

# **HALDB Must Have When Database Size is the Issue**

Feb 27, 2007

**Christian Koeppen**

**Ron Bisceglia**



# Agenda

- How to convert to HALDB
  - ▶ Manually
  - ▶ Using a tool
  - ▶ Database is 24\*7
    - ORF
    - Convert online
- How to maintain partitions
  - ▶ What will happen to our data
  - ▶ Maintain manually
  - ▶ Using a tool
  - ▶ Database is 24\*7
    - Maintain online
- Additional tools



# Conversion problem

- Not an easy task
  - ▶ Need experienced DBA
  - ▶ Time consuming
- Database must be offline
  - ▶ The largest databases are 24\*7
- Complex conversion process
  - ▶ New JCL and procedures
  - ▶ Partition definition
  - ▶ Dataset names
- Fallback



# How to Convert to HALDB

- Manually
  - ▶ Find the high keys
    - Remember them for later
  - ▶ Change and stage the DBD
    - Convert primary DBD to HALDB
    - Convert secondary indexes to HALDB
      - /SX is now 8 bytes
      - Check if applications are using the index as a database
    - Does the secondary index need more than 1 partition?
      - Index record size increases (+28, +root key length)
  - ▶ Create IDCAMS statements for all partitions
    - Remember the new dataset naming rules



# How to Convert to HALDB

- Manually (cont.)
  - ▶ Prepare the partitions
    - Create DBRC statements
    - Use the ISPF based program
      - Use a temporary recon and “EXPORT”
  - ▶ Take the database offline
  - ▶ Run UNLOAD
    - Using MIGRATX option
      - Creates multiple unload datasets
  - ▶ Delete the database (and its indexes) from DBRC
  - ▶ Activate the changed DBDs
  - ▶ Define the partitions



# How to Convert to HALDB

- Manually (cont.)
  - ▶ Run allocation for the new datasets
  - ▶ Run partition initialization
    - Make sure that it runs on all partitions when redoing the allocation
  - ▶ Run RELOAD for the primary database
  - ▶ Run RELOAD for the index databases
    - Will need to be sorted
    - Sort parameters provided during unload
  - ▶ Do the necessary image copies
    - ILDS and primary index do not have image copies
  - ▶ Run ACBGEN
  - ▶ Do online change.
    - May include some application changes



# How to Convert to HALDB

- Using a Tool
  - ▶ ISPF based application
    - Creates all necessary JCL and control cards
    - Creates the new DBDs
    - Creates the partition definition
    - Created the IDCAMS allocation statements
    - User submits the JCL
      - Next phase is created when previous phase was OK
      - JCL can be recreated if
    - Conversion options provided via ISPF panels



