998REL61100P
100 Rcv Inst ID             : 5046429997
102 Account ID 1            : 000401140433
125 Reserved (Euro new PIN) : 0210P201019788        Y000014000
126 Reserved - Private      : 0210P202019788        Y0000140
```

```
(6.12)             --- End of Message ---
```

11- How to pak and unpak for BASE24 file

pak pk*,filename,open,audited,listall

```
unpak PKDBTCH,*.*.*,vol $volume.subvolume,open,audited,listall,myid
```

12- Disk space management.

```
Need to be check mirror disk

$DATA03.PRODREP 2> fup vols

               -- Capacity (Mb) --    %     -- Free Extents --
Volume    (M)     Total       Free     Free     Count    Biggest
$AUDIT     Y    146814   103146.75     70        35    88253.67
$DATA01    Y    146814    35142.85     23     31305    20778.35
$DATA02    Y    146814    74930.53     51       510    74400.41
$DATA03    Y    146814    54276.96     36      6578    44801.15
$DATA04    Y    146814    49005.38     33      9664    27232.01
$DATA05    Y    146814    86750.59     59      3459    85723.69
$DATA06    Y    146814    43771.88     29       261    31784.71
$DATA07    Y    146814    60838.40     41       910    59952.97
$DATA08    Y    146814    41593.74     28       558    38398.85
$DATA09    Y    146814   111613.54     76      4272    56769.21
$DATA10    Y    146814    99976.13     68       592    99739.13
$DATA11    Y    146814    44294.77     30        52    31465.35
$DATA12    Y    146814   113324.68     77       271   112957.12
$DATA13    Y    146814    68893.98     46      3860    64685.75
$DATA14    Y    146814    98539.89     67        33    97646.93
$DATA15    Y    146814    70505.49     48       100    70488.80
$DATA16    Y    146814   112570.56     76       160   112135.21
$DATA17    Y    146814    84482.16     57        11    84453.57
$DATA18    Y    146814    97833.56     66        23    97761.63
$DSMSCM    Y    146814    33397.85     22      3354    29434.79
$EMS1      Y    146814    38357.99     26        11    27786.95
$GGS1      Y    146814    86164.58     58       131    85929.58
$GGS2      Y    146814    98790.17     67        35    98761.34
$OSS       Y    146814    61686.89     42       389    59732.28
$SYSTEM    Y    146814    55974.81     38      1217    47331.27
```

13- How to check queue and how to write

Command for checking queue
status s *,que
status p *,que

Command for checking abnorma
status s *,abn
status p *,abn

Command for write queue
con s <station name>,qwrite $volume.subvolume.q14
con p <process name>,qwrite $volume.subvolume.q14

Command for view queue
con s S10R003471002,qview

14- How to fup dup BASE24 file

fup dup <file name>,$volume.subvolume.*,saveall

15- ATD hanging issue

Need to be check properly ATM configuration in ATD, if ATD file will not allowing access then once stop/start ATD server like below command.

Pathcom--
$DATA01 PRO6CNTL 315> pathcom $npmn
=status server *
=freeze SRV-CCTL-PRO1
=stop SRV-CCTL-PRO1
=thaw SRV-CCTL-PRO1
=start SRV-CCTL-PRO1
=status SRV-CCTL-PRO1
=status SRV-CCTL-PRO1

SERVER #RUNNING ERROR INFO
SRV-CCTL-PRO1 1
=

16- How to monitor transaction throw macro

From both setup (Production and DR) you can monitor.

Take two session first for TLF and second for PTLF

For TLF-
Step 1 - Go in path $data01.firoz
Step 2 - Run SHOWTLF (This is for TLF transaction monitoring)

For PTLF-
Step 1 - Go in path $data01.firoz
Step 2 - Run SHOWPTLF (This is for TLF transaction monitoring)

Note:- During running these program need to check CPU utilization.

17- Interchange Cutover Issue

For Interchange cutover it's depends on interchange response , please check as usual because there is no exact time from ICFE file, till 06:00 am all cutover will happened except BNET.

For BNET check after 11:30 am.

```
$DATA01.FIROZ 5> load intcut                    (command)

Loaded from $DATA01.FIROZ.INTCUT:

INTCUT                                          (command)

$DATA01.FIROZ 6> intcut
Enform^Plus - T0295H01^AAT - (01MAR15)DATE - TIME :  3/10/2016 - 09:42:42
(C) Copyright 2005-2015 Hewlett-Packard Development Company


FIID      SWI-PRO             STA1          POST-DAT
----  ----------------  ----------------  --------

BFDC  P1G^FDCBIC1       S1GFDCBIC11       160310
BFDC  P1H^FDCBIC1       S1HFDCBIC11       160310
BHDE  P1D^BHDELEC       S1DBHDELEC        160310
BHDP  P1D^BHDPLUS       S1DBHDPLUS        160310
BICB  P1A^BICIB         S1ABICB           160310
BICF  P1H^BICIFSS       S1HBICIFSS        160310
BICN  P1A^BICIN         S1ABICN           160310
BICN  P1F^BICIN1        S1ABICN           160310
BICN  P1G^BICIN2        S1ABICN           160310
BICN  P1H^BICINT        S1HBICINAT        160310
BICS  P1E^BICINIS       S1EBICINI1        160310
BICS  P1E^BICINIS2      S1EBICINI2        160310
BICS  P1E^BICINIS3      S1EBICINI3        160310
BICS  P1E^BICINIS4      S1EBICINI4        160310
BICS  P1H^BICINIS2      S1HBICINI2        160310
```

