DB2 Utilities - A New Day - A New Way

IBM System z Software Teleconference

IBM zIIP Engines, IBM DB2 Utilities, DB2 Automation Tool and DB2 Thread Expert

IBM Information Management software

Bryan Smith bfsmith@us.ibm.com
Senior Technical Lead for DB2 for z/OS Utilities

David Schwartz dschwar@us.ibm.com
DB2 Tools Strategy and Planning

ON DEMAND BUSINESS

Abstract

- Teleconference
  - Have you thought about your DB2 Utilities from IBM lately? If you are like most of our customers…probably not…they just run. Join us today to learn how you can augment your Utilities….using IBM DB2 Automation Tool to automate those mundane tasks and then using the DB2 Thread Expert to cancel those long running applications that have stopped you from running your Utilities when they need to be run.
- Audience
  - DBAs, System DBAs and System Programmers.

Agenda

- Ways to improve
  - Performance
  - Integrity
  - Efficiency
  - Availability
  - Migration to Version 8 if you are not a DFSORT user
- REORG scenario to illustrate best practices
- Ways to automate periodic maintenance
- How to ensure successful completion of utilities
- New accounting fields called IIP and IIPCP
- Q&A

Question 1:

How can I improve the performance and integrity of my backup/recovery processes?

COPY/RECOVER/QUIESCE Best Practices

- COPY
  - PARALLEL keyword provides parallelism for lists of objects
  - CHECKPAGE YES
  - Incremental copy rule-of-thumb: Consider using incremental image copy if
    - <5% of pages are randomly updated (typically means less than 1% of rows updated)
    - <80% of pages are sequentially updated such that updated pages are together and separated from non updated pages
- RECOVER
  - PARALLEL keyword provides parallelism for lists of objects
  - Compressed pagesets result in faster restore phase
  - Enable Fast Log Apply (which can use dual-copy logs)
  - <=10 jobs/member with LOGAPSTG=100MB, 20-30 objects per RECOVER
- QUIESCE
  - WRITE NO is less disruptive (no quiescing of COPY=NO indexes)
  - Use TABLESPACESET
- Large BUFNO
  - Anecdotal evidence of improved performance with a large BUFNO (e.g., BUFNO=100) but we have not seen this in our benchmarks – we suspect that this helped in cases where I/O configuration was not well tuned

Question 2:

What options should I use on LOAD and REORG to ensure maximum efficiency?

LOAD/REORG Best Practices

- LOG NO reduces log volume; take inline copies when possible
- KEEPDICTIONARY (track dictionary effectiveness with history statistics PAGESAVE)
- Load Partition Parallelism (V7)
  - Not individual LOAD part level jobs
  - Enable ESS Parallel Access Volume (PAV)
- On V7, REORG SORTDATA to use table space scan and then sort
- REORG … NOSYSREC to avoid I/O (forced for SHRLEVEL CHANGE)
  - Use only if taking full image copy before REORG
- Inline COPY & STATISTICS (Also applies to REBUILD INDEX)
  - Collect only column stats on columns used in SQL and SAMPLE
- Index parallelism (SORTKEYS)
  - Remove SORTWKxx and use SORTDEVT/SORTNUM
  - Also applies to REBUILD INDEX
- Preprocess LOAD data if possible
  - Avoid data conversion, use internal representation if possible
  - Sort data in cluster order (unless data is randomly accessed via SQL)

Question 3:

What can I do to deliver the highest availability to my applications and users?

High availability Best Practices

- CREATE INDEX DEFER followed by REBUILD INDEX
  - As of V8, dynamic SQL will not select the index until it is built
- CHECK INDEX
  - SHRLEVEL CHANGE
    - Even if you don’t have FlashCopy V2
    - FlashCopy2 doesn’t require double the disk space
- REORG
  - SHRLEVEL REFERENCE (allows read access during RELOAD phase)
  - SHRLEVEL CHANGE
    - TIMEOUT TERM frees up the objects if timeouts occur in getting drains
    - DRAIN ALL (better change of successful drain)
    - MAXRO and DRAIN_WAIT = lock timeout ZPARM/2 (30 seconds by default)
    - RETRY = utility lock timeout multiplier (6 by default)
    - RETRY_WAIT = DRAIN_WAIT*RETRY
- RUNSTATS
  - SHRLEVEL CHANGE
- COPY
  - SHRLEVEL CHANGE
- LOAD RESUME
  - SHRLEVEL CHANGE

Question 4:

What do I need to do before migrating to Version 8 if I am not a DFSORT user?

Sorting w/DFSORT Best Practices

- DFSORT installation options
  - If not a DFSORT user, **MANDATORY READING -> APAR II14047**
  - Leave the default for SIZE set to MAX
  - Don’t bother with changing TMAXLIM (initial storage for each sort)
  - Consider adjusting DSA (Dynamic Size Adjustment)
    - R14 default is 32M; V1R5 default is 64M
    - You could set this to 128M, but then look to see if DFSORT ever uses this much (needs REGION > 128M!)
    - Follow DFSORT tuning recommendation to use hiperspaces, data spaces, etc. (if not on 64-bit LPAR)
- Remove SORTWKxx and use SORTDEVT/SORTNUM
  - This will use dynamic allocation
  - To direct datasets to storage group, use ACS routines

    http://www.redbooks.ibm.com/abstracts/TIPS0530.html?Open
- >64K track dsets for DFSORT supported in z/OS 1.7

A scenario

Reorg scenario

- Reorg tablespaces: ADHTSTDB.ADHTSTTS and ADHTSTDB.ADHTSTTT
- Invocation:

```
REORG TABLESPACE ADHTSTDB.ADHTSTTS
REORG TABLESPACE ADHTSTDB.ADHTSTTT
```

- Best practices followed?

```
LISTDEF APPL1 INCLUDE ADHTSTDB.ADHTSTT*
TEMPLATE BACKUP1 DSN &DB..&TS..D&DA..T&TI. UNIT=SYSALLDA
REORG TABLESPACE LIST APPL1
LOG NO
KEEPDICTIONARY
SORTDATA          -- Default in V8
NOSYSREC          -- Take an image copy first!
COPYDDN(BACKUP1)
STATISTICS
   TABLE (T1) SAMPLE 25 COLUMN (C1, C2)  -- Only cols in SQL
   TABLE (T2) SAMPLE 25 COLUMN (C5, C12) -- Only cols in SQL
SORTKEYS
SORTDEVT          -- SORTNUM defaults to sort component’s parms
```

Provide logic to routine maintenance

- Leverage the ability to invoke utilities programmatically via stored procedures
  - DSNUTILS for EBCDIC parameters
  - DSNUTILU for UNICODE parameters

Provide logic to routine maintenance

- Example (using REXX):

```
/* REXX */
…
ADDRESS DSNREXX "CONNECT DB2P"
IF SQLCODE ¬= 0 THEN CALL SQLCA

Uid='';Restart=''; Utstmt=,
‘REORG TABLESPACE ADHTSTDB.ADHTSTTS’,
  ‘LOG NO KEEPDICTIONARY’,
  ‘SORTDATA SORTKEYS SORTDEVT’
  ‘STATISTICS’,
    ‘TABLE (T1) SAMPLE 25 COLUMN (C1, C2)’,
    ‘TABLE (T2) SAMPLE 25 COLUMN (C5, C12)’
Utility='REORG TABLESPACE'
CopyDSN1=‘DSN.FIC.ADHTSTTS.VERSION(+1)'
CopyDEVT1='SYSDA'
CopySpace1=1
```

```
ADDRESS DSNREXX "EXECSQL" ,
"CALL DSNUTILS(:UID, :RESTART,                  " ,
"          :UTSTMT,                        " ,
"          :RETCODE,                        " ,
"          :UTILITY,                       " ,
"          :RECDSN  ,:RECDEVT  ,:RECSPACE  ," ,
"          :DISCDSN ,:DISCDEVT ,:DISCSPACE ," ,
"          :PNCHDSN ,:PNCHDEVT ,:PNCHSPACE ," ,
"          :COPYDSN1,:COPYDEVT1,:COPYSPACE1," ,
"          :COPYDSN2,:COPYDEVT2,:COPYSPACE2," ,
"          :RCPYDSN1,:RCPYDEVT1,:RCPYSPACE1," ,
"          :RCPYDSN2,:RCPYDEVT2,:RCPYSPACE2," ,
"          :WORKDSN1,:WORKDEVT1,:WORKSPACE1," ,
"          :WORKDSN2,:WORKDEVT2,:WORKSPACE2," ,
"          :MAPDSN  ,:MAPDEVT  ,:MAPSPACE  ," ,
"          :ERRDSN  ,:ERRDEVT  ,:ERRSPACE  ," ,
"          :FILTRDSN,:FILTDEVT ,:FILTRSPACE)"
IF SQLCODE < 0 THEN CALL SQLCA
…
```