# How to Convert to HALDB

The screenshot shows a terminal window titled "SmarTerm - [To Ron]" with a menu bar (File, Edit, View, Tools, Properties, Connection, Window, Help) and a toolbar. The main content area displays the following configuration for "PECK HALDB Tools":

```

----- PECK HALDB Tools -----
Command ==>

Primary DBD Partitioning Rules      for display only

Partitions:      1      1  Fixed Number of Partitions      2_
                  2      2  Fixed Partition Size (MB)      2048
                  3      3  Ask during Conversion
                  4      4  Specify High Keys

PDB Conversion:  2      1  Use existing Keys or Partitions
                  2      2  Create new Partition boundaries
                  3      3  Ask during Conversion

DEDB Conversion: 1      1  Same Number of Partitions
                  2      2  New Partition boundaries

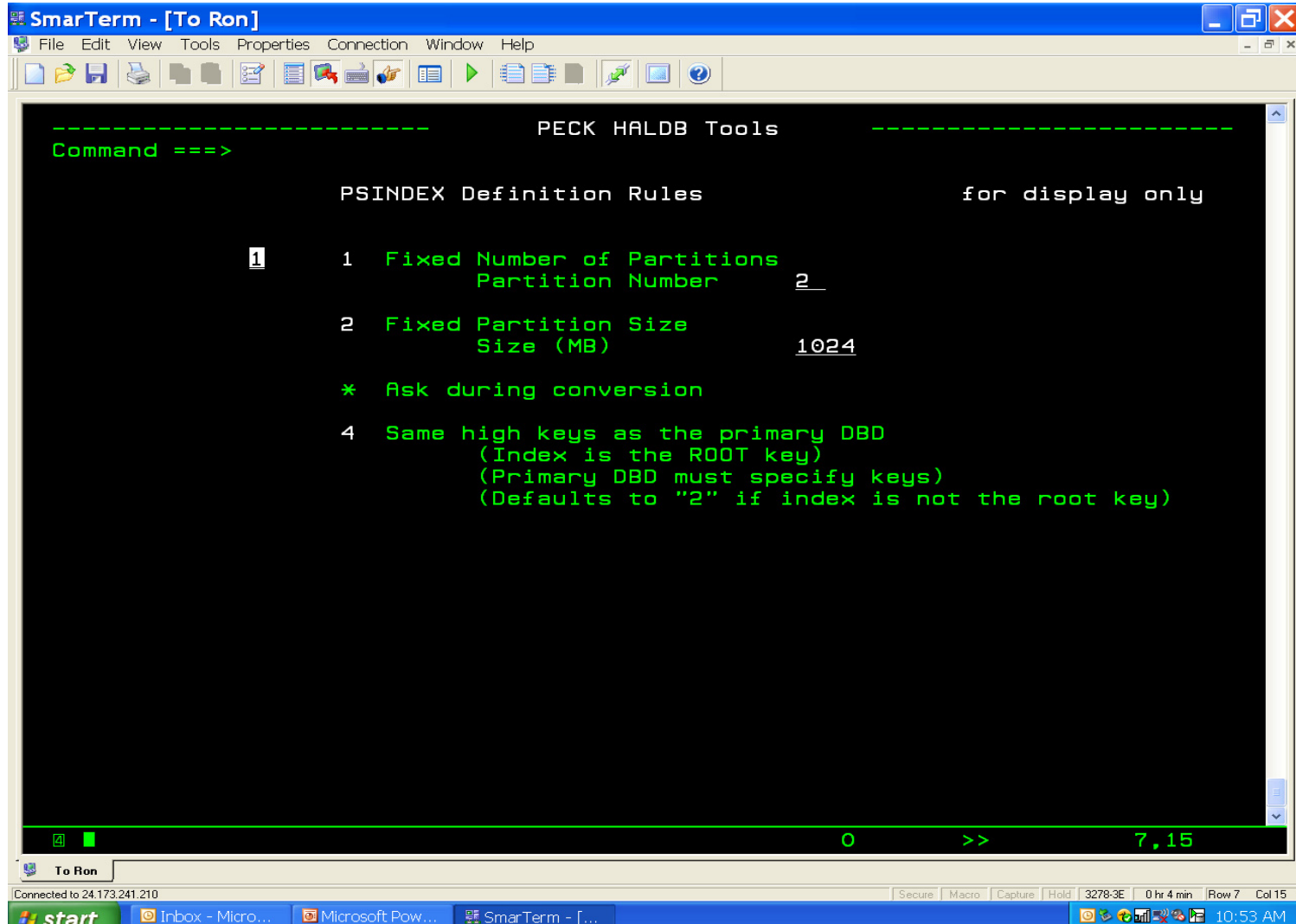
Additional Partitioning Layouts to be created?
                  2      1  Yes
                  2      2  No, use the one created during Collect

Combine Database records:
                  5000 (number of database records)
    
```

At the bottom of the terminal window, there is a status bar showing "0 >> 8,23". The taskbar at the bottom of the screen shows the Windows Start button, several open applications (Inbox - Micro..., Microsoft Pow..., SmarTerm - [...]), and the system tray with the time "10:51 AM".



# How to Convert to HALDB



The screenshot shows a terminal window titled "SmarTerm - [To Ron]" with a menu bar (File, Edit, View, Tools, Properties, Connection, Window, Help) and a toolbar. The terminal content displays the output of the "PECK HALDB Tools" command. The output is as follows:

```
----- PECK HALDB Tools -----  
Command ==>  
  
PSINDEX Definition Rules for display only  
  
1 1 Fixed Number of Partitions  
   Partition Number 2  
  
2 Fixed Partition Size  
   Size (MB) 1024  
  
* Ask during conversion  
  
4 Same high keys as the primary DBD  
   (Index is the ROOT key)  
   (Primary DBD must specify keys)  
   (Defaults to "2" if index is not the root key)
```

At the bottom of the terminal window, there is a status bar showing "0" on the left, ">>" in the center, and "7,15" on the right. Below the terminal window, the Windows taskbar is visible, showing the "start" button and several open applications: "Inbox - Micro...", "Microsoft Pow...", and "SmarTerm - [...]". The system tray on the right shows the date and time as "10:53 AM".

