Redbooks

Utility Enhancements

List of Topics

New utilities BACKUP SYSTEM and RESTORE SYSTEM

Delimited data support for LOAD and UNLOAD

RUNSTATS enhancements

Defaults for better performance

REORG TABLESPACE enhancements

REBUILD INDEX enhancements

COPY enhancements

REPAIR enhancements

Changes to utilities to support online schema evolution

Offline utility (DSN1*) enhancements

Unicode utility statements

System Level Point-in-Time Recovery

Easier, more flexible, less disruptive, faster backup and recovery

Handle large numbers of table spaces and indexes

Two new utilities are introduced

- BACKUP SYSTEM for fast volume-level backups
  - DB2 databases and logs
  - Data sharing group scope
  - z/OS V1R5 required for new COPYPOOL function and fast replication
- RESTORE SYSTEM
  - To an arbitrary point-in-time
  - Handles CREATEs, DROPs, LOG NO events
  - Data sharing group scope

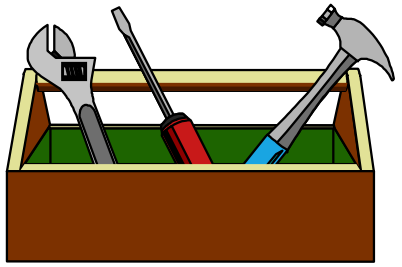
LOAD / UNLOAD Delimited Input / Output

Redbooks

LOAD / UNLOAD utilities will accept / produce delimited files

Benefits of these enhancements include:

- Eases the import / export of large amounts of data from DB2 for z/OS to other operating system platforms and vice versa
- Eliminates the requirement to write a program to convert non-z/OS platform data into the positional format for the DB2 for z/OS LOAD utility, or to use INSERT processing
- Unloads data from DB2 for z/OS in delimited file format and loads / imports it into another RDBMS



Delimited Files - Reminder

A delimited file is a sequential file with row and column delimiters

Is a string of characters consisting of cell values ordered by row, then by column

Row (Record) delimiters not needed since the end of record is inherent in the file structure

Columns are separated by column delimiters

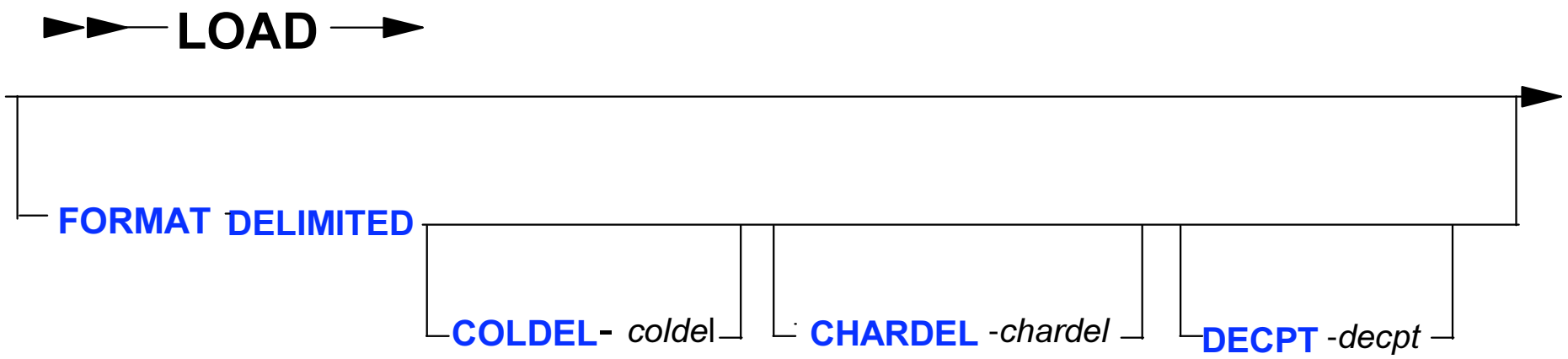
Character strings are delimited by character delimiters

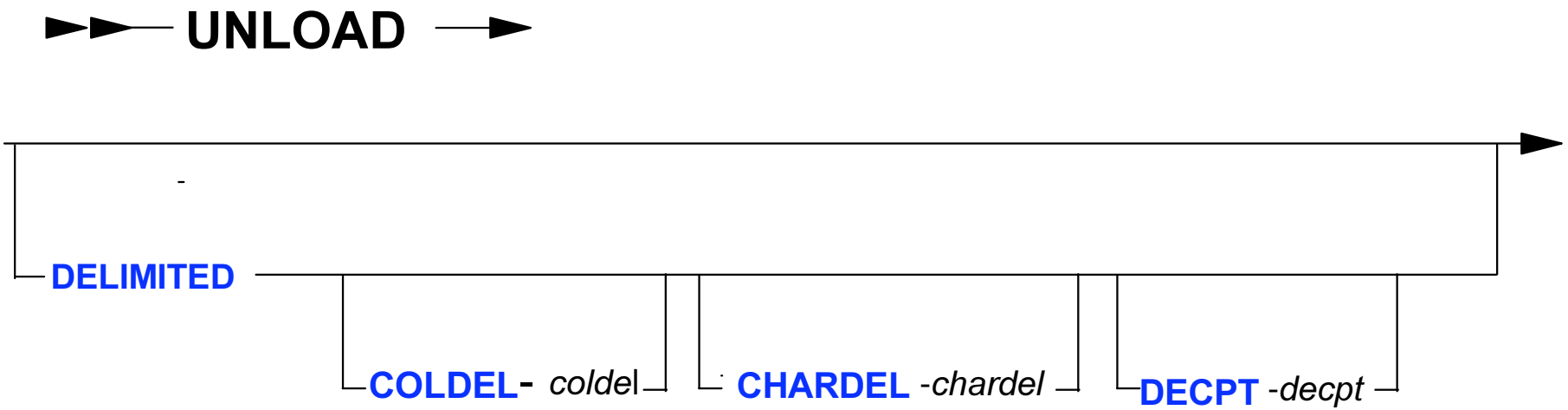
In z/OS, a row is a single BSAM record

Examples: "Bart", "Steegmans", "ITSO"