Provide logic to routine maintenance

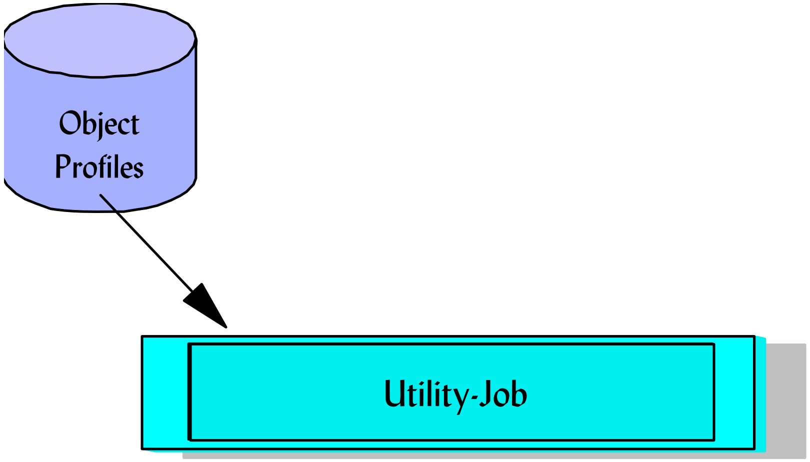
- Rich logic can be provided to
  - Take an image copy before running REORG with NOSYSREC
  - Examine statistics (from RUNSTATS or the Real-time statistics) to determine when to run a utility
  - Examine a control table to determine windows when maintenance can or cannot be run
  - …
- You have full control without needing individual threshold keywords on each utility
- But, maybe you don’t want to write this type of logic yourself… Dave, take it away…

How can we make this easier?

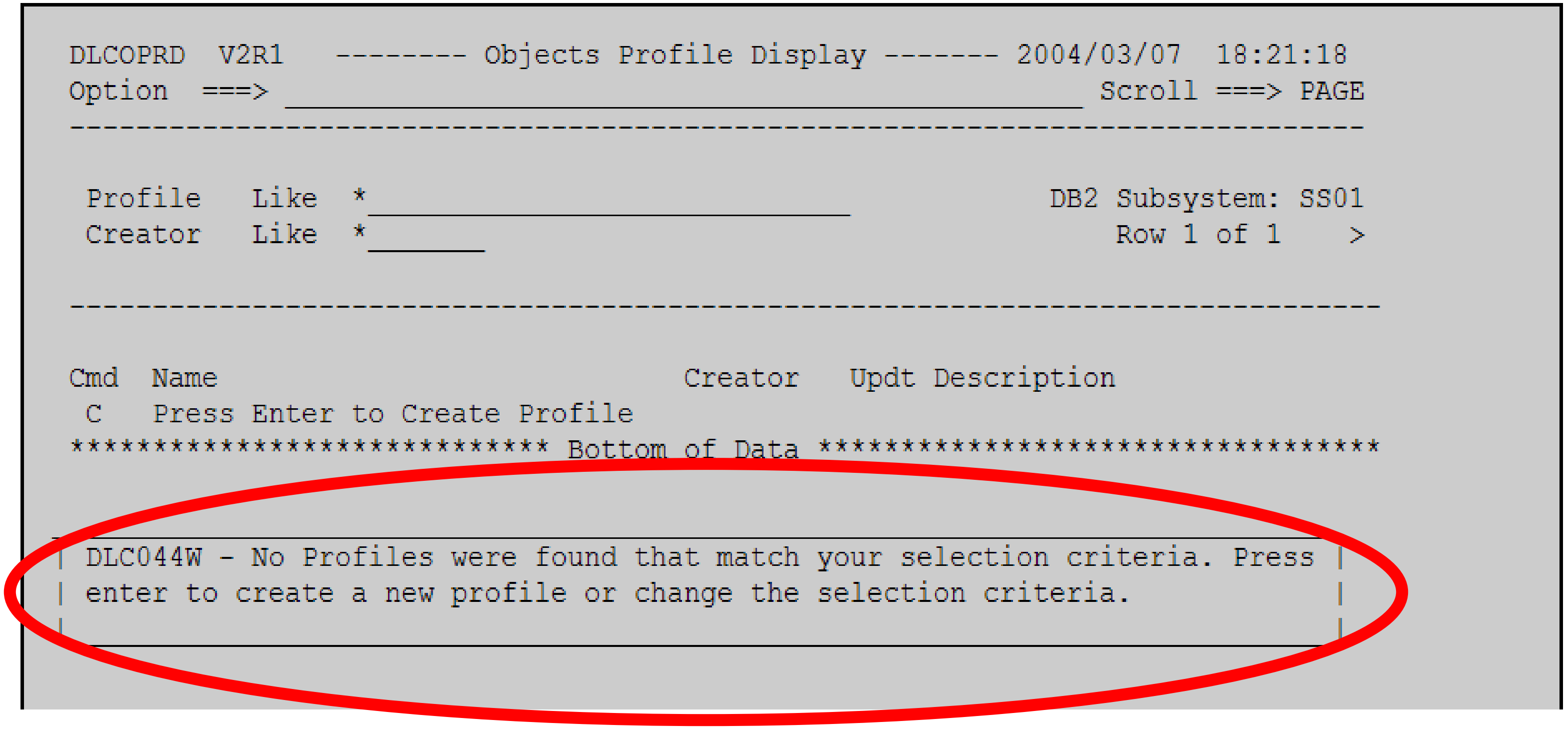
- If you don’t want to build this logic, then automation is available
- IBM DB2 Automation Tool for z/OS
- Automation Tool is keyed off of a series of profiles
  - To perform this same task
    - Create Object profile
    - Create Utility Profile
    - Set exceptions and forget it
  - Only those objects in the Object Profile that meet the exception criteria are chosen for the utility operation.

First we build an Object Profile

An Object Profile contains the list of objects to be considered for REORG

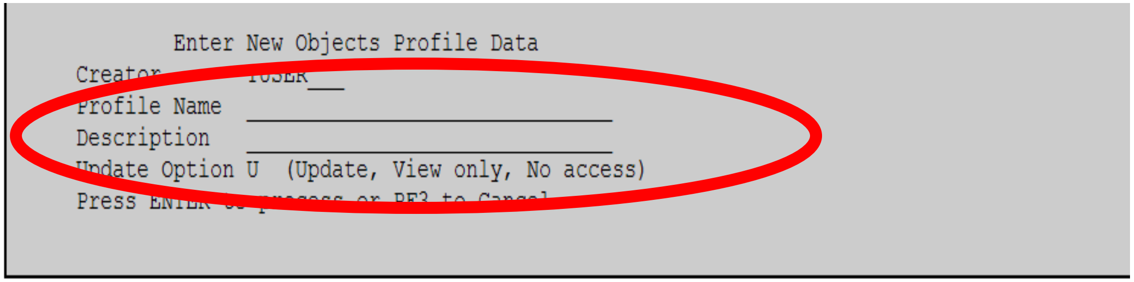


Generating an Object Profile

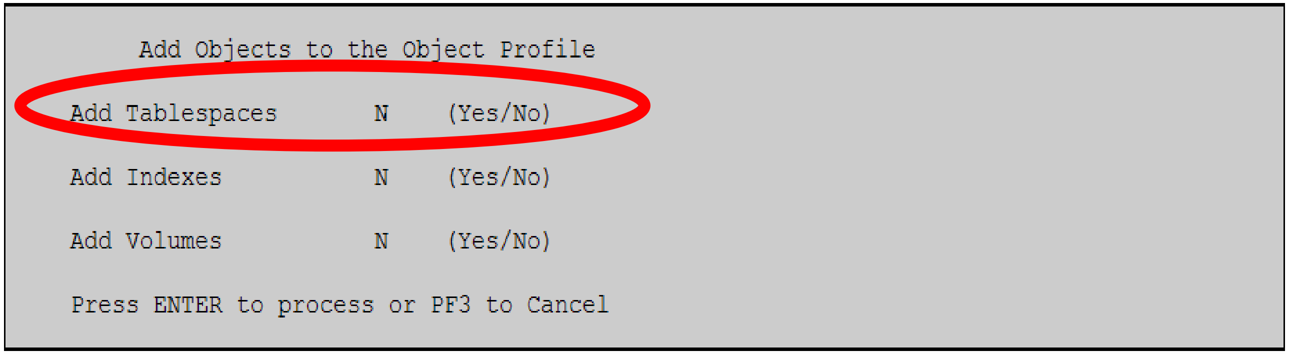


```
DLCOPRD  V2R1   -------- Objects Profile Display ------- 2004/03/07  18:21:18
Option  ===> ________________________________________________ Scroll ===> PAGE
------------------------------------------------------------------------------

 Profile   Like  *_____________________________        DB2 Subsystem: SS01
 Creator   Like  *_______                                  Row 1 of 1    >

-------------------------------------------------------------------------------

Cmd  Name                            Creator   Updt Description
 C   Press Enter to Create Profile
***************************** Bottom of Data **********************************


| DLC044W - No Profiles were found that match your selection criteria. Press |
| enter to create a new profile or change the selection criteria.            |
|                                                                            |
```