# How to Convert to HALDB

- ▶ Using JCL and Control Cards
  - Control card example  
    CONVERT DBD(dbdname) PARTSIZE(2048) –  
        DBDPATT(\*\*\*\*\* ...)
  - PARTNUM, PARTSIZE, KEYS
    - KEYS: the partition high keys are specified
  - DBDPATT
    - How to create a partition name from the DBD name
  - TAKEOVER concept
    - Everything is done “temporary”
    - Implemented at the end when all was OK
    - Restart-ability during TAKEOVER

# How to Convert to HALDB

- Database is 24\*7
  - ▶ Can not take the database offline
  - ▶ Conversion requires that the database is offline
  - ▶ REALLY ?
    - ORF may be able to help



# ORF Online Reorg Facility

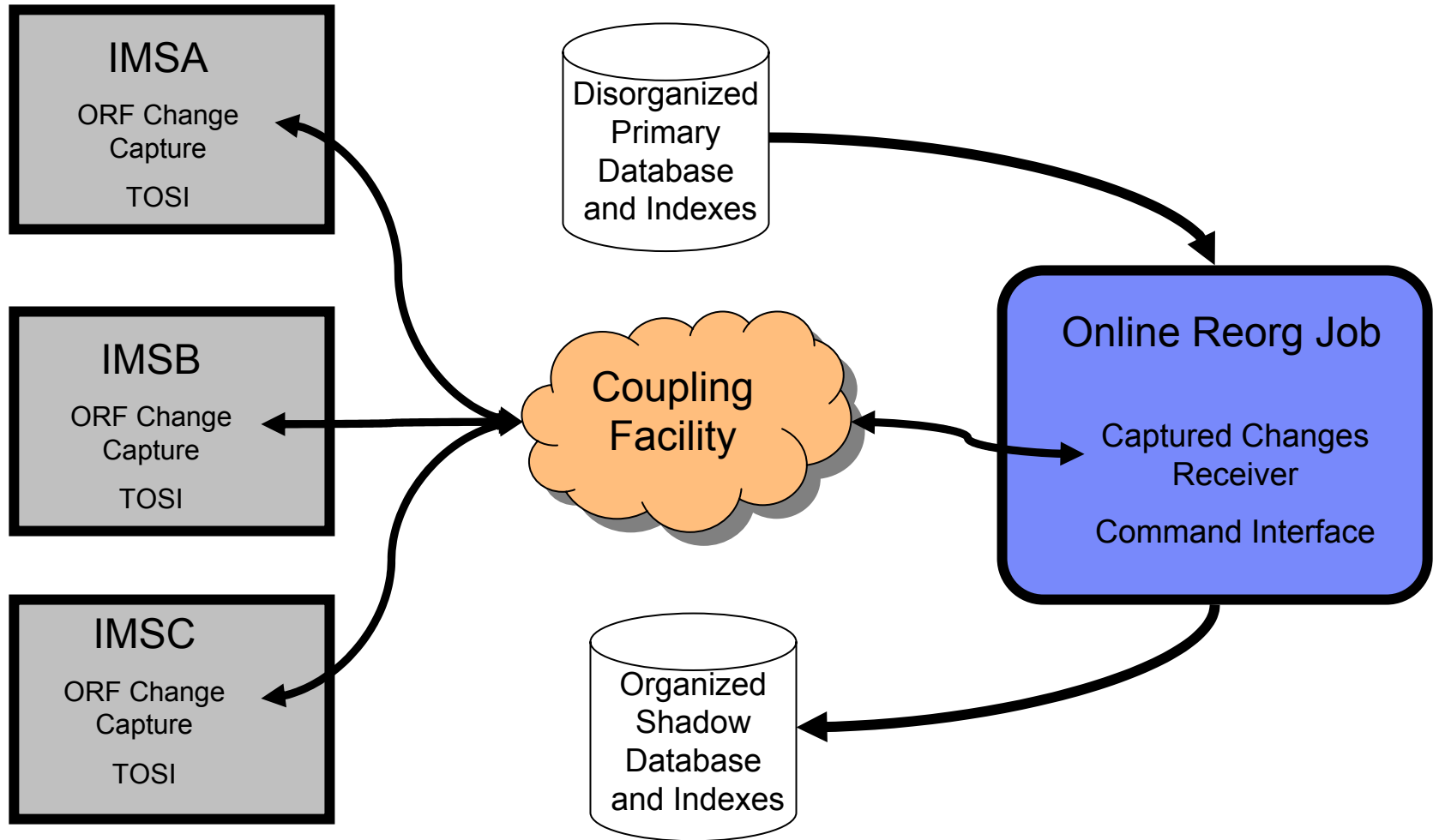


# An Online Reorg for **All** Full Function Databases

- HDAM/HIDAM
- PHDAM/PHIDAM - HALDB
  - ▶ Entire database – all partitions
  - ▶ **Single** partition
- Secondary indexes
  - ▶ When primary DB reorganized
  - ▶ Index only
- Partitioned secondary indexes
  - ▶ Entire index
  - ▶ **Single** partition