"Ravi", "Kumar", "ITES"

LOAD/UNLOAD Delimited Input/Output Syntax





Example of UNLOAD Statement


```
UNLOAD TABLESPACE   databasename.tablespacename
   DELIMITED CHARDEL '#' COLDEL ';'  DECPT '.'
   PUNCHDDN SYSPUNCH
   UNLDDN SYSREC EBCDIC
   FROM TABLE    tablename

   (LNAME          POSITION(*) VARCHAR(15),
   DEPTNO          POSITION(*) CHAR(8),
   SEX             POSITION(*) CHAR(1),
   COUNTRY         POSITION(*) DBCLOB(11),
   SALARY          POSITION(*) DECIMAL(8,2),
   SALARYRATE      POSITION(*) FLOAT )
```

Optional keywords

POSITION(*) keywords and char field lengths optional

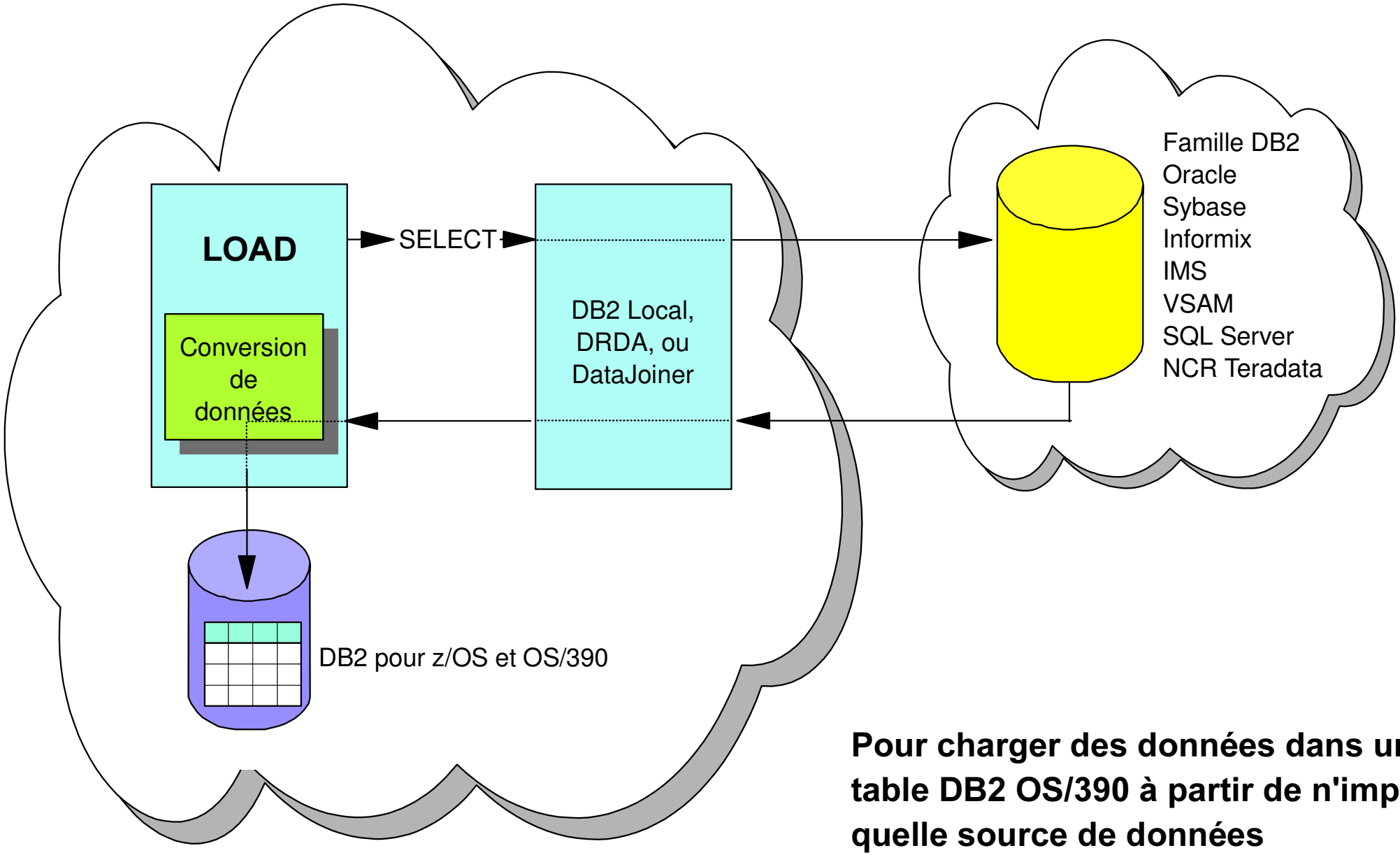
No trailing blanks for VARCHAR

Unloaded data looks like:

#warren#; #D10 #; #M#; # U S A #; #6500.00#; #.5E+1#

Note that field lists are optional for LOAD / UNLOAD and are primarily used for selecting a subset of columns or selecting columns in a different order

Fonction DB2 Family Cross-Loader



Pour charger des données dans une table DB2 OS/390 à partir de n'import quelle source de données

Chargement à l'aide d'un Curseur


```
EXEC SQL
  DECLARE C1 CURSOR FOR SELECT * FROM DSN8710.EMP
  WHERE SALARY > 10000
ENDEXEC

LOAD DATA INCURSOR(C1)
          REPLACE
          INTO TABLE DAVID.MYEMP
```

