Real Time Statistics (RTS)

Svetlana Kacapor
GEICO
skacapor@geico.com

Agenda

- RTS Tables in IBM DB2 V8
- RTS Tables in IBM DB2 V9
- Column mappings to catalog tables and updating reorg related columns
- IBM suggested reorg criteria and our reports
- Use of RTS for identifying obsolete indexes
- Creating and populating history tables
- Use of RTS History tables for reporting

RTS Tables in IBM DB2 V8

- DSNRTSDB.DSNRTSTS
- TABLESPACESTATS

DBNAME
NAME
PARTITION
DBID
PSID
UPDATESTATSTIME — Timestamp of the last update
TOTALROWS — Updated after it is set by Reorg or Load replace
NACTIVE — Updated after it is set by Reorg or Load replace
SPACE — Updated after it is set by Reorg or Load replace
EXTENTS — Updated after it is set by Reorg or Load replace

LOADRLASTTIME
REORGLASTTIME
REORGINSERTS
REORGDELETES
REORGUPDATES
REORGUNCLUSTINS
REORGDISORGLOB
REORGMASSDELETE
REORGNEARINDREF
REORGFARINDREF

Reorg/Load Replace

STATSLASTTIME
STATSINSERTS
STATSDELETES
STATSUPDATES
STATSMASSDELETE

Runstats

COPYLASTTIME
COPYUPDATEDPAGES
COPYCHANGES
COPYUPDATELRSN
COPYUPDATETIME

Image Copy

RTS Tables in IBM DB2 V8 (cont.)

- INDEXSPACESTATS

DBNAME
INDEXSPACE
PARTITION
DBID
ISOBID
PSID
UPDATESTATSTIME
TOTALENTRIES
NLEVELS
NACTIVE
SPACE
EXTENTS

LOADRLASTTIME
REBUILDLASTTIME
REORGLASTTIME
REORGINSERTS
REORGDELETES
REORGAPPENDINSERT
REORGPSEUDODELETES
REORGMASSDELETE
REORGLEAFNEAR
REORGLEAFFAR
REORGNUMLEVELS

STATSLASTTIME
STATSINSERTS
STATSDELETES
STATSMASSDELETE

COPYLASTTIME
COPYUPDATEDPAGES
COPYCHANGES
COPYUPDATELRSN
COPYUPDATETIME

RTS Tables in IBM DB2 V9

- DSNDB06.SYSRTSTS
- SYSTABLESPACESTATS

UPDATESTATSTIME
NACTIVE
NPAGES
EXTENTS
LOADRLASTTIME COPYLASTTIME
COPYUPDATEDPAGES
COPYCHANGES
COPYUPDATELRSN
COPYUPDATETIME
IBMREQD
DBID
PSID
PARTITION
INSTANCE
SPACE
TOTALROWS
DATASIZE
UNCOMPRESSEDDATASIZE
DBNAME
REORGLASTTIME
REORGINSERTS
REORGDELETES
REORGUPDATES
REORGUNCLUSTINS
REORGDISORGLOB
REORGMASSDELETE
REORGNEARINDREF
REORGFARINDREF
STATSLASTTIME
STATSINSERTS
STATSDELETES
STATSUPDATES
STATSMASSDELETE
NAME

RTS Tables in IBM DB2 V9 (cont.)

- SYSINDEXSPACESTATS

UPDATESTATSTIME
NLEVELS
NPAGES
NLEAF
NACTIVE
SPACE
EXTENTS
LOADRLASTTIME
REBUILDLASTTIME
REORGLASTTIME
REORGINSERTS
REORGDELETES
REORGAPPENDINSERT
REORGPSEUDODELETES
REORGMASSDELETE
REORGLEAFNEAR
REORGLEAFFAR
REORGNUMLEVELS
STATSLASTTIME
STATSINSERTS
STATSDELETES
STATSMASSDELETE
COPYLASTTIME
COPYUPDATEDPAGES
COPYCHANGES
COPYUPDATELRSN
COPYUPDATETIME
LASTUSED
IBMREQD
DBID
ISOBID
PSID
PARTITION
INSTANCE
TOTALENTRIES
DBNAME
NAME
CREATOR
INDEXSPACE