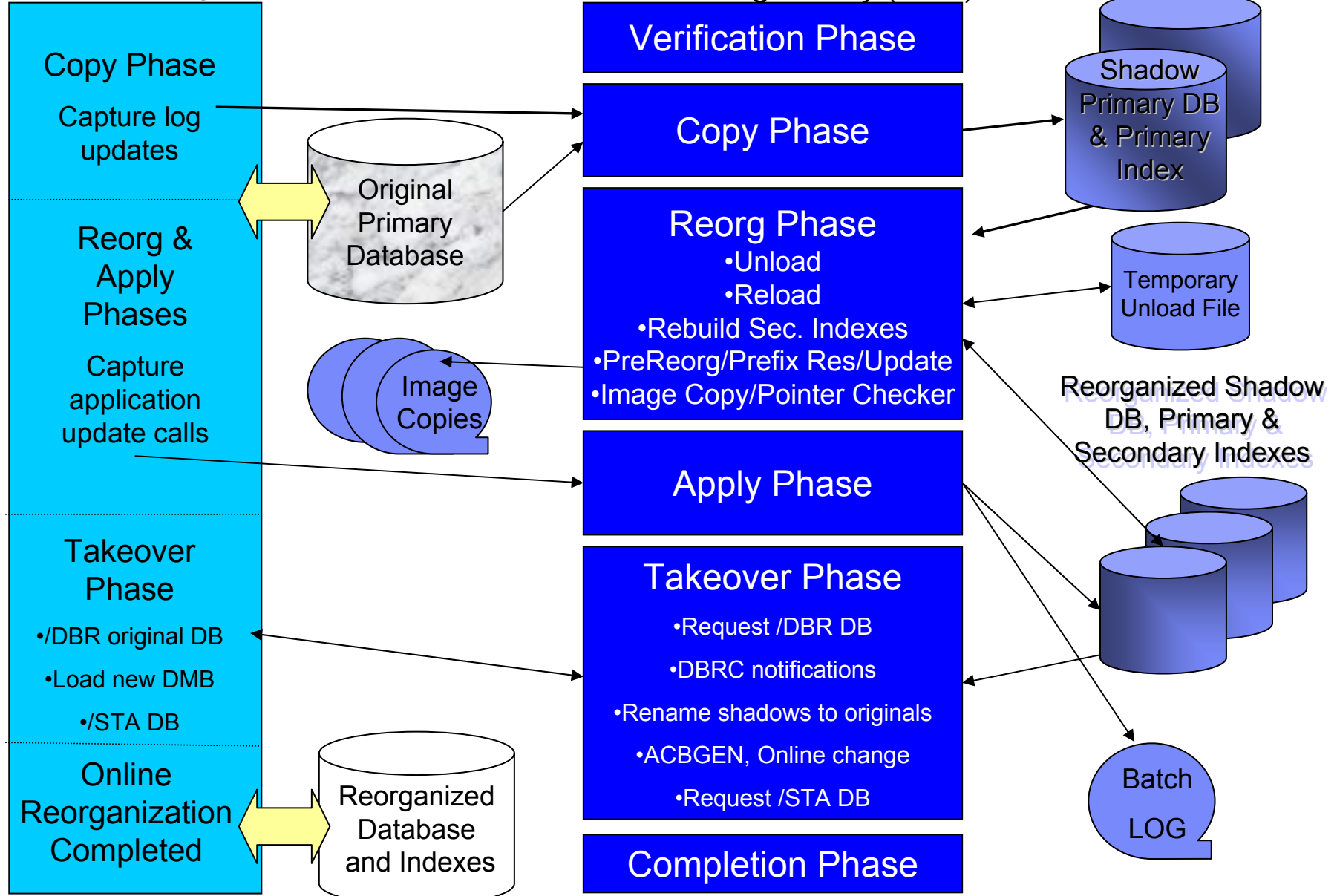


# Online Reorg in a Data Sharing Environment



IMS Control Region(s)

Online Reorg Facility (ORF)



## Copy Phase

- Activate capture of log updates to primary database in online systems
  - ▶ STOP/START primary database to create sync point
- Copy original database data sets to shadow data sets
- Switch from log updates to application updates in online systems
  - ▶ STOP/START primary database to create transition point
- Apply log updates to shadow data sets
- Build shadow primary index





# Reorg Phase

- Unload shadow database
  - ▶ Temporary unload data set
  - ▶ Optionally create permanent HD unload
  
- Preorganization
  - ▶ if logical relationship
  - ▶ DBR
  
- Reload into shadow data sets
  - ▶ USEREXIT support (also used in Apply Phase)



## Reorg Phase (continued)

- Build secondary indexes
  - ▶ Sparse routines and NULLVAL supported
  - ▶ Shared secondary indexes supported
  