```
BICS  P1H^BICINIS3      S1HBICINI3        160310
BICS  P1H^BICINIS4      S1HBICINI4        160310
BICS  P1H^BICINIS5      S1HBICINI5        160310
BNE1  P1E^BNET2         S1EBNET2          160309
BNEM  P1C^BNETM         S1CBNETM          160309
BNET  P1A^BNET          S1ABNET1          160309
BNTP  P1D^BNETPP        S1DBNETPP         160309
BRPI  P1G^BICRPAYI      S1ABICRPAYI       160310
ECOM  P1A^VISA          S1AVISA           160310
ELT   P1F^BICIE         S1FBICIE1         160310
IST   P1G^BICIST        S1GBICIST1        160310
MDS   P1B^MDS           S1BMDS            160310
MDS1  P1E^MDS2          S1EMDS2           160310
MDSP  P1B^MDSP          S1BMDS            160310
MTAP  P1C^BICMTAP       S1CBICMTAP        160310
PLUS  P1A^PLUS          S1APLUS           160311
SMS   P1A^PLUS1         S1APLUS1          160311
UK01  P1A^ELEC1         S1AELEC1          160310
USDE  P1C^USDELEC       S1CUSDELEC        160310
VISA  P1A^ELEC          S1AELEC           160310
VISP  P1D^ELECPP        S1DELECPP         160310
VMTP  P1H^PLUSVMT       S1HPLUSVMT        160310
```

18-Tandem backup related.

Backup Procedure :-

Step1:- After got confermation for tape loded successfuly.

Step2:- Check in scf

```
$DATA01.REFRESH 8> scf
SCF - T9082H01 - (23JUN11) (02MAY11) - 03/03/2016 11:13:29 System \IMAIN
(C) 1986 Tandem (C) 2006 Hewlett Packard Development Company, L.P.
(Invoking \IMAIN.$DATA03.PRODREP.SCFCSTM)
1-> status $tap*
STORAGE - Status TAPE \IMAIN.$TAPE1
 LDev  State      Primary   Backup    DeviceStatus
                  PID       PID
  175  STARTED     1,348     0,324    ONLINE, BOT           (before initiating obey command)
```

Step3:-

```
$SYSTEM.OPERATE 16> mediacom
MEDIACOM - T6028H01 (31JUL2014)
(C) Copyright 1993-2002, 2004 Hewlett-Packard Development Company, L.P.
MC>status

                    Drive  Tape    Tape      Label    Open
Tape Drive          Status Name    Status    Type     Mode  Process Name
------------------  -----  ------  --------  -------  ----  -----------------
$TAPE1              FREE

1 tape drive returned.
MC>status tapemount

Tape   Tape       Label                    Write
Mount  Name       Type    Node or Device   Prot    Action Needed
-----  ------     ------  ---------------  ----    -------------------------
83     *UNREADABLE TAPE*  $TAPE1                   ACCEPT OR REJECT TAPEMOUNT

1 tape mount returned.
MC>accept tapemount 83

1 tape mount accepted.
MC>status

                    Drive  Tape    Tape      Label    Open
Tape Drive          Status Name    Status    Type     Mode  Process Name
------------------  -----  ------  --------  -------  ----  -----------------
$TAPE1              FREE                     OMITTED                  (before initiating obey command)

1 tape drive returned.
MC>

$DATA01.REFRESH 9> mediacom
MEDIACOM - T6028H01 (31JUL2014)
(C) Copyright 1993-2002, 2004 Hewlett-Packard Development Company, L.P.
MC>status
```

```
                    Drive  Tape    Tape      Label    Open
Tape Drive          Status Name    Status    Type     Mode  Process Name
------------------  -----  ------  --------  -------  ----  -----------------
$TAPE1              INUSE                    OMITTED  NL    \IMAIN.$DLBK   (after initiating obey command)

1 tape drive returned.
MC>
```

Step4:- Check in scf

```
$DATA01.REFRESH 8> scf
SCF - T9082H01 - (23JUN11) (02MAY11) - 03/03/2016 11:13:29 System \IMAIN
(C) 1986 Tandem (C) 2006 Hewlett Packard Development Company, L.P.
(Invoking \IMAIN.$DATA03.PRODREP.SCFCSTM)
2-> status $tap*
STORAGE - Status TAPE \IMAIN.$TAPE1
 LDev  State      Primary   Backup    DeviceStatus
                  PID       PID
  175  STARTED     1,348     0,324    ONLINE              (after initiating obey command)
```

Step5:- Check backup status also from sysbusy--

```
 System \IMAIN                SYSBUSY V2.5                2016/03/03 11:42:16
 00  42%             01                 02                 03  34%
 Process          Pri PFR  Owner  Busy   %  CPU Time Object File
 $OVOCC     3,137  80     255,100 0.784  7  28:23:34 $DATA04.OVNM.ACC
            0,265 255 P   255,255 0.735  7  42:20:33 $SYSTEM.SYS01.TSMSGIP
 $DATA01    0,322 220 P   255,255 0.611  6  21:24:01 $SYSTEM.SYS01.TSYSDP2
 $N1EN      0,535 155     251,255 0.590  5  44:37:56 $DATA03.XPNET.NETWORK
 $TAPE1     1,348 200 P R 255,255 0.581  5  23:50:12 $SYSTEM.SYS01.OTPPROCP
            1,265 255 P   255,255 0.573  5  41:01:13 $SYSTEM.SYS01.TSMSGIP
 $N1AN      0,533 155     251,255 0.557  5  46:49:42 $DATA03.XPNET.NETWORK
 $N1GN      2,556 155     251,255 0.547  5  54:26:12 $DATA03.XPNET.NETWORK
 $N1HN      3,515 155     251,255 0.542  5  54:26:01 $DATA03.XPNET.NETWORK
 $N1BN      1,872 155   R 251,255 0.520  5  46:19:12 $DATA03.XPNET.NETWORK
 $N1FN      1,873 155     251,255 0.501  5  49:01:51 $DATA03.XPNET.NETWORK
 $N1DN      3,514 155     251,255 0.434  4  46:38:11 $DATA03.XPNET.NETWORK
            2,265 255 P   255,255 0.397  3  26:40:58 $SYSTEM.SYS01.TSMSGIP
            1,266 255 P   255,255 0.394  3  28:14:25 $SYSTEM.SYS01.TSMSGIP
 $DLBK      2,192 149 P   251,255 0.393  3  00:00:02 $SYSTEM.SYS01.BACKUP         (required backup)
 $ESM1      0,845 140     255,100 0.391  3  03:40:36 $DATA13.RDREMS.ETLAGENT
 $AOOCC     0,135  85     255,100 0.391  3  01:19:38 $DATA13.OVNM.ACC
 $N1CN      2,555 155     251,255 0.390  3  39:00:59 $DATA03.XPNET.NETWORK
            0,266 255 P   255,255 0.368  3  21:26:35 $SYSTEM.SYS01.TSMSGIP
```

Step6:- Check backup status from spoolcom