Generating an Object Profile – Identifying what’s to be included

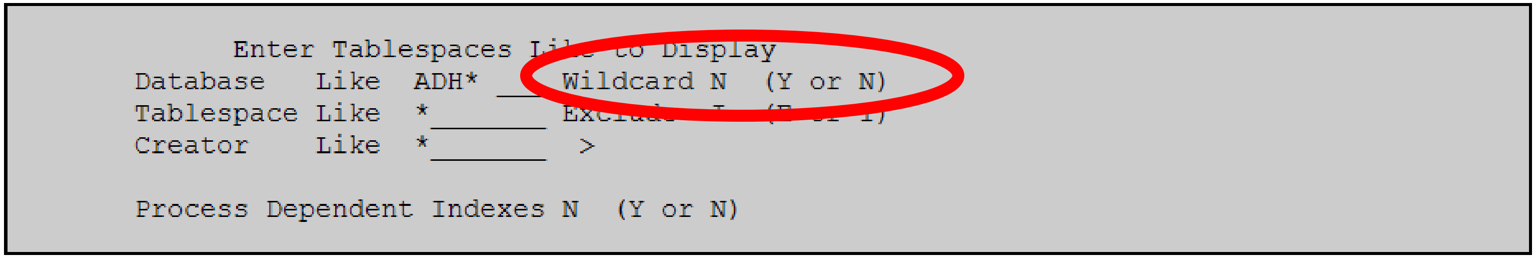


```
        Enter New Objects Profile Data
Creator         TUSER___
Profile Name    ______________________________
Description     ______________________________
Update Option U  (Update, View only, No access)
Press ENTER to process or PF3 to Cancel
```



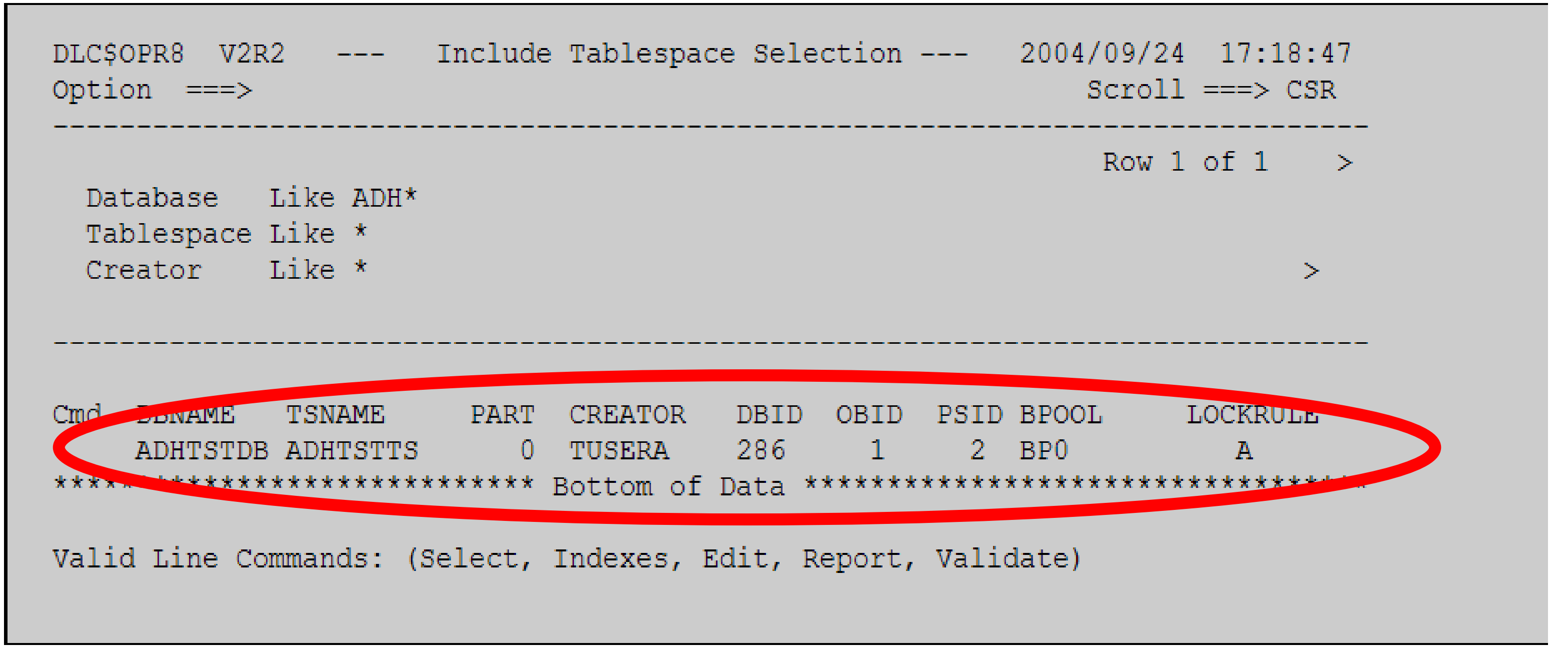
```
     Add Objects to the Object Profile

Add Tablespaces          N     (Yes/No)

Add Indexes              N     (Yes/No)

Add Volumes              N     (Yes/No)

Press ENTER to process or PF3 to Cancel
```

Generating an Object Profile – specifying the objects



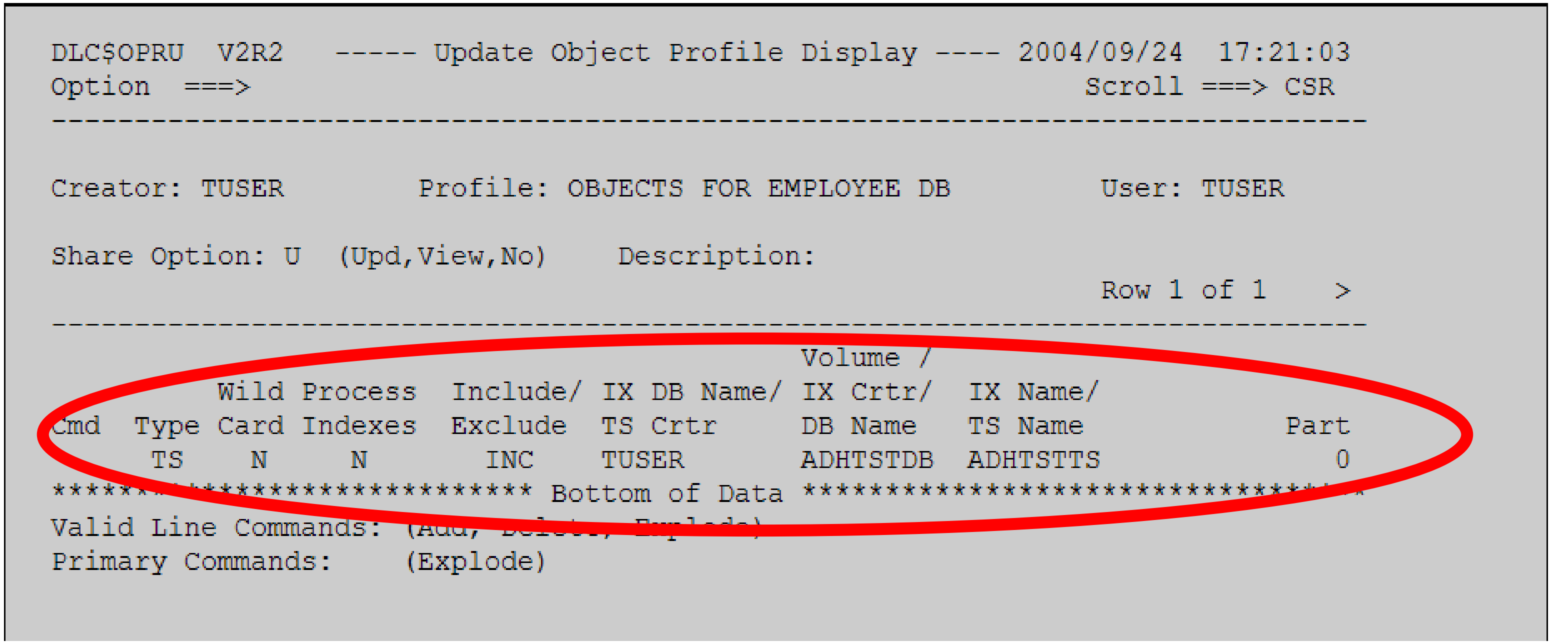
```
      Enter Tablespaces Like to Display
Database   Like  ADH* ____ Wildcard N  (Y or N)
Tablespace Like  *_______  Exclude  I  (E or I)
Creator    Like  *_______   >

Process Dependent Indexes N  (Y or N)
```