- Prefix Resolution & Update
  - ▶ if logical relationships
  
- Image copy all recoverable data sets
  - ▶ Optional Pointer Checker (of shadow data sets)
  - ▶ IC COMP/COMPRTN supported
  - ▶ Dynamic allocation or predefined image copy data sets
  - ▶ Registered to DBRC as batch IC during 'Takeover Phase'



## Apply Phase

- Replicate captured application update calls against shadow data sets
  - ▶ Reload userexit support
  
- When almost caught up...
  - ▶ CHANGE.DB NOAUTH
  - ▶ Request online IMS systems to DBR DBDs for primary DB and indexes
  
- Finish replicating captured application update calls
  
- Verify DBs are DBR'd
  - ▶ Deactivate capturing application update calls in online IMS system(s)
  - ▶ Replicate any additional update calls if needed
  
- Transition from Apply to Takeover Phase can be delayed until a specific time of day



# Takeover Phase

- Checkpoint restart data
  - ▶ We are now restartable
  
- DBRC notifications
  - ▶ NOTIFY.REORG
  - ▶ NOTIFY.IC
  - ▶ NOTIFY.PRILOG & SECLOG (in ERROR)
    - Captured changes applied to shadows
  - ▶ NOTIFY.ALLOC
  - ▶ All timestamps adjusted to after DBR and before START
  
- Rename Data Sets
  - ▶ Rename originals to .T
  - ▶ Rename shadows to originals
  - ▶ Delete or rename .T to .S
  - ▶ Copy changed DBDs to original DBD library



## Takeover Phase (continued)

- Implement Online Change
  - ▶ Perform ACBGEN for any changed DBDs
  - ▶ Copy new ACBs to all online systems' ACBLIBA and ACBLIBB libraries
  - ▶ Request DMB to be replaced on all online IMS systems
  - ▶ Can be turned off
    - DB will remain stopped and in NOAUTH status
    - Return code will be set to 4
- Change.DB AUTH
- Request online IMS systems to START DBs
- Additional checkpoints occur after each step is completed (for restart)

# Post Takeover

- Log recovery
  - ▶ Adjust timestamps in log created during Apply Phase
  - ▶ `CHANGE.PRILOG/SECLOG NORMAL`



# Stopping databases



## Database Being Accessed by BMPs

- BMPs must be stopped when ORF needs to STOP or DBR a DB
  - ▶ Potentially 'long running'
  - ▶ Interface with Program Restart Facility or ORF region controller front-end
    - New BMPs are 'paused' until DB is restarted
    - Existing BMPs – next CHKP
      - HALDB – BMP is paused until DB is restarted
      - Non-HALDB –
        - pseudo U3303
        - Job restarted from last checkpoint after DB is restarted





# Data Being Accessed by IMS Terminals

- New transaction arrives when DB is DBR'd
  - ▶ Transaction placed on suspend queue
    - Exit/Automation to process suspend queue and reissue transaction
    - /STA DB will requeue message
  - ▶ Takeover WINDOW to reduce impact



# Data Being Accessed by CICS Terminals

- SCHEDULE PSB request when DB is DBR'd
  - ▶ ORF detects that it has DBR'd the DB
    - Thread is put into wait
  - ▶ Takeover WINDOW to reduce impact
  
- Typically short lived



# Data Being Accessed through ODBA

- APSB request when DB is DBR'd
  - ▶ ORF detects that it has DBR'd the DB
    - Application TCB is put into wait
  - ▶ Takeover WINDOW to reduce impact
  
- Typically short lived



## Controlling When /DBR Occurs

- Specify a time range when ORF can issue /DBR to put shadow data sets online
  - ▶ Can reduce potential impact to incoming requests
  
- TAKEOVER.WINDOW(02:00,06:00,WTOR)
  - ▶ Begin time - Earliest time of day that takeover will start
    - ORF job 'idles' with DB still online (replicating changes)
  
  - ▶ End time - Latest time of day that takeover will start
  
  - ▶ Action – what to do if 'End time' has passed



# Converting To HALDB and Maintaining Partitions



# How to Convert to HALDB

- Convert online
  - ▶ ORF can help
    - Creating the shadow datasets
    - Capturing the changes during conversion
    - Reapply the changes
      - Have been captured as non HALDB
      - Applied to HALDB
    - Online change interface



# How to Convert to HALDB

- Online (cont.)
  - ▶ Using JCL and Control Cards

```
CONVERT DBD(dbdname) PARTNUM(5) -  
        DBDPATT(***** ...) -  
        ONLINE(Y)
```
  - ▶ TAKEOVER
    - Online change special
      - Utility can do ACBGEN
      - Will end with RC 4
      - User must do online change
        - To ACBLIBA, ACBLIBB
        - Utility to obtain the PSB names
    - Remember the takeover window parameter in ORF