```
$SYSTEM.OPERATE 19> spoolcom job (loc #dailbk)

JOB   BATCH STA FLAGS OWNER   TIME  COPY PAGE  REPORT          LOCATION
1286        HLD 4     251,255 02/22 1    313   PROD60  SUPER   #DAILBK
1426        HLD 4     251,255 02/25 1    302   PROD60  SUPER   #DAILBK
1605        HLD 4     251,255 02/27 1    306   PROD60  SUPER   #DAILBK
1653        HLD 4     251,255 02/28 1    307   PROD60  SUPER   #DAILBK
1745        RDY 4     251,255 03/01 1    311   PROD60  SUPER   #DAILBK
1789        RDY 4     251,255 03/02 1    313   PROD60  SUPER   #DAILBK
1808        RDY 4     251,255 10:05 1    2     PROD60  SUPER   #DAILBK
1879        OPN 4     251,255 11:41 1    0     PROD60  SUPER   #DAILBK    (required backup)
3959        HLD 4     251,255 02/23 1    315   PROD60  SUPER   #DAILBK
```

19- EOF checking for all file.

Macro 1 :-
For checking End Of File of all files which are created on daily basis.
Take the log incoming...

$DATA01.FIROZ 148> LOAD EOFCHECK (Command)
Loaded from $DATA01.FIROZ.EOFCHECK:
EOFCHECK
$DATA01.FIROZ 149> EOFCHECK (Command)

Macro 2 :-
For checking End Of File of all files important files.
Take the log incoming...

$DATA01.FIROZ 17> obey INFOEOF (Command)

Example:-

```
$DATA04.PRO6EILF.VE160308            10 Mar 2016, 10:12
    ENSCRIBE
    TYPE E
    FORMAT 1
    EXT ( 1200 PAGES, 1200 PAGES )
    REC 4072
    BLOCK  4096
    ALTKEY ( "PR", FILE 0, KEYOFF 0, KEYLEN 55 )
    ALTFILE ( 0, $DATA06.PRO6EILF.E1160308 )
    MAXEXTENTS 800
    BUFFERED
    OWNER 251,255
    SECURITY (RWEP): GOGO
    DATA MODIF:   9 Mar 2016, 14:17
    CREATION DATE:   7 Mar 2016, 15:30
    LAST OPEN:   9 Mar 2016, 16:00
    FILE LABEL: 2132 (52.4% USED)
    EOF: 1196785664 (60.9% USED)              (need to be check EOF)
    EXTENTS ALLOCATED: 487
$DATA04.PRO6EILF 13>
```

20- ATM configuration

NCS configuration:-

Collect node,ATM-ID,IP, port and DH from concern team and take obeyform from any similar working ATM .
First add line after that station

RESET LINE
SET LINE ERANALYSIS 30
SET LINE PORT $CDMA1
SET LINE CMETHOD ACCEPT
SET LINE FMM ALL
SET LINE CLASS TCPSRV
SET LINE PROTOCOL CTS
ADD LINE L1AN0075, UNDER SYSNAME \IMAIN, UNDER NODE P1A^GATE

RESET STATION
SET STATION DESTINATION P1A^N506
SET STATION ERRORACT REINEVERY
SET STATION REINSTATE 10
SET STATION RADDR 0:10.128.17.42
SET STATION CLASS TCPSRV
SET STATION DEVICE TCP^IP
SET STATION LINENAME \IMAIN.P1A^GATE.L1AN0075
SET STATION USERDATA OFFSET 0 "-0 -K+" 0
SET STATION FMMBROADCAST OFF
ADD STATION S1AN0075, UNDER SYSNAME \IMAIN, UNDER NODE P1A^GATE

ATD configuration--

Collect ATM-ID, DH,AUTH,postal code,branch fiid,device type,location,city state, configuration file and cassette denomination from respective team.

Take any similar working ATM and retrieve , on same atm fill new atm's record and press f3 and go to page 4 and 5 and make sure hopper value will be zero and check config file from page 7 and check digits from page 9, hopper algorithm from page 13 and 14.

```
 BASE24-ATM   ATM TERMINAL DATA       PRO1  0000  16/03/10  10:36  01 OF 22
 TERM ID:  S1AN0075          FIID: 0000  REGION: HYDR  BRANCH: ICIC  LN: PRO1

   DEVICE TYPE:  22 (NCR 5XXX)              NATIVE MESSAGE FORMAT:  0
        PHONE:  TCBIL                   LOG ROUTING CODE:     0
     HARDWARE:  2 (NCR)                DUAL SITE INDICATOR:
   OWNER NAME:  ICICI BANK               LOCATION:  SHOPNO 17 ORANGEBLOCKS
         CITY:  HYDERABAD                 STATE:  TS
      COUNTRY:  IN                   POSTAL CODE:  000500008
    DH PROCESS:  P1A^N506          AUTH PROCESS:  P1A^QUEUE
   INST ID NUM:  50464200000  ALT ROUTING GROUP:  00000000000
                         NCD ROUTING GROUP:  00000000000
 SHARING INFORMATION
```

```
 COUNTY CODE: 356        STATE CODE:  0      COUNTRY CODE:  356
    GROUPS: A                    SURCHARGE PROFILE:

 POSTING DATE:  160310   TIME OFFSET:    0  MULTIPLE ACCT SELECT:  Y  (Y/N)
 MINIMUM DISPENSE AMOUNT:        FROM TIME:     HRS   TO TIME:     HRS

RECORD LAST CHANGED: 15/07/10  10:59   BY USER: 0254 , 00000017   CHANGE
********************************** BASE24 ***********************************
```

21- SPOOLCOM

Command for checking spoolcom space -

From tacl prompt--

spoolcom collect

example -

```
$DATA03.PRODREP 7> spoolcom collect

COLLECT  STATE         FLAGS  CPU    PRI  UNIT  DATA FILE                  %FULL
$S       ACTIVE               1 , 0 145  4     $AUDIT.SPLDATA.DATA        34
$DATA03.PRODREP 8>
```

For particular job status –