Generating an Object Profile – the results.



```
DLC$OPR8  V2R2   ---   Include Tablespace Selection ---   2004/09/24  17:18:47
Option  ===>                                                  Scroll ===> CSR
-------------------------------------------------------------------------------
                                                               Row 1 of 1    >
  Database   Like ADH*
  Tablespace Like *
  Creator    Like *                                                        >

-------------------------------------------------------------------------------

Cmd  DBNAME   TSNAME      PART  CREATOR    DBID  OBID  PSID BPOOL    LOCKRULE
     ADHTSTDB ADHTSTTS       0  TUSERA     286      1     2  BP0           A
***************************** Bottom of Data ********************************

Valid Line Commands: (Select, Indexes, Edit, Report, Validate)
```

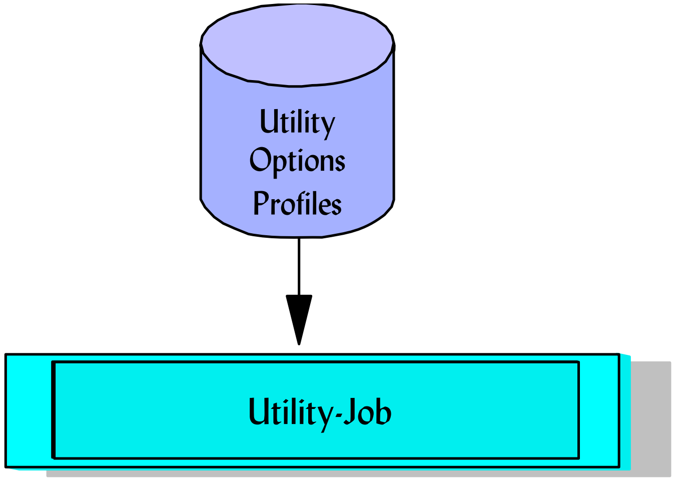
Generating an Object Profile – we're done



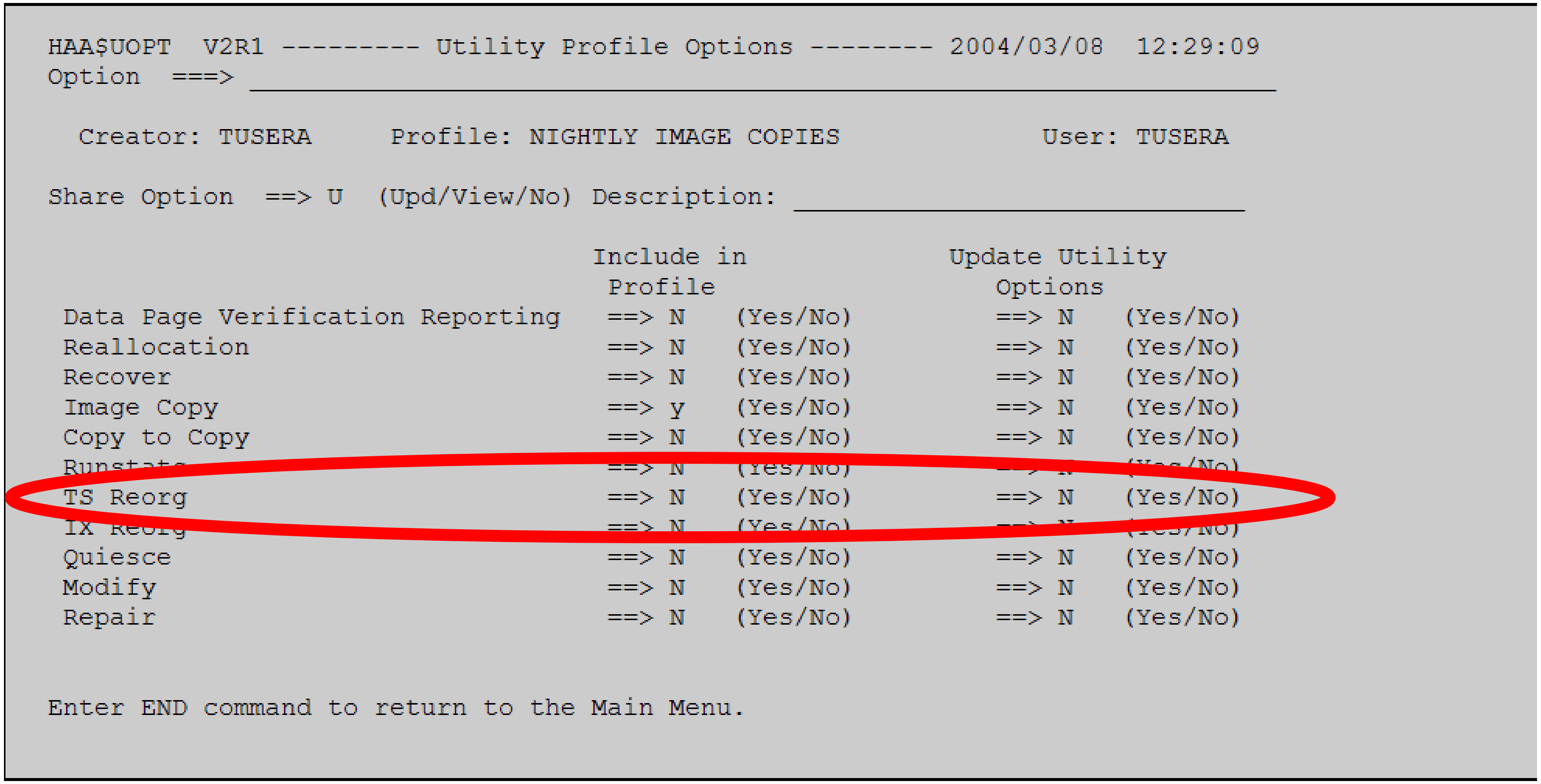
```
DLC$OPRU  V2R2   ----- Update Object Profile Display ---- 2004/09/24  17:21:03
Option  ===>                                                Scroll ===> CSR
--------------------------------------------------------------------------------

Creator: TUSER         Profile: OBJECTS FOR EMPLOYEE DB          User: TUSER

Share Option: U  (Upd,View,No)    Description:
                                                               Row 1 of 1    >
--------------------------------------------------------------------------------
                                             Volume /
          Wild Process  Include/ IX DB Name/ IX Crtr/  IX Name/
Cmd  Type Card Indexes  Exclude  TS Crtr     DB Name   TS Name             Part
      TS    N     N        INC   TUSER       ADHTSTDB  ADHTSTTS               0
***************************** Bottom of Data ********************************
Valid Line Commands: (Add, Delete, Explode)
Primary Commands:    (Explode)
```

Now, we build the Utility Profile

This is where we specify both the utility and the utility options. I've omitted the initial screens where you specify the profile name, etc. They are basically the same as for the Object Profile