Column Mappings and Update

```
SELECT DBNAME
     ,NAME
     ,DBID
     ,PSID
     ,PARTITION
     ,INSTANCE
     ,STATSLASTTIME - x DAYS AS REORGLASTTIME
     ,STATSINSERTS          AS REORGINSERTS
     ,STATSDELETES          AS REORGDELETES
     ,STATSUPDATES          AS REORGUPDATES
 ,(SELECT INT(SPACEF)
     FROM SYSIBM.SYSTABLEPART TP
    WHERE TP.DBNAME    = RTS.DBNAME
      AND TP.TSNAME    = RTS.NAME
      AND TP.PARTITION = RTS.PARTITION
  )                   AS SPACEF
 ,(SELECT EXTENTS
     FROM SYSIBM.SYSTABLEPART TP
    WHERE TP.DBNAME    = RTS.DBNAME
      AND TP.TSNAME    = RTS.NAME
      AND TP.PARTITION = RTS.PARTITION
  )                   AS EXTENTS
 ,INT(COALESCE (
 (SELECT MAX(FAROFFPOSF)
   FROM SYSIBM.SYSINDEXPART IP
     JOIN SYSIBM.SYSINDEXES IX
ON IP.IXNAME     = IX.NAME
    AND IP.IXCREATOR  = IX.CREATOR
    AND IX.DBNAME     = RTS.DBNAME
    AND IX.CLUSTERING = 'Y'
    AND IP.PARTITION  = RTS.PARTITION
    JOIN SYSIBM.SYSTABLES TB
      ON TB.CREATOR    = IX.TBCREATOR
     AND TB.NAME       = IX.TBNAME
     AND TB.TSNAME     = RTS.NAME
  ),
```

Column Mappings and Update (cont.)

```
 (SELECT COALESCE(MIN(FAROFFPOSF),-1)
  FROM SYSIBM.SYSINDEXPART IP
    JOIN SYSIBM.SYSINDEXES IX
     ON IP.IXNAME     = IX.NAME
    AND IP.IXCREATOR  = IX.CREATOR
    AND IX.DBNAME     = RTS.DBNAME
    AND IP.PARTITION  = RTS.PARTITION
   JOIN SYSIBM.SYSTABLES TB
     ON TB.CREATOR    = IX.TBCREATOR
    AND TB.NAME       = IX.TBNAME
    AND TB.TSNAME     = RTS.NAME
  )
 ))                   AS REORGUNCLUSTINS
,COALESCE(STATSMASSDELETE,0) AS REORGMASSDELETE
,(SELECT INT(COALESCE(NEARINDREF,0))
  FROM SYSIBM.SYSTABLEPART TP
  WHERE TP.DBNAME    = RTS.DBNAME
    AND TP.TSNAME    = RTS.NAME
    AND TP.PARTITION = RTS.PARTITION
 )                    AS REORGNEARINDREF
,(SELECT INT(COALESCE(FARINDREF,0))
  FROM SYSIBM.SYSTABLEPART TP
  WHERE TP.DBNAME    = RTS.DBNAME
    AND TP.TSNAME    = RTS.NAME
    AND TP.PARTITION = RTS.PARTITION
 )                    AS REORGFARINDREF
     ,(SELECT BIGINT(COALESCE(CARDF,0))
       FROM SYSIBM.SYSTABLEPART TP
       WHERE TP.DBNAME    = RTS.DBNAME
         AND TP.TSNAME    = RTS.NAME
         AND TP.PARTITION = RTS.PARTITION
      )                   AS TOTALROWS
```

Column Mappings and Update (cont.)

```
FROM SYSIBM.SYSTABLESPACESTATS RTS
WHERE (   RTS.REORGLASTTIME IS NULL
      OR RTS.REORGLASTTIME IS NOT NULL
      AND EXISTS (SELECT 1
                  FROM SYSIBM.SYSCOPY CP
 WHERE CP.DBNAME = RTS.DBNAME
                    AND CP.TSNAME = RTS.NAME
                    AND CP.ICTYPE = 'W'
                    AND CP.TIMESTAMP > RTS.REORGLASTTIME
                    )
    )
  AND RTS.STATSLASTTIME IS NOT NULL
  AND RTS.STATSLASTTIME > CURRENT TIMESTAMP – X DAYS
  AND NOT EXISTS (SELECT 1
                  FROM SYSIBM.SYSCOPY CP
                  WHERE CP.DBNAME = RTS.DBNAME
                    AND CP.TSNAME = RTS.NAME
                    AND CP.ICTYPE = 'W'
                    AND CP.TIMESTAMP > RTS.STATSLASTTIME
  AND RTS.DBNAME          IN ('XXXXXDB','XXXXXYDB')
  AND RTS.NAME            IN ('XXXXXXTS','XXXXXYTS')
ORDER BY RTS.DBNAME
       ,RTS.NAME
       ,RTS.PARTITION
WITH UR
;
```