```
$SYSTEM.OPERATE 19> spoolcom job (loc #dailbk)

JOB  BATCH STA FLAGS OWNER   TIME  COPY PAGE  REPORT          LOCATION
1286       HLD 4     251,255 02/22 1    313   PROD60  SUPER   #DAILBK
1426       HLD 4     251,255 02/25 1    302   PROD60  SUPER   #DAILBK
1605       HLD 4     251,255 02/27 1    306   PROD60  SUPER   #DAILBK
1653       HLD 4     251,255 02/28 1    307   PROD60  SUPER   #DAILBK
1745       RDY 4     251,255 03/01 1    311   PROD60  SUPER   #DAILBK
1789       RDY 4     251,255 03/02 1    313   PROD60  SUPER   #DAILBK
1808       RDY 4     251,255 10:05 1    2     PROD60  SUPER   #DAILBK
1879       OPN 4     251,255 11:41 1    0     PROD60  SUPER   #DAILBK
3959       HLD 4     251,255 02/23 1    315   PROD60  SUPER   #DAILBK
```

For deletion of particular job –

```
$DATA03.PRODREP 9> peruse;1745;del
$DATA03.PRODREP 10>
```

22- Architecture of Data File, Which Data Disk holds what files

For any particular file like ILF file if we need file formation path , then please check in LNCONF with the help of particular process.

```
5 > status p P1A^ELEC
                                                CURRENT   LOGICAL       QUEUE
   PROCESS                                       STATE     STATE        COUNT
   --------------------------------            --------- ---------  -----------
   \IMAIN.P1F^GATE.P1A^ELEC                     STARTED   STARTED             0


 BASE24-BASE  LOGICAL NET CONFIG FILE  PRO1        16/03/11  12:32  02 OF 04
                          LNCF ASSIGN SCREEN
                     READ BY: P1A^ELEC
                ASSIGN NAME: ILF
                LOCATION/ID: \IMAIN.$DATA04.PRO6EILF.VEYYMMDD
              TEMPLATE FILE: \IMAIN.$DATA04.PRO6TPLT.VEYYMMDD
 USAGE CODES:
 BASE ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____
 ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____
 ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____
 ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____
 ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____
 ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____
 ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____ ____
```

```
 COMMENTS:   ISO HOST INTERFACE'S STORE AND FORWARD FILE NAME. MUST HAVE AN
             ASSOCIATED INTERCHANGE INTERFACE HOST PROCESS NAME. TEMPLATE
             FILE NAME IS REQUIRED. READ BY THE RELEASE 6.0 HOST INTERFACE
             PROCESS.
 USER FIELD:
 RECORD LAST CHANGED: 02/12/19  20:59   BY USER: 0255 , 00000255   CHANGE
 ********************************** BASE24 ***********************************

$DATA04.PRO6EILF

              CODE              EOF   LAST MODIFIED  OWNER  RWEP   PExt   SExt
VE160311  O      0       1015386112 11MAR2016 12:35 251,255 GOGO   1200   1200
$DATA04.PRO6EILF 8>
```

Few file path:-

```
config-->>
$DATA01.BAHCNFG.BDPNATME
$DATA01.SRILANKA.MDIPDBCD
$DATA04.IATMCNFG.CFDSBI3
$DATA04.IATMCNFG.CFDSMI3
$DATA04.IATMCNFG.CFNSBI1
$DATA04.IATMCNFG.CFNSBI11
$DATA04.IATMCNFG.CFNSBI3
$DATA04.IATMCNFG.CFNSBI33
$DATA04.IATMCNFG.CFNSBT1
$DATA04.IATMCNFG.CFNSMI1
$DATA04.IATMCNFG.CFNSMI11
$DATA04.IATMCNFG.CFNSMI3
$DATA04.IATMCNFG.CFNSMI33
$DATA04.IATMCNFG.CFNSMT1
$DATA04.IATMCNFG.DBNACAM
$DATA04.IATMCNFG.DBNACFG
$DATA04.IATMCNFG.DBNACFG1
$DATA04.IATMCNFG.HBNACFG
$DATA04.IATMCNFG.HBNACFG2
$DATA04.IATMCNFG.NBNACAM
$DATA04.IATMCNFG.NBNACFG
$DATA04.IATMCNFG.NBNACFG1
$DATA04.IATMCNFG.WBNACFG
$DATA04.JATMCNFG.CFNSBT1
$DATA04.JATMCNFG.CFNSMT1
$DATA04.NATMCNFG.CFNSBI1
$DATA04.NATMCNFG.CFNSBI3
$DATA04.NATMCNFG.CFNSMI1
$DATA04.NATMCNFG.CFNSMI3
$DATA04.SIATMCNF.CFDSBI3
$DATA04.SIATMCNF.CFDSMI3
$DATA04.SIATMCNF.CFDSMT3
$DATA04.WBNANEW.CFNSMI1
$DATA09.PRODGPIN.GCFDSMI3
$DATA09.PRODGPIN.GCFNSMI1