# How to Maintain Partitions

- What will happen to our data
  - ▶ Data separated by high key
  - ▶ Some database records will be deleted
  - ▶ New database records will be inserted
  - ▶ Some partitions may have less data
    - Does not pose a problem
  - ▶ Some partitions may have grown
    - May require us to split that partition
  - ▶ Rearrange the partition number and boundaries
    - Consolidate “smaller” partitions
    - Less image copies to maintain





# How to Maintain Partitions

- Maintain manually
  - ▶ Locate the partition(s)
    - If more than one, they must be in sequence
  - ▶ Evaluate new keys and partition number
  - ▶ Prepare the DBRC partition statements
    - Avoid to delete the partition with the highest key
    - What will the new partition numbers be?
    - Make sure that the partition names are not used
      - In DBRC
      - In the control region
  - ▶ Prepare the IDCAMS allocation statements
  - ▶ Take the partitions offline



# How to Maintain Partitions

- Maintain manually (cont.)
  - ▶ Run UNLOAD
    - Make sure you use the DFSHALDB DD statement

```
//DFSHALDB DD *  
HALDB PCB=(partname,number)
```
  - ▶ Run the DBRC list
  - ▶ Run the IDCAMS list
  - ▶ Run partition initialization
  - ▶ Run RELOAD
  - ▶ Do the image copies
  - ▶ Get the new partitions online
    - Issue /STA mastername OPEN to force structure rebuild
    - Need to start the remaining DBRed partitions.