Sur un site éloigné, utilisation de la procédure stockée DSNUTILS

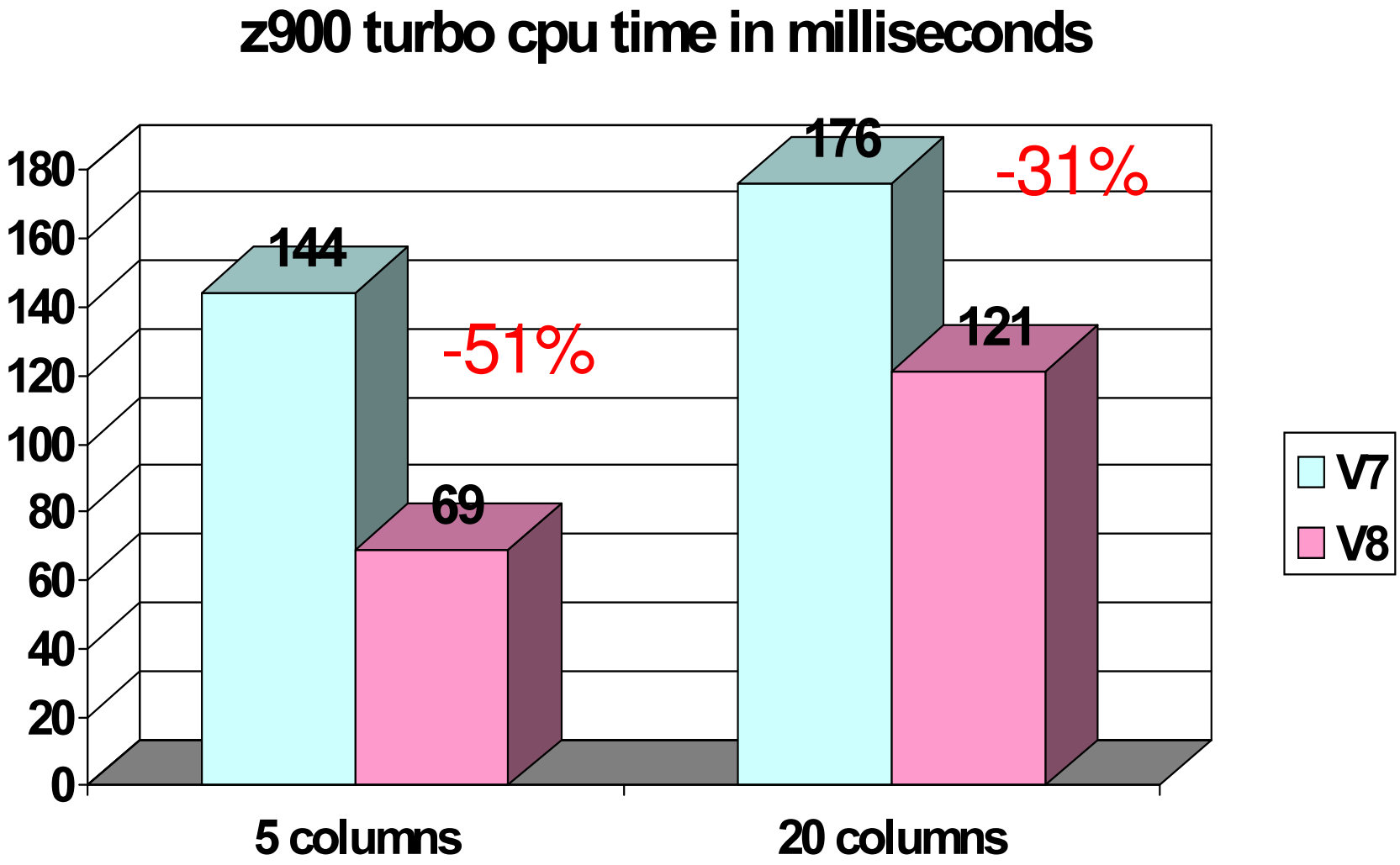
Performance Enhancements

- Cross Loader two times faster V7 PQ84160 8/04
  - Now roughly even with Unload and Load
- Up to 65000 open data sets
  - By moving dataset control blocks above 16MB line with z/OS1.6
  - Could dramatically reduce Open/Close frequency by keeping frequently used datasets open
  - Compression dictionary above 2GB in V8 can really help

Automatic exploitation of multi-row Operation

- DSNTEP4=DSNTEP2 with automatic multi-row fetch
  - Up to 35% cpu reduction in fetching 10000 rows with 5 and 20 columns
- DSNTIAUL (sample Unload utility)
  - Up to 50% cpu reduction in fetching 10000 rows with 5 and 20 columns

DSNTIAUL fetching 10000 rows



RUNSTATS Enhancements

Non-uniform distribution statistics on non-indexed columns

Ability to update statistics history tables with the latest information without updating the statistics used by the optimizer

- Facilitates monitoring and analysis
- RUNSTATS TABLESPACE DB1.TS1
  UPDATE NONE HISTORY ALL