LNCF-->>
$DATA03.PR60DATA.L1CONF

Precmd file-->>
$DATA03.PRO6CNTL.PRECMD

Sproute file-->>
$DATA03.PR60DATA.SPROUTE
$DATA03.PR60DATA.SPROUTEO

NCS Security Files-->>
$DATA03.PR60DATA.NCSP
$DATA03.PR60DATA.NCSS
$DATA03.PR60DATA.NCSS0

IPF file-->>
$DATA03.PRO6DATA.AFTIPF
$DATA03.PRO6DATA.AMEXIPF
$DATA03.PRO6DATA.CHK2IPF
$DATA03.PRO6DATA.CNSLIPF
$DATA03.PRO6DATA.ILNKIPF
```

$DATA03.PRO6DATA.IPF
$DATA03.PRO6DATA.IPFBICB
$DATA03.PRO6DATA.IPFBICI
$DATA03.PRO6DATA.IPFBICN
$DATA03.PRO6DATA.IPFBICR
$DATA03.PRO6DATA.IPFCIR
$DATA03.PRO6DATA.IPFDEL
$DATA03.PRO6DATA.IPFMAE
$DATA03.PRO6DATA.IPFMAS
$DATA03.PRO6DATA.IPFMDS
$DATA03.PRO6DATA.PLUSIPF
$DATA03.PRO6DATA.PSYSIPF
$DATA03.PRO6DATA.VCDBIPF
$DATA03.PRO6DATA.VISAIPF
$DATA03.PRO6DATA.VISPIPF

EMT file -->>
$DATA07.PRO6DATA.APCFEMT
$DATA07.PRO6DATA.DIDFEMT
$DATA07.PRO6DATA.ERFEMT
$DATA07.PRO6DATA.IDFEMT
$DATA07.PRO6DATA.IPCFEMT
$DATA07.PRO6DATA.OAPCFEMT
$DATA07.PRO6DATA.ODIDFEMT
$DATA07.PRO6DATA.OERFEMT
$DATA07.PRO6DATA.OIDFEMT
$DATA07.PRO6DATA.OIPCFEMT
$DATA07.PRO6DATA.ORCDFEMT
$DATA07.PRO6DATA.ORCPTEMT
$DATA07.PRO6DATA.OSTRFEMT
$DATA07.PRO6DATA.OTRFEMT
$DATA07.PRO6DATA.RCDFEMT
$DATA07.PRO6DATA.RCPTEMT
$DATA07.PRO6DATA.STRFEMT
$DATA07.PRO6DATA.TRFEMT

POBJ files-->>
$DATA03.PRO6OBJ.POBJCOD
$DATA03.PRO6OBJ.POBJDIR
$DATA03.PRO6OBJ.POBJSYM

Others file-->>
$DATA09.PRO6DATA.VBIN
$DATA02.CRSDATA.TBRF
$DATA03.PRO6DATA.ECOMFILE
$DATA03.PRO6DATA.MCCFILE
$DATA09.DEBIFILE.BMCCFILE
$DATA07.PRO6DATA.LAF
$DATA12.PRO6DATA.IBIN

Object file-->>
$DATA09.PRO6OBJ.AEGN
$DATA09.PRO6OBJ.AHI
$DATA09.PRO6OBJ.AHIAUTH
$DATA09.PRO6OBJ.ALTRTAU
$DATA09.PRO6OBJ.ARCTHISO
$DATA09.PRO6OBJ.ATSETL
$DATA09.PRO6OBJ.AUTH1
$DATA09.PRO6OBJ.AUTHFSS
$DATA09.PRO6OBJ.AUTHLIAS
$DATA09.PRO6OBJ.AUTHLIBS
$DATA09.PRO6OBJ.AUTHP
$DATA09.PRO6OBJ.AUTHP1
$DATA09.PRO6OBJ.AUTHQ
$DATA09.PRO6OBJ.AUTHQ1
$DATA09.PRO6OBJ.AUTHS
$DATA09.PRO6OBJ.AUTHT
$DATA09.PRO6OBJ.AUTHT1
$DATA09.PRO6OBJ.AUTHT2
$DATA09.PRO6OBJ.AUTHT3
$DATA09.PRO6OBJ.AUTHT4
$DATA09.PRO6OBJ.AUTHUK
$DATA09.PRO6OBJ.AUTHUK1

$DATA09.PRO6OBJ.BABICIB
$DATA09.PRO6OBJ.BABICIN
$DATA09.PRO6OBJ.BABICIN1
$DATA09.PRO6OBJ.BABICINT
$DATA09.PRO6OBJ.BICIE
$DATA09.PRO6OBJ.BICIFSS
$DATA09.PRO6OBJ.BICIFSS1
$DATA09.PRO6OBJ.BICIST
$DATA09.PRO6OBJ.BNET
$DATA09.PRO6OBJ.BNET1
$DATA09.PRO6OBJ.C10001
$DATA09.PRO6OBJ.C1000B1
$DATA09.PRO6OBJ.C1000B2
$DATA09.PRO6OBJ.C1000BN
$DATA09.PRO6OBJ.C1000BNA
$DATA09.PRO6OBJ.C1000Q
$DATA09.PRO6OBJ.C1000T
$DATA09.PRO6OBJ.C1000T1
$DATA09.PRO6OBJ.C1000T15
$DATA09.PRO6OBJ.C1000T2
$DATA09.PRO6OBJ.C100BNAO
$DATA09.PRO6OBJ.C100BNAT
$DATA09.PRO6OBJ.C100BNAV
$DATA09.PRO6OBJ.CMFHAUTH
$DATA09.PRO6OBJ.CTMP0001
$DATA09.PRO6OBJ.EMUKREFR
$DATA09.PRO6OBJ.EUROHISO
$DATA09.PRO6OBJ.EXTR
$DATA09.PRO6OBJ.FDCB0610
$DATA09.PRO6OBJ.FDCBICI
$DATA09.PRO6OBJ.FDCBICI1