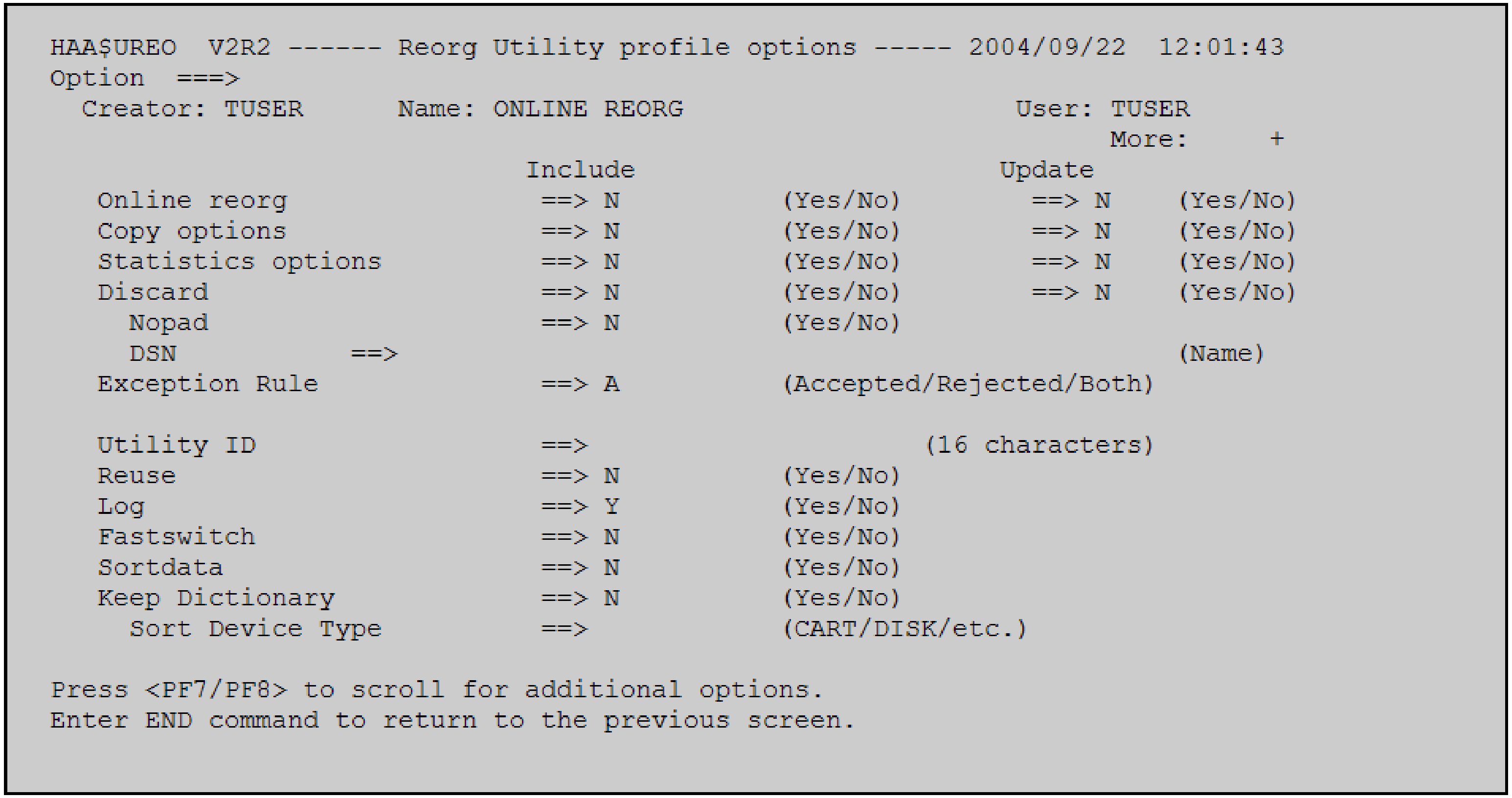


Generating a Utility Profile – We want REORG



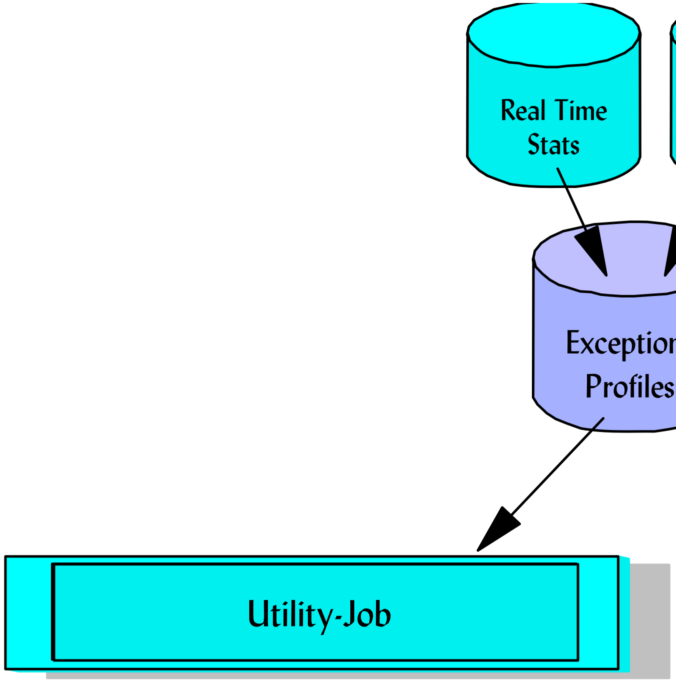
```
HAA$UOPT  V2R1 --------- Utility Profile Options -------- 2004/03/08  12:29:09
Option  ===> _________________________________________________________________

  Creator: TUSERA      Profile: NIGHTLY IMAGE COPIES            User: TUSERA

Share Option  ==> U  (Upd/View/No) Description: _____________________________

                                   Include in             Update Utility
                                    Profile                  Options
 Data Page Verification Reporting   ==> N   (Yes/No)         ==> N   (Yes/No)
 Reallocation                       ==> N   (Yes/No)         ==> N   (Yes/No)
 Recover                            ==> N   (Yes/No)         ==> N   (Yes/No)
 Image Copy                         ==> y   (Yes/No)         ==> N   (Yes/No)
 Copy to Copy                       ==> N   (Yes/No)         ==> N   (Yes/No)
 Runstats                           ==> N   (Yes/No)         ==> N   (Yes/No)
 TS Reorg                           ==> N   (Yes/No)         ==> N   (Yes/No)
 IX Reorg                           ==> N   (Yes/No)         ==> N   (Yes/No)
 Quiesce                            ==> N   (Yes/No)         ==> N   (Yes/No)
 Modify                             ==> N   (Yes/No)         ==> N   (Yes/No)
 Repair                             ==> N   (Yes/No)         ==> N   (Yes/No)


Enter END command to return to the Main Menu.
```

Entering Reorg Options – all the options are available, this is screen one but we won’t go through the rest.



```
HAA$UREO  V2R2 ------ Reorg Utility profile options ----- 2004/09/22  12:01:43
Option  ===>
  Creator: TUSER      Name: ONLINE REORG                     User: TUSER
                                                                   More:     +
                              Include                       Update
   Online reorg                ==> N          (Yes/No)        ==> N    (Yes/No)
   Copy options                ==> N          (Yes/No)        ==> N    (Yes/No)
   Statistics options          ==> N          (Yes/No)        ==> N    (Yes/No)
   Discard                     ==> N          (Yes/No)        ==> N    (Yes/No)
     Nopad                     ==> N          (Yes/No)
     DSN           ==>                                                 (Name)
   Exception Rule              ==> A          (Accepted/Rejected/Both)

   Utility ID                  ==>                     (16 characters)
   Reuse                       ==> N          (Yes/No)
   Log                         ==> Y          (Yes/No)
   Fastswitch                  ==> N          (Yes/No)
   Sortdata                    ==> N          (Yes/No)
   Keep Dictionary             ==> N          (Yes/No)
     Sort Device Type          ==>            (CART/DISK/etc.)

Press <PF7/PF8> to scroll for additional options.
Enter END command to return to the previous screen.
```

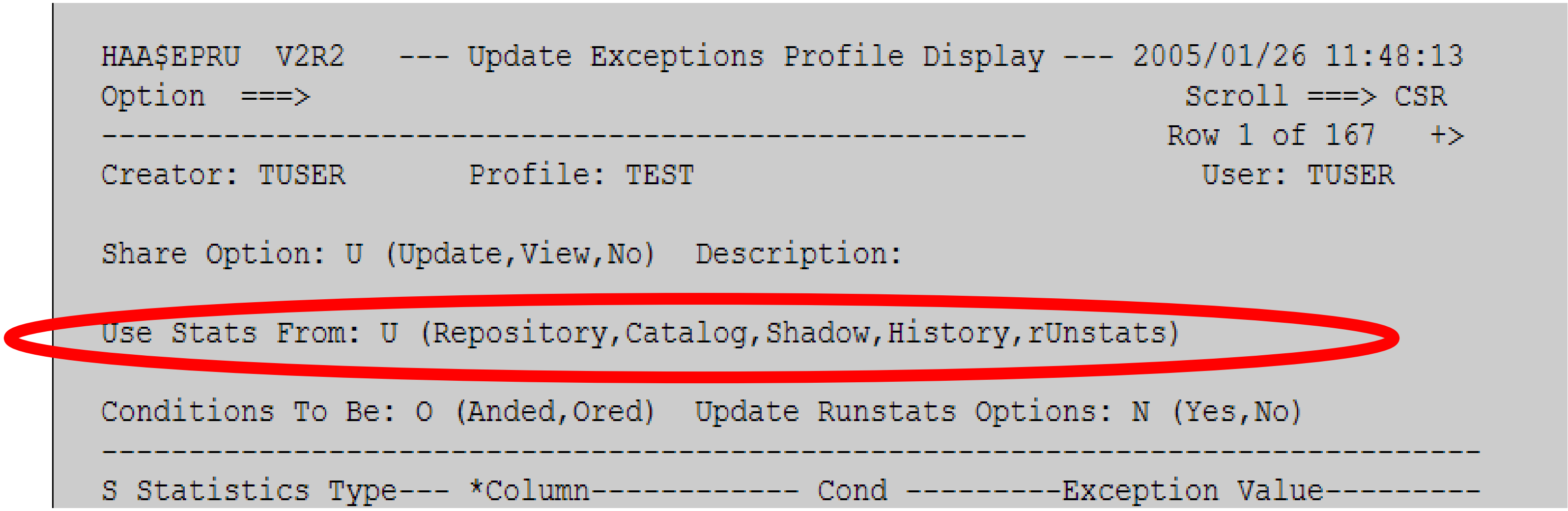
We’re almost done, now we’ll define what we want to trigger the REORG on