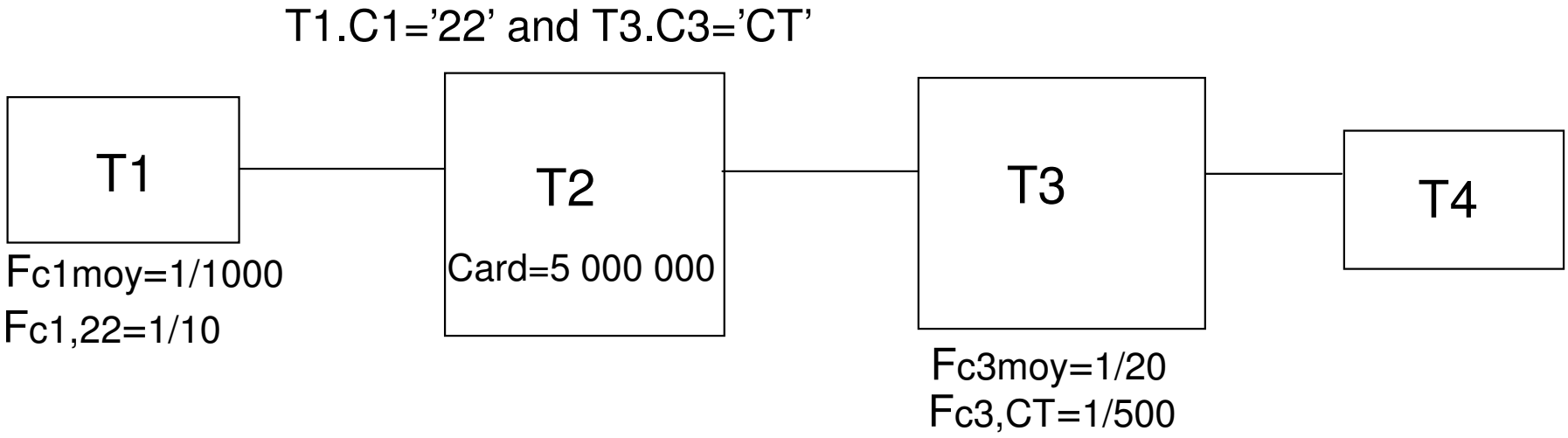
RUNSTATS with UPDATE NONE REPORT NO to invalidate dynamic SQL cache without collecting any statistics

The Need for Extra Statistics

Distribution statistics are currently collected for indexed columns only (if ... FREQVAL NUMCOLS(1) COUNT(10))

Non-uniform distribution statistics for non-leading indexed columns are not collected by RUNSTATS, which can result in non-optimal performance

- Less efficient join sequences
- Inappropriate table join method
- Increase in the number of rows that need to be processed



Distributions non uniformes (en V 7)

DSTATS permet de collecter les distributions non uniformes de colonnes quelconques.

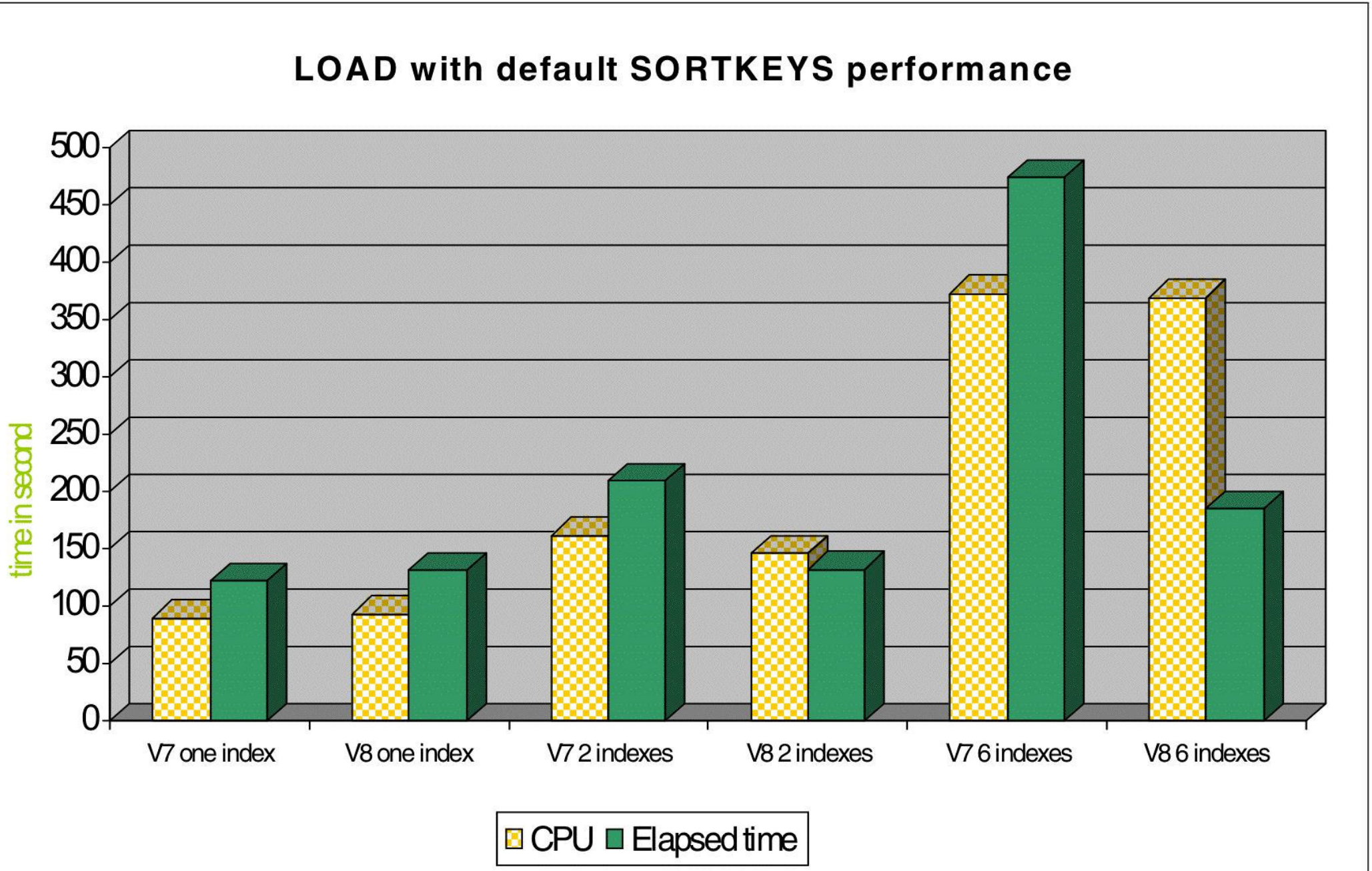
Download depuis : : http://www-1.ibm.com/support/docview.wss?uid=swg240015