$DATA09.PRO6OBJ.FDCRTAU
$DATA09.PRO6OBJ.FDCRTAU1
$DATA09.PRO6OBJ.FDCRTAU2
$DATA09.PRO6OBJ.FDCRTAU3
$DATA09.PRO6OBJ.FHCMHISO
$DATA09.PRO6OBJ.FHMA1013
$DATA09.PRO6OBJ.FHMAUTH
$DATA09.PRO6OBJ.FHMHISO
$DATA09.PRO6OBJ.FILE1
$DATA09.PRO6OBJ.GPINHISO
$DATA09.PRO6OBJ.HISOP
$DATA09.PRO6OBJ.HISOQ
$DATA09.PRO6OBJ.HISOQ1
$DATA09.PRO6OBJ.HPDH
$DATA09.PRO6OBJ.HPDHRPI
$DATA09.PRO6OBJ.MDS
$DATA09.PRO6OBJ.MYFILE
$DATA09.PRO6OBJ.N501
$DATA09.PRO6OBJ.N50BH
$DATA09.PRO6OBJ.N50BNAV
$DATA09.PRO6OBJ.N50HBNA
$DATA09.PRO6OBJ.N50HBNAO
$DATA09.PRO6OBJ.N50HBNAT
$DATA09.PRO6OBJ.N50NBNA
$DATA09.PRO6OBJ.N50NBNA1
$DATA09.PRO6OBJ.N50NBNAO
$DATA09.PRO6OBJ.N50NBNAT
$DATA09.PRO6OBJ.N50Q
$DATA09.PRO6OBJ.N50T
$DATA09.PRO6OBJ.N50T1
$DATA09.PRO6OBJ.N50T2
$DATA09.PRO6OBJ.N50WBNA
$DATA09.PRO6OBJ.N50WBNAO
$DATA09.PRO6OBJ.N50WVOCO
$DATA09.PRO6OBJ.N50WVOCT
$DATA09.PRO6OBJ.NBNACFG
$DATA09.PRO6OBJ.NPCSHISO
$DATA09.PRO6OBJ.NREFR
$DATA09.PRO6OBJ.OAHI
$DATA09.PRO6OBJ.OAUTHT
$DATA09.PRO6OBJ.OBABICIN
$DATA09.PRO6OBJ.OC1000T
$DATA09.PRO6OBJ.OC100BNA

$DATA09.PRO6OBJ.OEMUKREF
$DATA09.PRO6OBJ.ON50HBNA
$DATA09.PRO6OBJ.ON50HBNT
$DATA09.PRO6OBJ.OOREFR
$DATA09.PRO6OBJ.OREFR
$DATA09.PRO6OBJ.OUK01REF
$DATA09.PRO6OBJ.PCSHISO
$DATA09.PRO6OBJ.PGHISO
$DATA09.PRO6OBJ.PSETL
$DATA09.PRO6OBJ.RPBICI
$DATA09.PRO6OBJ.RTAU
$DATA09.PRO6OBJ.RTAU1
$DATA09.PRO6OBJ.RTAU2
$DATA09.PRO6OBJ.RTAU3
$DATA09.PRO6OBJ.RTAU4
$DATA09.PRO6OBJ.RTAUP
$DATA09.PRO6OBJ.UK01REFR
$DATA09.PRO6OBJ.UKRTAU
$DATA09.PRO6OBJ.VANHISO
$DATA09.PRO6OBJ.VISA
$DATA09.PRO6OBJ.VISAP

TSS OLTP Files--->>
$DATA07.TS11DATA.ASAFOD
$DATA07.TS11DATA.BNTH2OD
$DATA07.TS11DATA.CAOD
$DATA07.TS11DATA.CRDVOD
$DATA07.TS11DATA.CVROD
$DATA07.TS11DATA.CVROOD
$DATA07.TS11DATA.DCRDOD
$DATA07.TS11DATA.DNTOD
$DATA07.TS11DATA.DPVNOD
$DATA07.TS11DATA.DRVKOD
$DATA07.TS11DATA.EIADOD
$DATA07.TS11DATA.EMVSOD
$DATA07.TS11DATA.EVTOD
$DATA07.TS11DATA.FPFLOD
$DATA07.TS11DATA.FPLSOD
$DATA07.TS11DATA.IDESOD
$DATA07.TS11DATA.IDNTOD
$DATA07.TS11DATA.ISECOD
$DATA07.TS11DATA.NCROD
$DATA07.TS11DATA.OBNTH2OD
$DATA07.TS11DATA.OCAOD
$DATA07.TS11DATA.OCRDVOD
$DATA07.TS11DATA.OCVROD
$DATA07.TS11DATA.OCVROOD
$DATA07.TS11DATA.ODCRDOD
$DATA07.TS11DATA.ODNTOD
$DATA07.TS11DATA.ODPVNOD
$DATA07.TS11DATA.ODRVKOD
$DATA07.TS11DATA.OEIADOD
$DATA07.TS11DATA.OEMVSOD
$DATA07.TS11DATA.OEVTOD
$DATA07.TS11DATA.OFPFLOD
$DATA07.TS11DATA.OFPLSOD
$DATA07.TS11DATA.OIDESOD
$DATA07.TS11DATA.OIDNTOD
$DATA07.TS11DATA.OISECOD
$DATA07.TS11DATA.ONCROD
$DATA07.TS11DATA.OROCOD
$DATA07.TS11DATA.OSCPTOD
$DATA07.TS11DATA.OTRNSOD
$DATA07.TS11DATA.OTVROD
$DATA07.TS11DATA.OTVROOD
$DATA07.TS11DATA.OVPVVOD
$DATA07.TS11DATA.ROCOD
$DATA07.TS11DATA.SCPTOD
$DATA07.TS11DATA.TRNSOD
$DATA07.TS11DATA.TVROD
$DATA07.TS11DATA.TVROOD
$DATA07.TS11DATA.UJSFOD
$DATA07.TS11DATA.USAFOD
$DATA07.TS11DATA.VPVVOD