Creating Exception Profile

Determining what stats to use

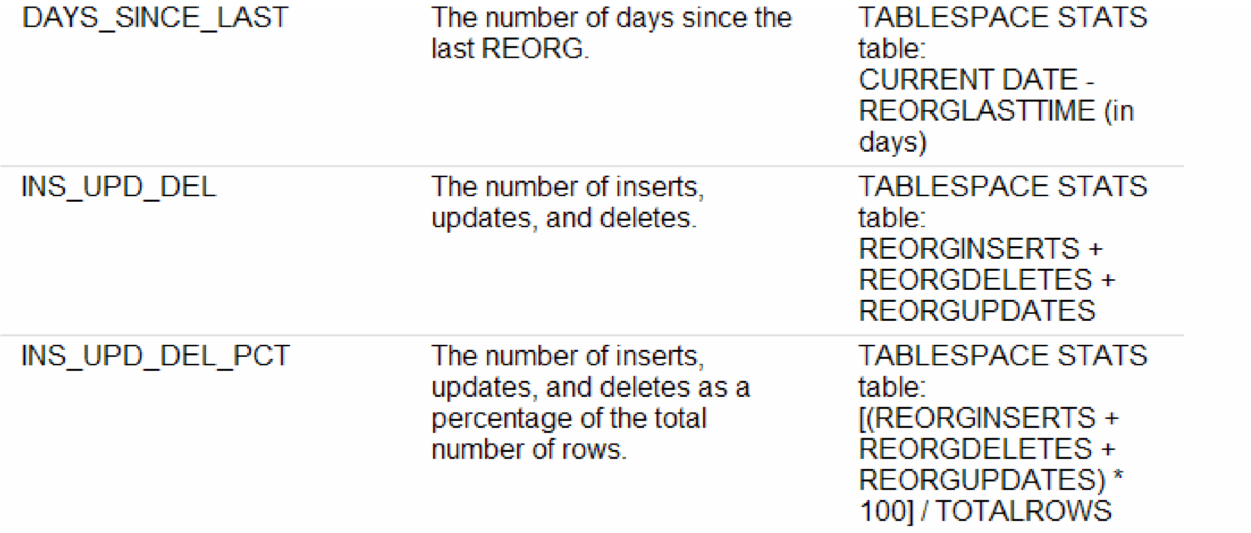
You'll notice there isn't a choice for RTS, that's because real time stats are specified just as any other statistic so the choice here is Catalog



```
HAA$EPRU  V2R2   --- Update Exceptions Profile Display --- 2005/01/26 11:48:13
Option  ===>                                                  Scroll ===> CSR
-----------------------------------------------------        Row 1 of 167   +>
Creator: TUSER       Profile: TEST                             User: TUSER

Share Option: U (Update,View,No)  Description:

Use Stats From: U (Repository,Catalog,Shadow,History,rUnstats)

Conditions To Be: O (Anded,Ored)  Update Runstats Options: N (Yes,No)
--------------------------------------------------------------------------------
S Statistics Type--- *Column------------ Cond ---------Exception Value---------
```

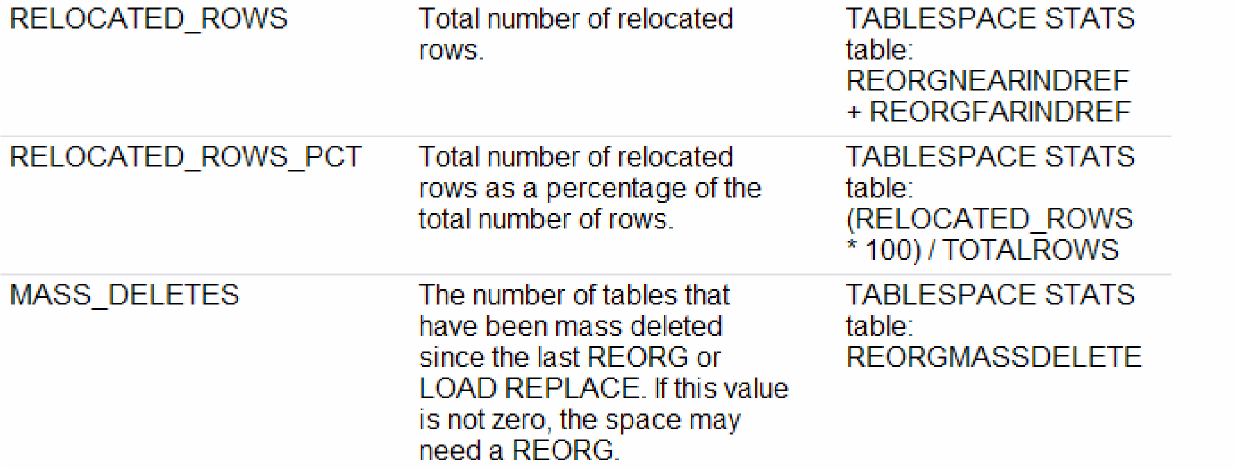
Specifying RTS values (not a complete list)

All RTS values are supported, I listed these since we're concerned with REORGing a tablespace. The next slide shows a few more.



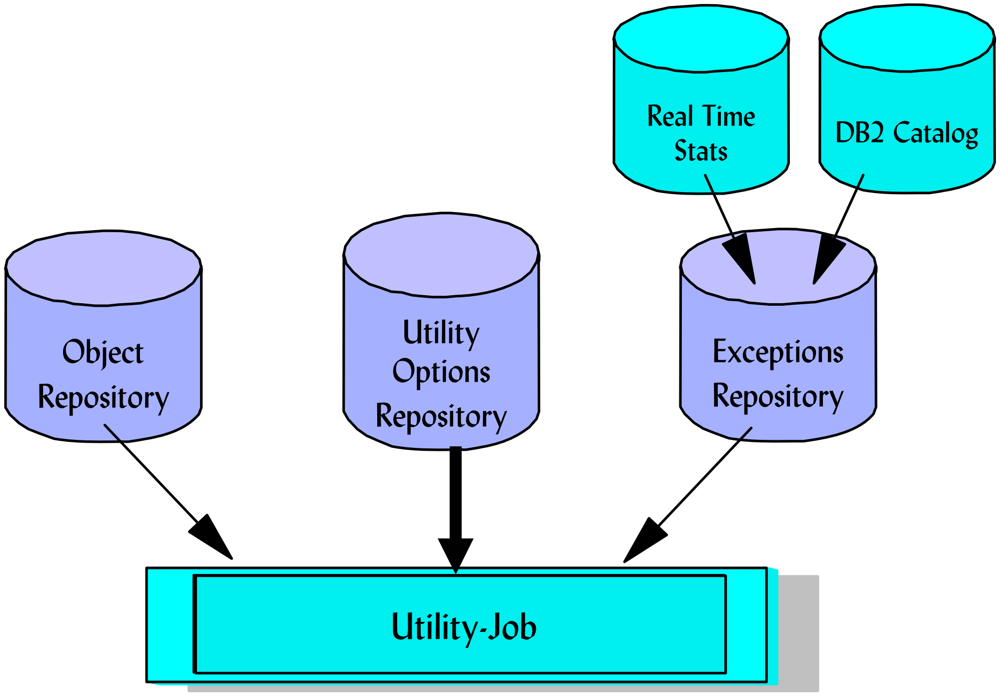
DAYS_SINCE_LAST	The number of days since the last REORG.	TABLESPACE STATS table: CURRENT DATE - REORGLASTTIME (in days)
INS_UPD_DEL	The number of inserts, updates, and deletes.	TABLESPACE STATS table: REORGINSERTS + REORGDELETES + REORGUPDATES
INS_UPD_DEL_PCT	The number of inserts, updates, and deletes as a percentage of the total number of rows.	TABLESPACE STATS table: [(REORGINSERTS + REORGDELETES + REORGUPDATES) * 100] / TOTALROWS