Paramètres du DSTATS

```
//SYSIN    DD *
VALUES 99,50
COLCARDF1 quick
creator.TAB1.col1
creator.TAB1.coll2
creator.TAB2.col5
```

New Defaults for Better Performance

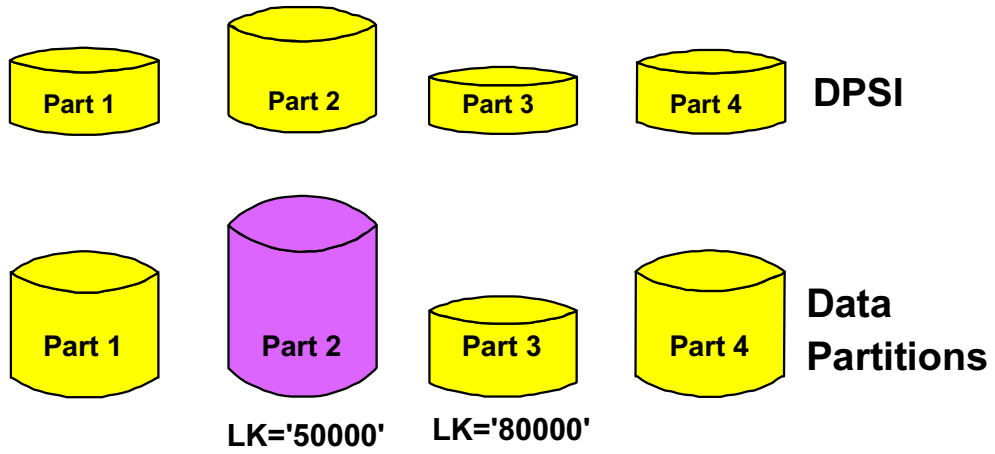
- RESTART is new default for Utilities (also in V7 with PQ72337)
- SORTKEYS is used by default for LOAD, REORG, and REBUILD
- SORTDATA is used by default for REORG
  - SORTDATA now allowed for 32K records with DFSORT
- REORG will use implicit clustering index
  - if no clustering index, first index defined is used
  - If table space has no indexes, SORTDATA operates as in pre-V8 releases



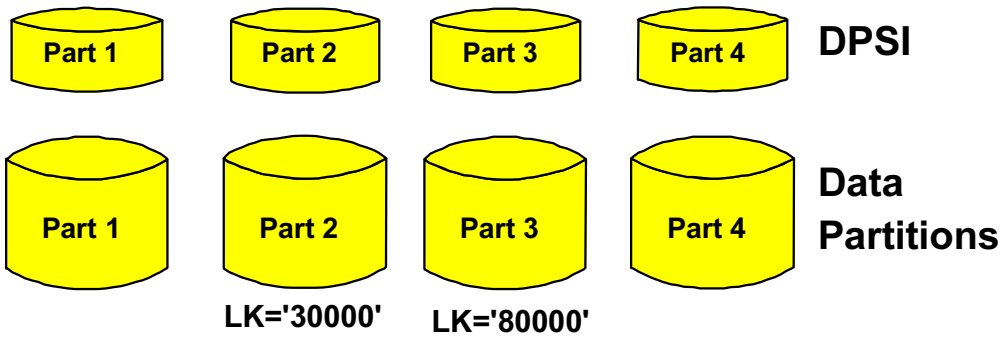
REORG REBALANCE

Sets new partition boundaries for even distribution of rows across the partitions being reorganized

Before REORG TABLESPACE



After REORG TABLESPACE ... REBALANCE



What Does Rebalance Do?

1. **Unload rows from the table space or partition range**
2. **Sort rows by partitioning column(s) and divide by number of parts**
   - Is not perfect if lots of duplicate keys exist
3. **Reload the data**
4. **Update limit key values in the catalog**
5. **Invalidate plans, packages and dynamic statement cache**

When clustering does not match partitioning, REORG must be run twice:

- First to move rows to the right partition
- Second to sort in clustering sequence

REORG Rebalance & ALTER + REORG Comparison

REORG TABLESPACE ... REBALANCE works out new partitioning values for you

- One step automated process
- Supported with SHRLEVEL REFERENCE and NONE
- Data available (read only) almost all of the time
- Not supported for table spaces with LOB columns

ALTER INDEX or ALTER TABLE ALTER PART

- Gives you more control to allow for future skewed growth
- Leaves affected partitions in REORP
- Data unavailable until REORG completes

REORG TABLESPACE - SCOPE PENDING

Redbooks

Reorganizes only the table space part(s) that are in REORG-pending (REORP) or advisory REORG-pending state (AREO*)