```
TSS Objects-->>
$DATA07.TS11OBJ.APINITO
$DATA07.TS11OBJ.BUADTSLO
$DATA07.TS11OBJ.BUDENCLO
$DATA07.TS11OBJ.BUEMVLO
$DATA07.TS11OBJ.BUERTELO
$DATA07.TS11OBJ.BUMCLO
$DATA07.TS11OBJ.BUSCPTLO
$DATA07.TS11OBJ.BUTSECLO
$DATA07.TS11OBJ.BUTSSPLO
$DATA07.TS11OBJ.BUTXNLO
$DATA07.TS11OBJ.BUXMLILO
$DATA07.TS11OBJ.CIDTEBO
$DATA07.TS11OBJ.CIDTFIO
$DATA07.TS11OBJ.CIDTFO
$DATA07.TS11OBJ.CIDTIO
$DATA07.TS11OBJ.CIDTO
$DATA07.TS11OBJ.CISEPO
$DATA07.TS11OBJ.DALCILO
$DATA07.TS11OBJ.FNGNDSLO
$DATA07.TS11OBJ.FNHEAPLO
$DATA07.TS11OBJ.FNLO
$DATA07.TS11OBJ.FNQLO
$DATA07.TS11OBJ.HKDCLO
$DATA07.TS11OBJ.MDQO
$DATA07.TS11OBJ.METAMNLO
$DATA07.TS11OBJ.SAXLIBLO
$DATA07.TS11OBJ.SCPTCCLO
$DATA07.TS11OBJ.SDCCLO
$DATA07.TS11OBJ.SIMAINLO
$DATA07.TS11OBJ.SISLO
$DATA07.TS11OBJ.TRCVWLO
$DATA07.TS11OBJ.TSAMBLO
$DATA07.TS11OBJ.TSATALLO
$DATA07.TS11OBJ.TSBNTLO
$DATA07.TS11OBJ.TSDEPLO
$DATA07.TS11OBJ.TSECOMLO
$DATA07.TS11OBJ.TSSTDELO
$DATA07.TS11OBJ.TSTHALLO
$DATA07.TS11OBJ.UIAKDSLO
$DATA07.TS11OBJ.UIAKEYLO
$DATA07.TS11OBJ.UIELELO
$DATA07.TS11OBJ.UIISECLO
$DATA07.TS11OBJ.UIKEYCLO
$DATA07.TS11OBJ.UIKEYFLO
$DATA07.TS11OBJ.UILIBSLO
$DATA07.TS11OBJ.UIPCTLLO
$DATA07.TS11OBJ.UISECLO
$DATA07.TS11OBJ.UITSECLO
$DATA07.TS11OBJ.UTCILO
$DATA07.TS11OBJ.UTCMDLO
```

23- VISA and Master card BIN upload overview and importance of Sproute table

VISA BIN REFRESH PROCEDURE

==========

From production--

Step 1-

Take bakup of below file in path - $data03.ipfbkp

Master Card IPF File Location: $DATA03.PRO6DATA.IPFMAS

VisaIPF file location: $data03.pro6data.visaipf

fup dup IPFMAS,$data03.ipfbkp.*,saveall

fup dup visaipf,$data03.ipfbkp.*,saveall

Step2-

From production--

$DATA03.BINREFR

CODE EOF LAST MODIFIED OWNER RWEP PExt SExt

FRI 0 10481520 03FEB2015 16:01 251,255 GOGO 28 28

THU 0 14254670 03FEB2015 16:01 251,255 GOGO 28 28

TT458030 0 5522000 03FEB2015 16:01 251,255 GOGO 14 28

$DATA03.BINREFR 14>

$DATA03.BINREFR 14> pak pkfri03,fri,open,audited,listall
$DATA03.BINREFR 17> pak pkthu03,thu,open,audited,listall

Step3-
From DR--
On DR $DATA01.TESTCNTL check and rename fri and thu file.
File transfer on DR in path - $DATA01.TESTCNTL
Step4-
From production--
$DATA03.BINREFR 22> **ftp 10.75.9.13**
Name (10.75.9.13:user): base24reps
Password:*******
ftp> cd $DATA01.TESTCNTL
ftp> pro
ftp> has
ftp> bin
ftp> put pkfri03
ftp> put pkthu03
ftp> exit

Step5-
From DR--
Unpak all 2 files.
$DATA01.TESTCNTL 32> unpak PKFRI03,*.*.*,vol $data01.testcntl,open,audited,listall
$DATA01.TESTCNTL 37> unpak PKTHU03,*.*.*,vol $data01.testcntl,open,audited,listall
check all 2 file size from both setup (production and DR)

Step 6-
From DR--