Specifying RTS values (not a complete list)

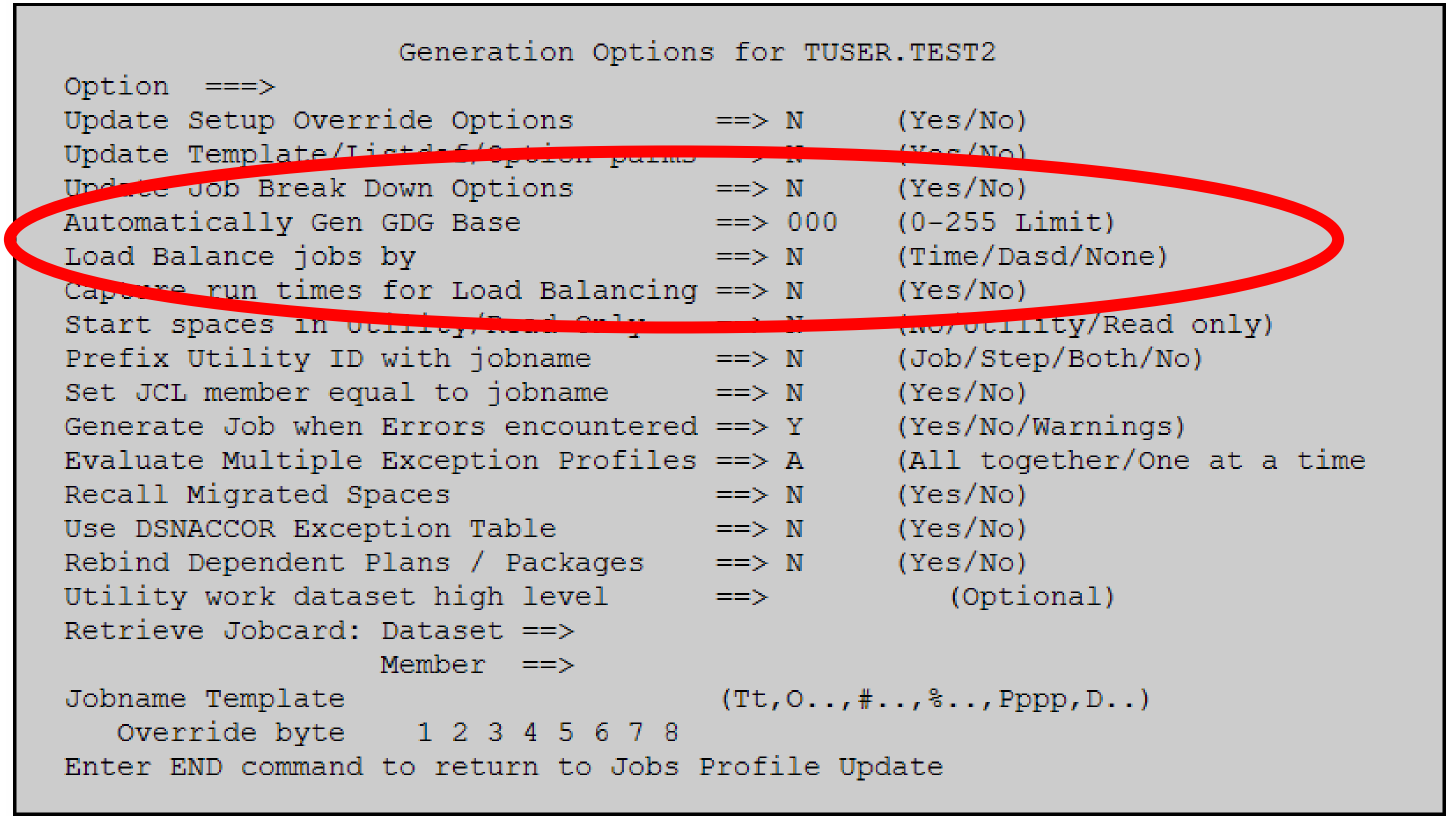


RELOCATED_ROWS	Total number of relocated rows.	TABLESPACE STATS table: REORGNEARINDREF + REORGFARINDREF
RELOCATED_ROWS_PCT	Total number of relocated rows as a percentage of the total number of rows.	TABLESPACE STATS table: (RELOCATED_ROWS * 100) / TOTALROWS
MASS_DELETES	The number of tables that have been mass deleted since the last REORG or LOAD REPLACE. If this value is not zero, the space may need a REORG.	TABLESPACE STATS table: REORGMASSDELETE

Now we put it all together to get the result



Generating the Job Profile – This is where you specify what object, utility and exception profiles are treated as a unit



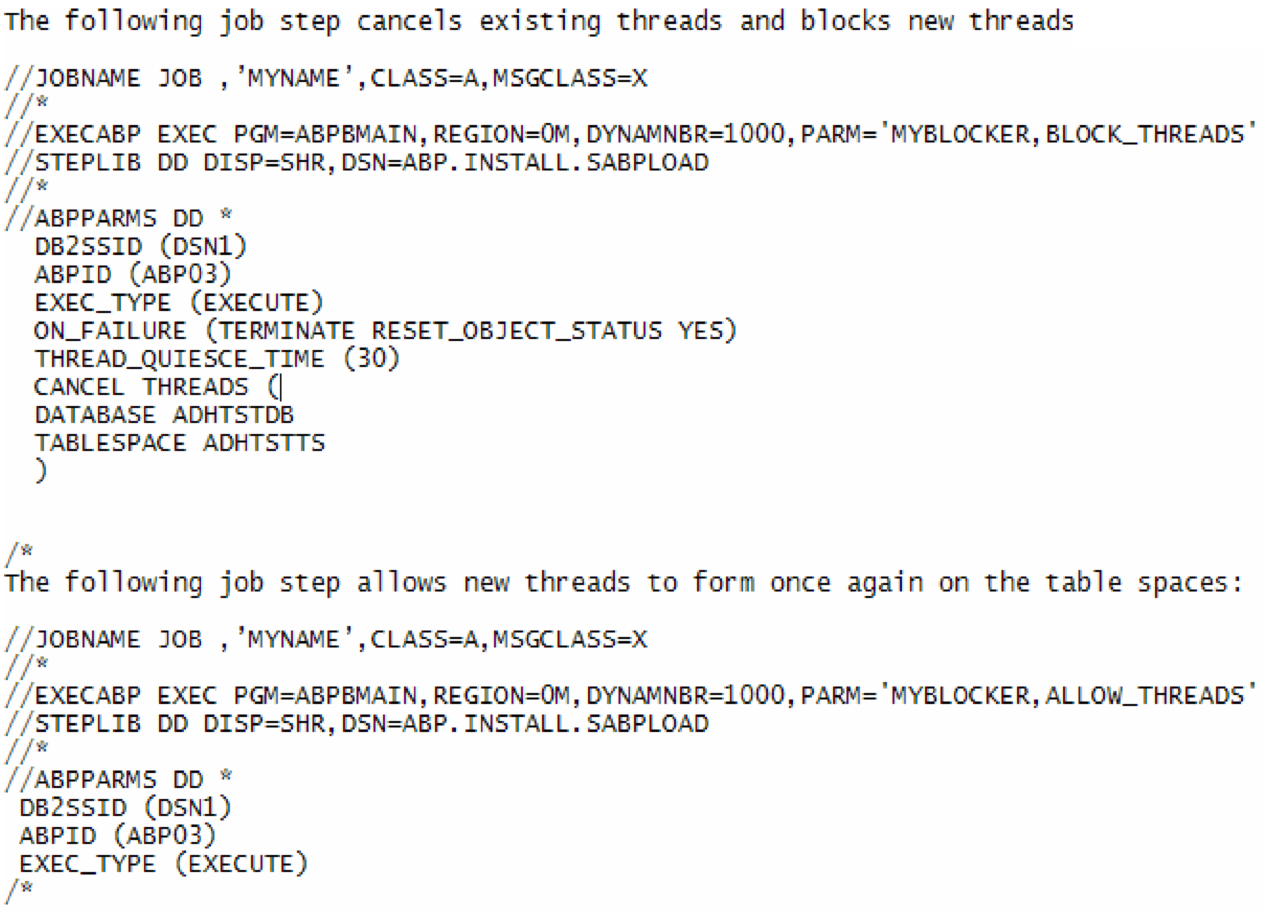
```
                   Generation Options for TUSER.TEST2
Option   ===>
Update Setup Override Options          ==> N      (Yes/No)
Update Template/Listdef/Option parms   ==> N      (Yes/No)
Update Job Break Down Options          ==> N      (Yes/No)
Automatically Gen GDG Base             ==> 000    (0-255 Limit)
Load Balance jobs by                   ==> N      (Time/Dasd/None)
Capture run times for Load Balancing ==> N        (Yes/No)
Start spaces in Utility/Read Only      ==> N      (No/Utility/Read only)
Prefix Utility ID with jobname         ==> N      (Job/Step/Both/No)
Set JCL member equal to jobname        ==> N      (Yes/No)
Generate Job when Errors encountered ==> Y        (Yes/No/Warnings)
Evaluate Multiple Exception Profiles ==> A        (All together/One at a time
Recall Migrated Spaces                 ==> N      (Yes/No)
Use DSNACCOR Exception Table           ==> N      (Yes/No)
Rebind Dependent Plans / Packages      ==> N      (Yes/No)
Utility work dataset high level        ==>           (Optional)
Retrieve Jobcard: Dataset ==>
                  Member  ==>
Jobname Template                          (Tt,O..,#..,%..,Pppp,D..)
   Override byte     1 2 3 4 5 6 7 8
Enter END command to return to Jobs Profile Update
```