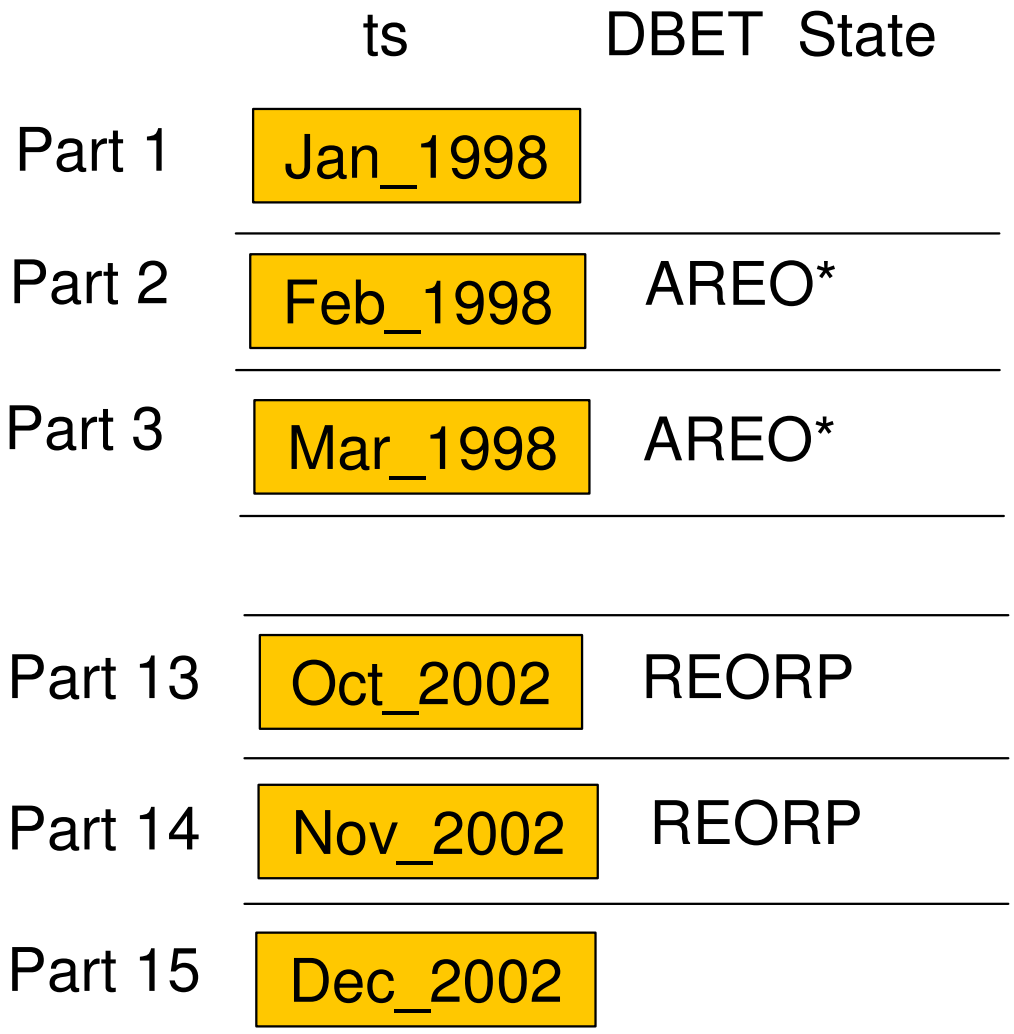
When specifying a partition range, the adjacent high and low parts that are not included in the range must not be in REORP

SYSCOPY records are only written for those partitions that are actually reorganized

REORG TABLESPACE - SCOPE PENDING Example 1

Redbooks

REORG TABLESPACE ... SCOPE PENDING



Parts 2,3,13,14 will be REORG'd

REORG TABLESPACE ... DISCARD

Now supported by REORG ... SHRLEVEL CHANGE

During the window when discarding the data rows in REORG, these data rows cannot be modified

If a data row that matches the discard criteria gets updated while REORG is in the process of discarding, REORG stops with RC=8 and a DSNU1127I message

REBUILD INDEX - SCOPE PENDING

Rebuilds only the index or part(s) that are in rebuild-pending (RBDP), recovery pending (RECP), or advisory REORG pending (AREO*) state

Unlike REORG TABLESPACE, the adjacent high and low parts not included in the range are not checked for RBDP

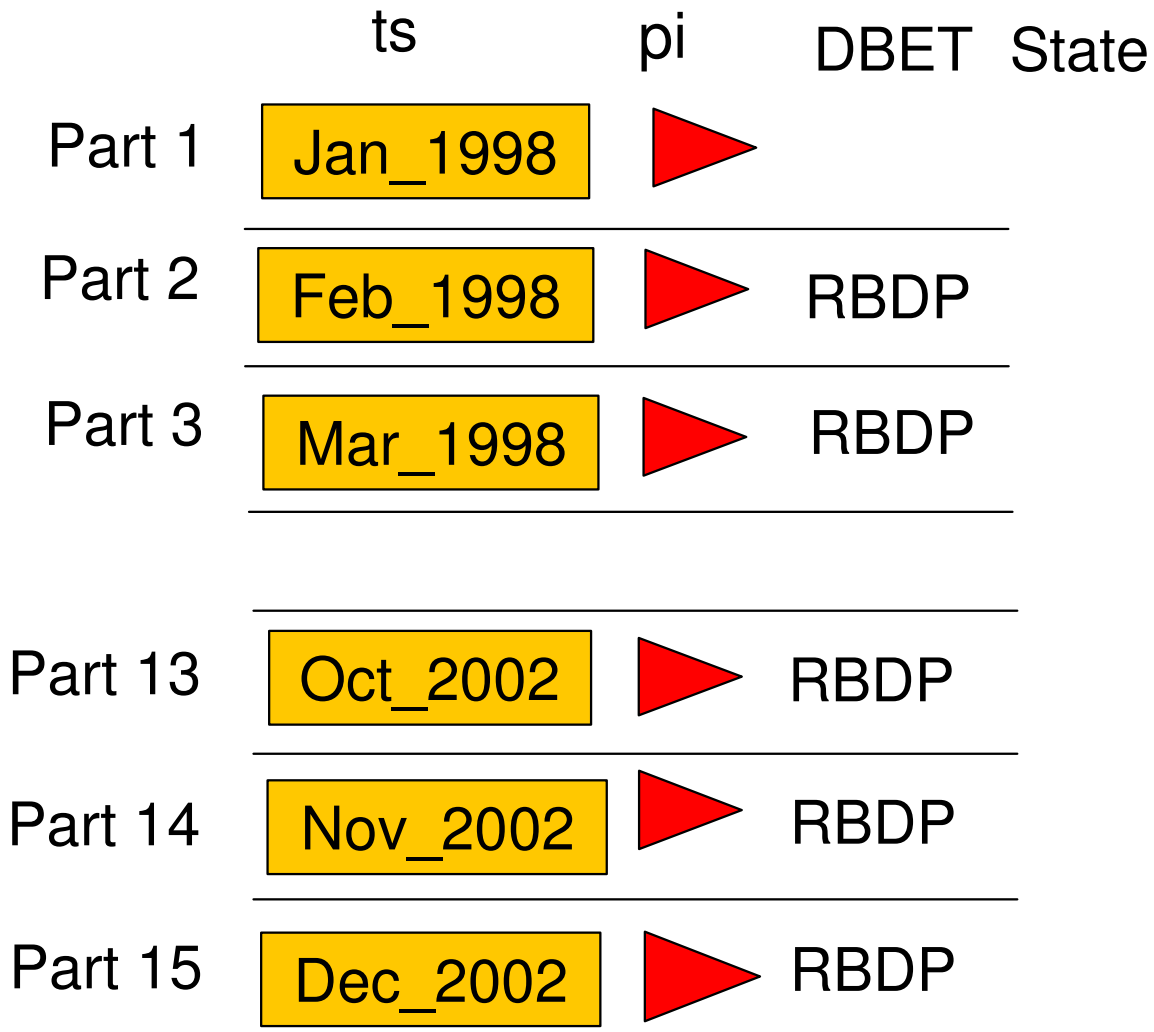
You can specify the index space name, instead of the index name in the following utilities:

- REBUILD INDEX ... or REBUILD **INDEXSPACE** ...
- REORG INDEX ... or REORG **INDEXSPACE**...
- RECOVER INDEX ... or RECOVER **INDEXSPACE** ...

REBUILD INDEX - SCOPE PENDING Example

Redbooks

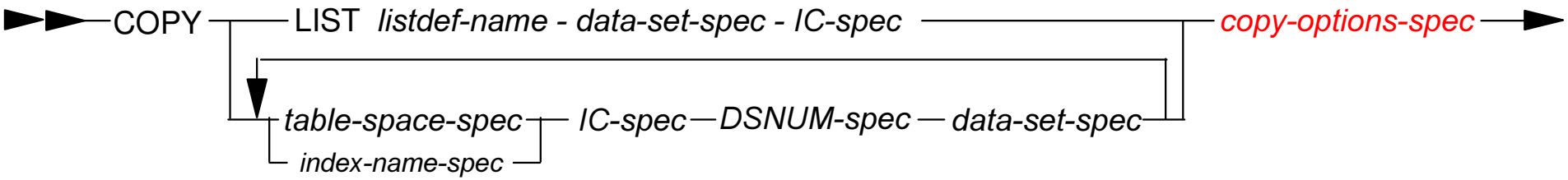
REBUILD INDEX ... SCOPE PENDING PART 2:14



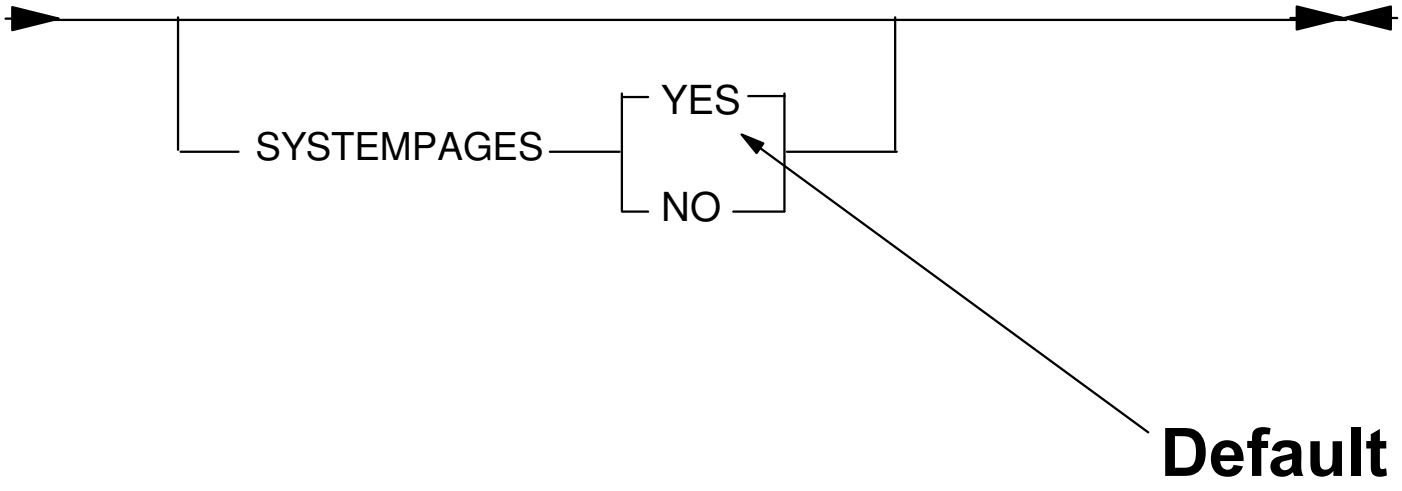
REBUILD

Parts 2, 3, 13, 14

COPY Enhancements



copy-options-spec:



COPY Utility SYSTEMPAGES Option

Indicates whether the dictionary and version system pages are copied at the beginning of the object to be copied. Especially important when

- Copying a piece, or a single data set of a table space or index
- Using incremental image copy and those system pages have not changed

With SYSTEMPAGES YES

- Dictionary pages for the compression dictionary are included
- V8 system pages that contain version information are included
- Both are included at the beginning of the image copy
- Version pages can occur multiple times in the image copy