```
$DATA01.TESTCNTL 45> fup copy THU,,a,count 4
$DATA01.TESTCNTL.THU RECORD 0 KEY 0 (%0) LEN 132 6FEB15 12:44
0: 3300466705400082TRSVISABIN00055573HEADER BINDISTR 15026TOTAL
VIS (Check VISABIN)
35: A
$DATA01.TESTCNTL.THU RECORD 1 KEY 132 (%204) LEN 132
0: VISA TABLE
0..3300466705400082TRSVISABIN000555
35: 74000001111094012123460008400000002111094012123 47
$DATA01.TESTCNTL.THU RECORD 2 KEY 264 (%410) LEN 132
0: 000840 00000311109401212348000840 VISA
TABLE
35: 0..3300466705400082TRSVISABIN000555750000041110940121234900
$DATA01.TESTCNTL.THU RECORD 3 KEY 396 (%614) LEN 132
0: 0840 00000511209401212346000840
000006112094
35: 01212347000840 VISA TABLE 0..3300466705400082TR
4 RECORDS TRANSFERRED
$DATA01.TESTCNTL 46>
$DATA01.TESTCNTL 47> fup copy FRI,,a,count 4
$DATA01.TESTCNTL.FRI RECORD 0 KEY 0 (%0) LEN 132 6FEB15 12:45
0: 3300466705400081DSPLUSBIN 00054649TAPEHEADER BINDISTR 15029TOTAL
PLP (Check PLUSBIN)
35: LUS BIN TABLE
$DATA01.TESTCNTL.FRI RECORD 1 KEY 132 (%204) LEN 132
0: 0..3300466705400081DSPLUSBIN
000546
35: 5000000111312450653321380NAS 196PLUSPLUS BIN TABLE
$DATA01.TESTCNTL.FRI RECORD 2 KEY 264 (%410) LEN 132
0:
35: 0..3300466705400081DSPLUSBIN 000546510000021131245065332138
$DATA01.TESTCNTL.FRI RECORD 3 KEY 396 (%614) LEN 132
0: 1NAS 196PLUSPLUS BIN TABLE
35: 0..3300466705400081DS
4 RECORDS TRANSFERRED
$DATA01.TESTCNTL 48>
```

Step 7-
From DR--
VISA File Packing Procedure:
$DATA01.TESTREFR 57> FI TAPE
$DATA01.TESTREFR 58> FUP PURGEDATA TAPE
$DATA01.TESTREFR 59> FI TAPE
$DATA01.TESTREFR 60> FUP COPY $DATA01.TESTCNTL.THU, TAPE, RECIN 170,RECOUT 168
$DATA01.TESTREFR 61> FI TAPE

$DATA01.TESTREFR 62> O RUNVISAR
$DATA01.ANIRUDDH 2> SPOOLCOM JOB (LOC #IPFOUT)
$DATA01.ANIRUDDH 4> PERUSE;2138;LL (check with proper job no.)

Step 08-
From DR--
$DATA01.TESTREFR 63> FI VPAK
$DATA01.TESTREFR 39> #PURGE VPAK
$DATA01.TESTREFR 65> FI VPAK
$DATA01.TESTREFR 66> PAK VPAK, TAPE
$DATA01.TESTREFR 67> FI VPAK

Step 09-
From DR--
PLUS File Packing Procedure:-
$DATA01.TESTREFR 72> FI TAPE
$DATA01.TESTREFR 73> FUP PURGEDATA TAPE
$DATA01.TESTREFR 74> FI TAPE
$DATA01.TESTREFR 75> FUP COPY $DATA01.TESTCNTL.FRI, TAPE, RECIN 170,RECOUT 168
$DATA01.TESTREFR 77> FI TAPE
$DATA01.TESTREFR 78> O RUNVISAR
$DATA01.TESTREFR 80> SPOOLCOM JOB (LOC #IPFOUT)
$DATA01.TESTREFR 82> PERUSE;2141;LL

Step 11-
From DR--
$DATA01.TESTREFR 84> FI PPAK
$DATA01.TESTREFR 52> #PURGE PPAK
$DATA01.TESTREFR 53> FI PPAK
$DATA01.TESTREFR 87> PAK PPAK, TAPE
$DATA01.TESTREFR 88> FI PPAK

Step 12-
From production--
$DATA01.SRP 35> FI VPAK
$DATA01.SRP 36> #PURGE VPAK
$DATA01.SRP 37> FI VPAK
$DATA01.SRP 40> FI PPAK
$DATA01.SRP 41> #PURGE PPAK
$DATA01.SRP 42> FI PPAK

Step 13-
From production--
$DATA01.SRP 44> ftp 10.75.9.13 (DR IP)
Name (10.75.9.13:user): base24reps
Password:********
ftp> pro
ftp> has
ftp> bin
ftp> cd $DATA01.TESTREFR
ftp> dir VPAK
ftp> get VPAK
ftp> dir PPAK
ftp> get PPAK
ftp> bye

Step 14-
From production--
IMP:Please check the DR and production VPAK & PPAK files it should be same.
Check Procedure:
ON DR: ($DATA01.TESTREFR)
fi vpak
fi ppak
On Production: ($DATA01.SRP)
fi vpak
fi ppak

Step 15-
From production--
$DATA03.PRO6DATA 51> FI TAPE
$DATA03.PRO6DATA 52> #purge TAPE
$DATA03.PRO6DATA 53> FI TAPE
$DATA01.SRP 56> UNPAK VPAK, *.*.*,myid, VOL $DATA03.PRO6DATA

$DATA03.PRO6DATA 58> FI TAPE
$DATA03.PRO6DATA 61>O RUNVISAR
$DATA03.PRO6DATA 62> SPOOLCOM JOB (LOC #IPFOUT)
$DATA03.PRO6DATA 63> PERUSE;3335;LL

Step 16-
From production--
$DATA01.SRP 70> FI PPAK
$DATA01.SRP 71> UNPAK PPAK, *.*.*,myid, VOL $DATA03.PRO6DATA
$DATA03.PRO6DATA 73> FI TAPE
$DATA03.PRO6DATA 75> O RUNVISAR
$DATA03.PRO6DATA 79> SPOOLCOM JOB (LOC #IPFOUT)
$DATA03.PRO6DATA 80> PERUSE;3336;LL
MDS Bin refresh

==========

From Production---
Step 17-
$DATA03.BINREFR 66> fi TT458058
$DATA03.BINREFR 67>fup dup TT458058,$DATA03.MDSNREFR.*,saveall
$DATA03.MDSNREFR 98> fup rename TT458058,TT010300 (TTddmm00)
$DATA03.MDSNREFR 98> RUN LOAD
INPUT FILE NAME :TT010300
$DATA03.MDSNREFR 99> fi TT010300
$DATA03.MDSNREFR 100> FI CIRRUS
$DATA03.MDSNREFR 102> FUP COPY CIRRUS ,,A,COUNT 2 (check file header)
$DATA03.MDSNREFR 104> O RUNCIRR
$DATA03.MDSNREFR 106> SPOOLCOM JOB (LOC #IPFCIRR)
$DATA03.MDSNREFR 107> PERUSE;3339;LL

Step 18-
Note: Before RUN the GOPRE Command ensure that in volume $data03.pr60data only Sproute should be open. If any other sproute is open then give warm boot all the process and check that sproute is open and all rest sproute is closed.
Step 19-
V $data03.pr60data
FI *SP*
$DATA03.PRO6CNTL > FI GOPRE
O GOPRE (By using this command we change the data present in permanent memory as we carried out the changes in the data (as we refreshed the files))
SPOOLCOM JOB (LOC #PRE1A)
Please check the sproute file.It should be closed and SprouteO file should be opened.
V $data03.pr60data
FI *SP*
Step 20--
Warmboot is given by the login of NCS ...
Warm boot is given to keep the current sproute file open and to close the existing sproute file.Check status for queue and abnormal and give warmboot to each process.
Please put a ems filter for word “Warmboot”
EMS filter
pass when text="Warmboot"
Warmboot is given to change the data present in the buffered memory as by using obey gopre we changed the data present in permanent memory so that whenever any transaction occur it will fetch the data from the buffered memory .
NCS
control node P1A^GATE, WARMBOOTSPROUTE
control node P1B^GATE, WARMBOOTSPROUTE
control node P1C^GATE, WARMBOOTSPROUTE
control node P1D^GATE, WARMBOOTSPROUTE
control node P1E^GATE, WARMBOOTSPROUTE
control node P1F^GATE, WARMBOOTSPROUTE
control node P1G^GATE, WARMBOOTSPROUTE
control node P1H^GATE, WARMBOOTSPROUTE
Please check any queue and abnormality on the processes by the following commands in NCS)
STATUS P *,QUEUE (To check Queue)
STATUS P *,ABNORMAL (To check Abnormal)
After giving WARMBOOTSPROUTE to each node please check the open sproute file.
V $data03.pr60data
FI *SP*

For more details and remaining points please refer ACI doc.