Prior to this panel, you'll see panels similar to what we've already seen where you select the various profiles.

Except we need to guarantee the job will run

- But the reorg can’t break in threads accessing the object (which could just be some uncommitted readers who were some TSO users who didn’t log off)
- So, we use Thread Expert to cancel all threads (and insures the success of the utility!)

Canceling threads – Once Automation Tool generates the REORG job. We can manually insert the following two steps



The following job step cancels existing threads and blocks new threads

```
//JOBNAME JOB ,'MYNAME',CLASS=A,MSGCLASS=X
//*
//EXECABP EXEC PGM=ABPBMAIN,REGION=0M,DYNAMNBR=1000,PARM='MYBLOCKER,BLOCK_THREADS'
//STEPLIB DD DISP=SHR,DSN=ABP.INSTALL.SABPLOAD
//*
//ABPPARMS DD *
  DB2SSID (DSN1)
  ABPID (ABP03)
  EXEC_TYPE (EXECUTE)
  ON_FAILURE (TERMINATE RESET_OBJECT_STATUS YES)
  THREAD_QUIESCE_TIME (30)
  CANCEL THREADS (
  DATABASE ADHTSTDB
  TABLESPACE ADHTSTTS
  )


/*
```

The following job step allows new threads to form once again on the table spaces:

```
//JOBNAME JOB ,'MYNAME',CLASS=A,MSGCLASS=X
//*
//EXECABP EXEC PGM=ABPBMAIN,REGION=0M,DYNAMNBR=1000,PARM='MYBLOCKER,ALLOW_THREADS'
//STEPLIB DD DISP=SHR,DSN=ABP.INSTALL.SABPLOAD
//*
//ABPPARMS DD *
 DB2SSID (DSN1)
 ABPID (ABP03)
 EXEC_TYPE (EXECUTE)
/*
```

End of scenario

- We started with just

```
REORG TABLESPACE ADHTSTDB.ADHTSTTS
REORG TABLESPACE ADHTSTDB.ADHTSTTT
```

- Then, we added best practices, like

```
LOG NO, KEEPDICTIONARY, SORTDATA, NOSYSREC, SORTKEYS,
STATISTICS TABLE (T1) SAMPLE 25 COLUMN (C1, C2)…
```

- Then we saw how we could programmatically run the utilities using stored procedures triggered by statistics
- Then we were shown an alternative way to do this using the DB2 Automation Tool
- And lastly, we saw how the DB2 Thread Expert can help ensure the success of jobs, like DB2 Utilities

- One last thing that will look a little different with the DB2 Utilities in V8…

New DB2 accounting trace fields: IIP and IIPCP Time

```
 LOCATION: DSND81B          OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V3)
      GROUP: N/P                       ACCOUNTING REPORT - LONG
     MEMBER: N/P
  SUBSYSTEM: D81B                           ORDER: PLANNAME
DB2 VERSION: V8                             SCOPE: MEMBER

PLANNAME: DSNUTIL

AVERAGE       APPL(CL.1)  DB2 (CL.2)  IFI (CL.5)
------------  ----------  ----------  ----------
ELAPSED TIME  1:44.25600   53.465083         N/P
 NONNESTED    1:44.25600   53.465083         N/A
 STORED PROC    0.000000    0.000000         N/A
 UDF            0.000000    0.000000         N/A
 TRIGGER        0.000000    0.000000         N/A

CP CPU TIME    48.979165   18.069021         N/P
 AGENT         12.694310   10.249968         N/A
  NONNESTED    12.694310   10.249968         N/P
  STORED PRC    0.000000    0.000000         N/A
  UDF           0.000000    0.000000         N/A
  TRIGGER       0.000000    0.000000         N/A
 PAR.TASKS     36.284855    7.819053         N/A

 IIPCP CPU      3.432845         N/A         N/A

IIP CPU TIME   23.411692   23.411692         N/A
```

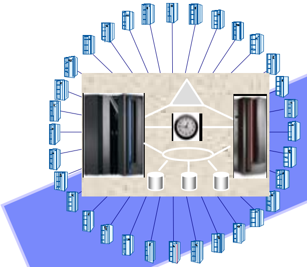
Work offloaded to zIIPs

zIIP-eligible on CP

zIIP (IBM System z9 Integrated Information Processor) -- 2006

Technology Evolution with Mainframe Specialty Engines

- **Building on a strong track record of technology innovation with specialty engines, IBM intends to introduce the System z9 Integrated Information Processor**

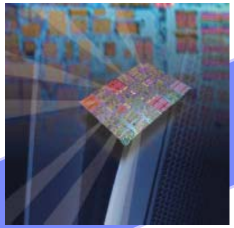


Centralized data sharing across mainframes

Internal Coupling Facility (ICF) 1997

Integrated Facility for Linux (IFL) 2001

- Support for new workloads and open standards



System z9 Application Assist Processor (zAAP) 2004

- Designed to help improve resource optimization for z/OS JAVA-based workloads

IBM System z9 Integrated Information Processor (IBM zIIP) planned for 2006

- Designed to help improve resource optimization for eligible data workloads within the enterprise
- DB2 exploitation - for mission critical ERP, CRM, and Data Warehousing workloads

IBM zIIP Technical Details

- New specialty engine for the System z9 mainframe (planned for 2006)
  - Priced similar to other speciality engines
  - Does not affect IBM software license charges
  - zIIPs per z9-109 not to exceed number of standard processors
  - z/OS manages/directs work between general purpose proc and zIIPs
    - zIIP is designed to have all or a portion of its enclave Service Request Block (SRB) work directed to the zIIP
    - The enclave SRB interface is available upon request
  - DB2 will be first exploiter of the zIIP with:
    - System z9 109
    - z/OS 1.6 or later
    - DB2 for z/OS V8 with APARs PK18454, PK19920, and PK19921
- No changes required for DB2 for z/OS V8 applications
- Minimal overhead anticipated with redirect to zIIP
  - Work is dispatched to zIIP the same way as on a general purpose proc

Types of DB2 for z/OS workloads that may benefit from zIIP

1 -ERP or CRM application serving

For applications, running on z/OS, UNIX, Linux, Intel, or Linux on System z, that access DB2 for z/OS V8 on a System z9 109, via DRDA over a TCP/IP connection DB2 gives z/OS the necessary information to have portions of these SQL requests directed to the zIIP

2 - Data warehousing applications

Requests that utilize DB2 for z/OS V8 star schema parallel queries may have portions of these SQL requests directed to the zIIP when DB2 gives z/OS the necessary information

3 – Some DB2 for z/OS V8 utilities

A portion of DB2 utility functions used to maintain index maintenance structures (example LOAD, REORG, and REBUILD INDEX) typically run during batch, can be redirected to zIIP.

Summary

- There are performance, efficiency, and availability best practices for your utilities
- There are ease of use techniques to invoke and control them
  - LISTDEFs / TEMPLATEs
  - As a stored procedure
- Automation reduces your effort
  - Mange by exception
  - Set it and forget it
- Synergy between different solutions
- IBM System z9 Integrated Information Processors
  - Reduce cost of ownership
  - Coming soon

Q&A