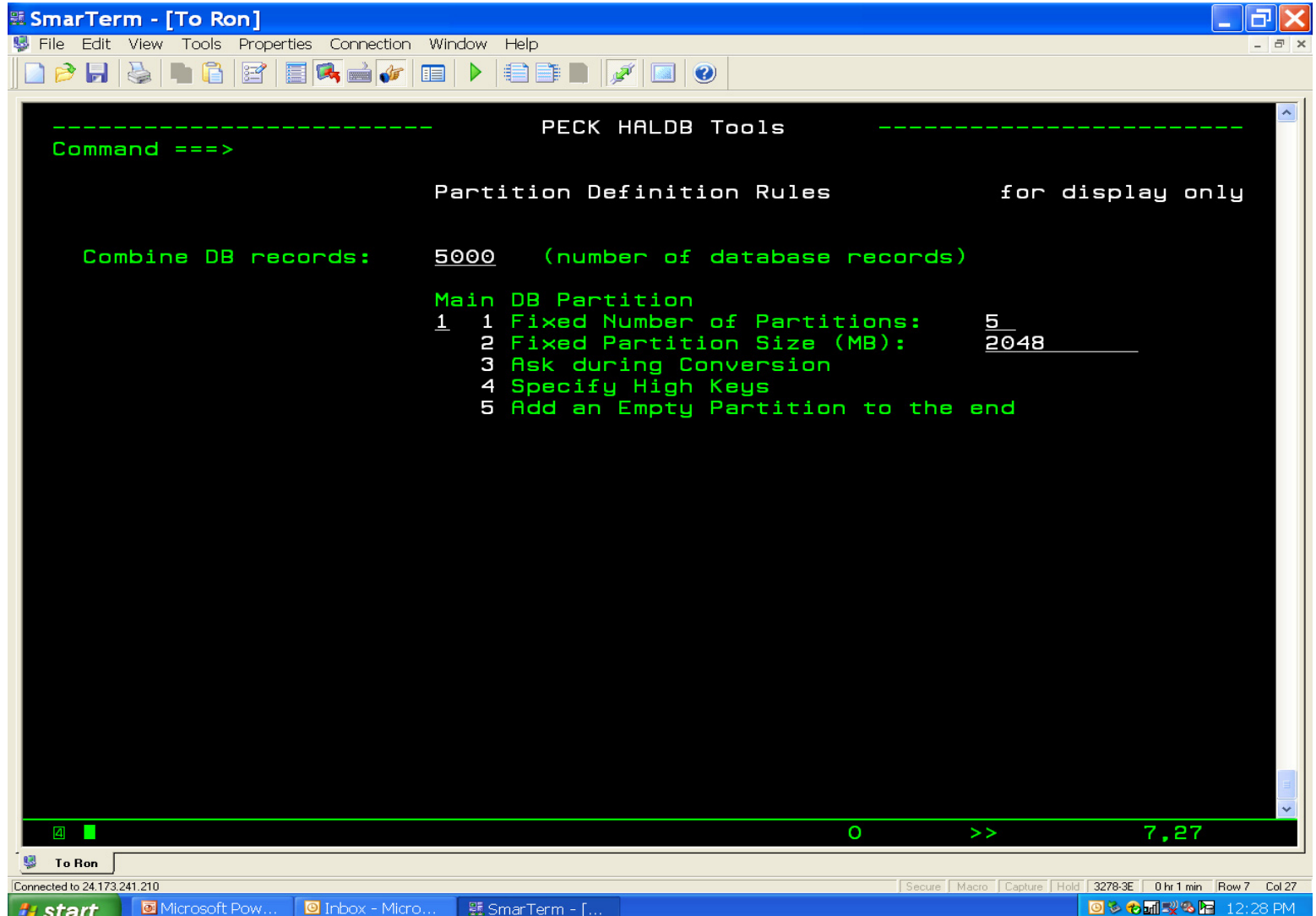


# How to Maintain Partitions

- Using a tool
  - ▶ ISPF based application
    - Creates all the necessary JCL and control cards
    - User submits each step
    - Next step is created when previous step was OK
    - ISPF panels to specify options



# How to Maintain Partitions



The screenshot shows a terminal window titled "SmarTerm - [To Ron]" with a menu bar (File, Edit, View, Tools, Properties, Connection, Window, Help) and a toolbar. The main display area shows the following text:

```
----- PECK HALDB Tools -----
Command ==>

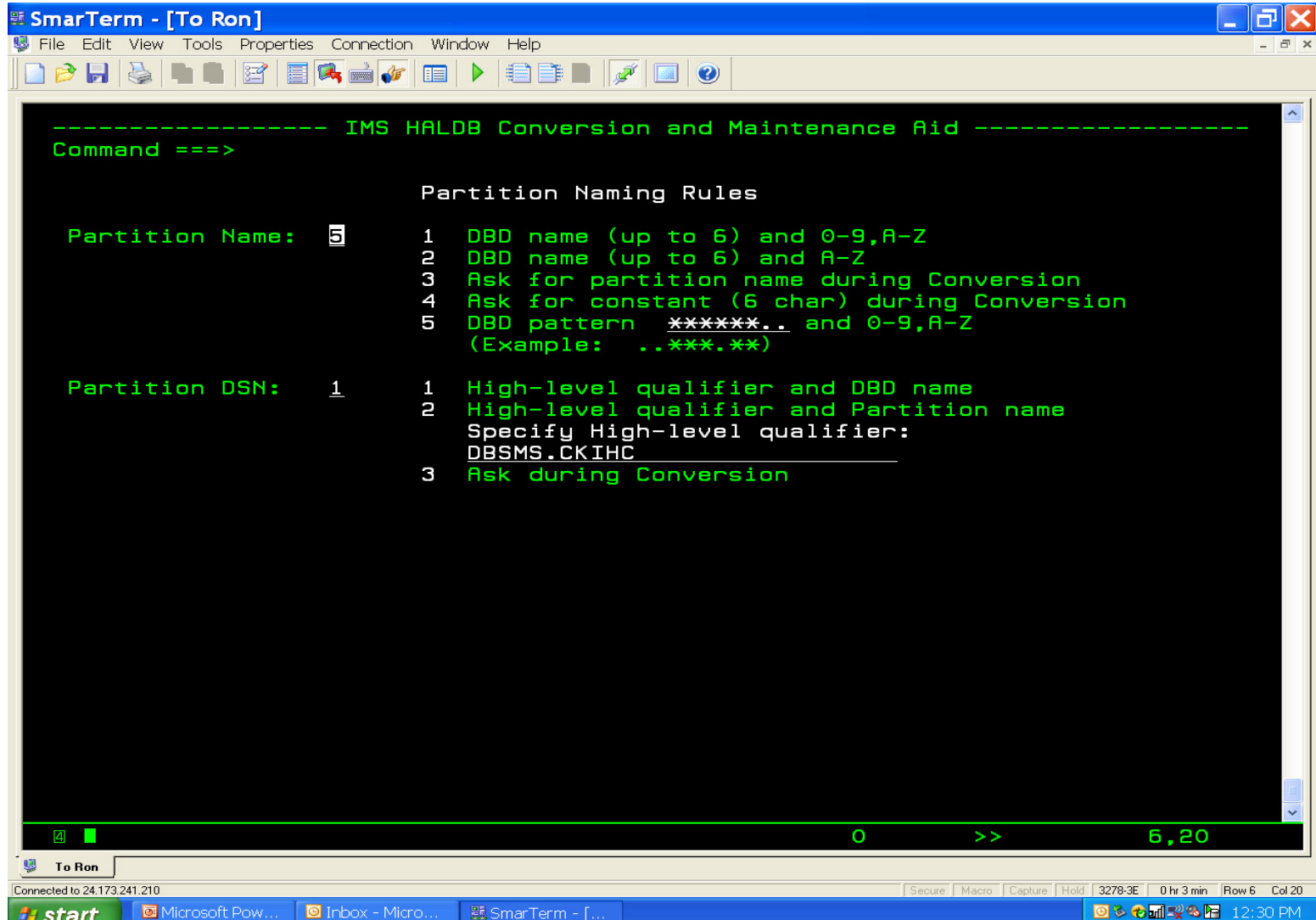
Partition Definition Rules          for display only

Combine DB records:   5000   (number of database records)

Main DB Partition
 1  1 Fixed Number of Partitions:      5
 2  2 Fixed Partition Size (MB):      2048
 3  3 Ask during Conversion
 4  4 Specify High Keys
 5  5 Add an Empty Partition to the end
```