With SYSTEMPAGES NO

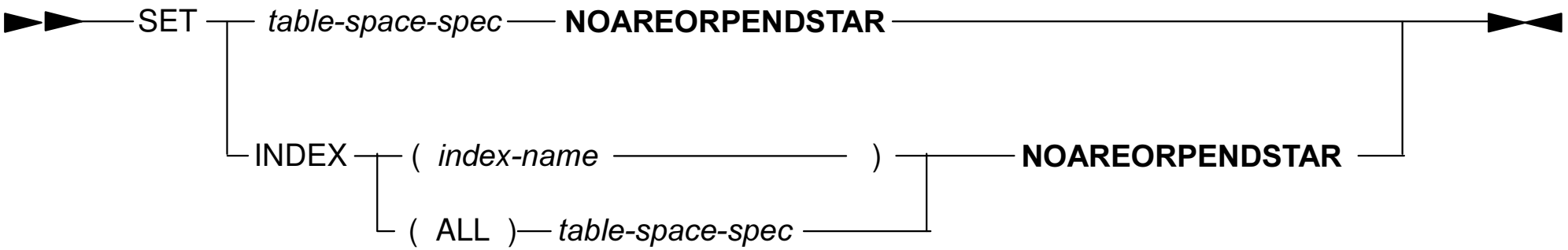
- Copy pages as they appear in the object (pre-V8)

Header page is always included no matter what SYSTEMPAGES option is used

Note that for SYSTEMPAGES YES, the UNLOAD utility can process image copies with data versioning

REPAIR - Switch Off New Status

To switch off the new AREO* status:



REPAIR SET TABLESPACE DELIMITD.DELIMITS **NOAREORPENDSTAR**

Utility Changes to Support Informational RI Constraints

LOAD and CHECK DATA do not check informational RI constraints

REPORT TABLESPACESET also reports all table spaces related to the named table space through informational RI constraints

QUIESCE TABLESPACESET also quiesces all table spaces related to the named table space through informational RI constraints

LISTDEF also includes all table spaces related through informational RI constraints when the keyword RI is specified

Utility Changes to Support DPSI - 1/6

CHECK DATA

- When run against entire partitioned table space scans DPSI, extracts keys, and sorts keys
- When run against a partition scans partition of DPSI corresponding to table partition, extract keys, and skips sort

CHECK INDEX

- When run with PART keyword checks specified partition of DPSI

COPY

- Supports specifying a partition of DPSI with DSNUM keyword

LISTDEF -

- PARTLEVEL keyword specifies the partition granularity for partitioned objects has been extended to DPSIs in V8

TEMPLATE

- Templates created for DPSIs may wish to make use of the &PA OBJECT variable

QUIESCE

- Drain classes and restrictive states for DPSIs mirror PIs:
- WRITE YES: DPSI partitions are DW / UTRO
- WRITE NO: No drains or restrictive states on DPSI partitions

QUIESCE

- NPSIs are DW / UTRO during WRITE YES for table space/partition

Utility Changes to Support DPSI - 5/6

REBUILD INDEX

Recreates indexes / index partitions from the table / table partition that they reference

- REBUILD INDEX . . . PART
  - PI or DPSI: Recreates the physical partition
  - NPSI: Recreates the logical partition
- Multiple partitions of DPSI can be rebuilt in parallel
- Concurrency:
  - For a DPSI, each partition being rebuilt is DA / UTUT
  - For a NPSI, each (logical) partition being rebuilt is DR

Utility Changes to Support DPSI - 6/6

Redbooks

REORG ... PART

Reorganizes a table space partition (or a range), or indexspace partition

- REORG TABLESPACE PART n
- REORG TABLESPACE PART n:m
- REORG INDEX PART n

REORG phases affected:

- SORT and SORTBLD require more space if there is a mix of DPSIs and NPSIs
- BUILD -- REORG PART SHRLEVEL(NONE)
  - DPSIs are rebuilt -- no contention
  - NPSIs are corrected -- contention is possible
- BUILD2 only for NPSIs, not used with DPSIs

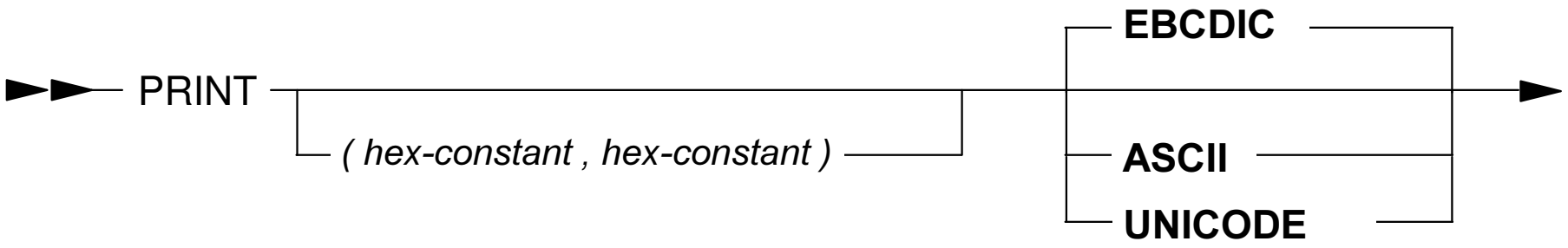
Stand-Alone Utility Changes

DSN1COMP

- Retrieve row "as is" instead of converting to current version

DSN1PRNT

- Recognize a system page and print in hex for the format option
- Basic support for ASCII or UNICODE data conversion



DSN1COPY

- Handling of version system pages
- CHECK option also checks system pages

Utility Unicode Statements

Utility control statements may be specified in Unicode or EBCDIC

- DB2 detects which encoding scheme is being used
- Must be all UTF-8 or EBCDIC -- no mixing!
- Object names in messages will be in EBCDIC

New utility stored procedure interface DSNUTILU for Unicode

DSNUTILU Stored Procedure

Identical to DSNUTILS except:

- Inputs are in Unicode
- UTILITY_NAME parameter dropped
- Data set DYNALLOC keywords dropped
  - Use TEMPLATE for all data sets