At the bottom of the terminal window, there is a status bar showing "0 >> 7.27". Below the terminal window, the Windows taskbar is visible, showing the "start" button and several open applications: "Microsoft Pow...", "Inbox - Micro...", and "SmarTerm - [...]". The system tray on the right shows the date and time as "12:28 PM".

# How to Maintain Partitions



The screenshot shows a terminal window titled "SmarTerm - [To Ron]". The window contains the following text:

```
----- IMS HALDB Conversion and Maintenance Aid -----  
Command ==>  
  
Partition Naming Rules  
  
Partition Name: 5  
1 DBD name (up to 6) and 0-9,A-Z  
2 DBD name (up to 6) and A-Z  
3 Ask for partition name during Conversion  
4 Ask for constant (6 char) during Conversion  
5 DBD pattern *****. and 0-9,A-Z  
  (Example: ..***.**)  
  
Partition DSN: 1  
1 High-level qualifier and DBD name  
2 High-level qualifier and Partition name  
  Specify High-level qualifier:  
  DBSMS.CKIHC  
3 Ask during Conversion
```

At the bottom of the terminal window, there is a status bar showing "0 >> 6,20". Below the terminal window, the Windows taskbar is visible, showing the "start" button and several open applications: "Microsoft Pow...", "Inbox - Micro...", and "SmarTerm - [...]". The system tray shows the time as "12:30 PM".

# How to Maintain Partitions

- Using JCL and Control Cards
  - MAINTAIN DBD(mastername) PARTITION(partname) -  
DBDPATT(\*\*\*\*\* ...) -  
PARTNUM(num)
- PARTITION or PARTLIST
  - ▶ Mutually exclusive
  - ▶ When a list is specified, partitions must be in sequence
    - Key sequence is not the same as name sequence
    - Partition number may jump
      - There have been deleted partitions
- PARTNUM, PARTSIZE, KEYS
  - ▶ When keys are specified, we check for key range conflict
  - ▶ Highest key is reset to high partition key



# How to Maintain Partitions

- Database is 24\*7
  - ▶ Partitions can not be offline
    - At least not for a long time
  - ▶ ORF can do already partitions
    - Using shadow datasets
    - Reorg with changing partitions
    - Temporary RECONs are used
    - TAKEOVER will use the original RECON



# How to Maintain Partitions

- Online
  - ▶ Using JCL and Control Cards
    - MAINTAIN DBD(master) PARTLIST(part1,part2) -
    - PARTNUM(3) -
    - DBDPATT(\*\*\*\*\* ...) -
    - ONLINE(Y)
  - ▶ Rearrange 2 partitions into 3
    - A new partition will be created
    - The existing partitions are reused
  - ▶ TAKEOVER
    - Partitions are defined at that time
    - All necessary IMS commands are issued





# Additional Tools

- Loading and deleting a single partition
  - ▶ When loading (with PROCOPT=L)
    - Secondary index performance problem
    - Tool collects the secondary index records
    - It sorts and inserts in sequential mode
  - ▶ When deleting a partition
    - Need to maintain the secondary indexes
    - Tool deletes only the index records pointing to the deleted partition



## Additional Tools

- DBRC support
  - ▶ Cloning HALDB definitions
    - Using the definition of an existing RECON
    - Create DBRC partition definition with a different DSNPREFIX
    - Ideally for test environments
  - ▶ Copying HALDB database to a different IMS system
    - Using the original HALDB definition
      - With a different DSNPREFIX
    - Copies the “non fuzzy” IC records to a different recon
    - Creates the IDCAMS statements
    - User needs to do a recovery
      - Using the alternate RECON
    - Ideally for transporting production to a different IMS



# Additional Tools

- Fallback to non HALDB
  - ▶ To be done after several days.
  - ▶ Convert has save options
    - DBD
    - “old” DBRC
    - “old” allocation
  - ▶ Unload done as HALDB
  - ▶ Utility to convert to non HALDB format
  - ▶ Reload using the saved elements



# Q&A