Reorg Criteria

- IBM suggested reorg criteria for tablespaces

(((*QueryType*='REORG' OR *QueryType*='ALL') AND
(*ObjectType*='TS' OR *ObjectType*='ALL')) AND
(REORGLASTTIME IS NULL AND LOADRLASTTIME IS NULL) OR
(NACTIVE IS NULL OR NACTIVE > 5) AND
((((REORGINSERTS×100)/TOTALROWS>*RRTInsertPct*) AND
REORGINSERTS>*RRTInsertAbs*) OR
(((REORGDELETE×100)/TOTALROWS>*RRTDeletePct*) AND
REORGDELETE>*RRTDeleteAbs*) OR
(REORGUNCLUSTINS×100)/TOTALROWS>*RRTUnclustInsPct* OR
(REORGDISORGLOB×100)/TOTALROWS>*RRTDisorgLOBPct* OR
(SPACE×1024)/DATASIZE>*RRTDataSpaceRat* OR
((REORGNEARINDREF+REORGFARINDREF)×100)/TOTALROWS>*RRTIndRefLimit* OR
REORGMASSDELETE>*RRTMassDelLimit* OR
EXTENTS>ExtentLimit)) OR
((*QueryType*='RESTRICT' OR*QueryType*='ALL') AND
(*ObjectType*='TS' OR *ObjectType*='ALL') AND
The table space is in advisory or informational reorg pending status))

Figure 16. DSNACCOX formula for recommending a REORG on a table space (Performance Monitoring and Tuning Guide)

Reorg Criteria (cont.)

- IBM suggested reorg criteria for indexes

(((*QueryType*='REORG' OR *QueryType*='ALL') AND
(*ObjectType*='IX' OR *ObjectType*='ALL') AND
(REORGLASTTIME IS NULL AND REBUILDLASTTIME IS NULL) OR
(NACTIVE IS NULL OR NACTIVE > 5) AND
((((REORGINSERTS×100)/TOTALENTRIES>*RRIInsertPct*) AND
REORGINSERTS>*RRIInsertAbs*) OR
(((REORGDELETE×100)/TOTALENTRIES>RRIDeletePct) AND
REORGDELETE>*RRIDeleteAbs*) OR
(REORGAPPENDINSERT×100)/TOTALENTRIES>*RRIAppendInsertPct* OR
(REORGPSEUDODELETES×100)/TOTALENTRIES>*RRIPseudoDeletePct* OR
REORGMASSDELETE>*RRIMassDeleteLimit* OR
(REORGLEAFFAR×100)/NACTIVE>*RRILeafLimit* OR
REORGNUMLEVELS>*RRINumLevelsLimit* OR
EXTENTS>*ExtentLimit*)) OR
((*QueryType*='RESTRICT' OR *QueryType*='ALL') AND
(*ObjectType*='IX' OR *ObjectType*='ALL') AND
An index is in advisory-REBUILD-pending stats (ARBDP)))

Figure 17. DSNACCOX formula for recommending a REORG on an index space (Performance Monitoring and Tuning Guide)

User Report: Tablespaces to Reorg

```
SELECT SUBSTR(RTS.DBNAME,1,8)    DBNAME
     ,SUBSTR(RTS.NAME,1,8)      TSNAME
     ,SUBSTR(TB.NAME,1,20)      TBNAME
     ,CHAR(RTS.PARTITION)       PART
     ,DECIMAL((((RTS.REORGNEARINDREF+RTS.REORGFARINDREF)*1.00
              / RTS.TOTALROWS)*100),6,2)
                                PCINDREF
     ,DECIMAL(((RTS.REORGUNCLUSTINS*1.00
              / RTS.TOTALROWS)*100),6,2)    PCUNCLUST
     ,RTS.EXTENTS
     ,RTS.TOTALROWS
     ,RTS.REORGLASTTIME
     ,RTS.STATSLASTTIME
FROM SYSIBM.SYSTABLESPACESTATS  RTS
LEFT JOIN SYSIBM.SYSTABLES      TB
  ON TB.DBNAME = RTS.DBNAME
 AND TB.TSNAME = RTS.NAME
 AND TB.DBID   = RTS.DBID
WHERE RTS.REORGLASTTIME IS NOT NULL
  AND NOT EXISTS (SELECT 1
                  FROM SYSIBM.SYSCOPY SC
                  WHERE  SC.DBNAME = RTS.DBNAME
                    AND  SC.TSNAME = RTS.NAME
                    AND  SC.DSNUM  = RTS.PARTITION
                    AND  SC.ICTYPE = 'W'
                    AND  SC.TIMESTAMP > RTS.REORGLASTTIME
                  )
  AND RTS.DBNAME LIKE XXXDB%'
  AND RTS.TOTALROWS       > 0
  AND RTS.REORGUNCLUSTINS  < RTS.TOTALROWS
  AND (RTS.REORGMASSDELETE > 0
    OR (((RTS.REORGNEARINDREF+RTS.REORGFARINDREF)*1.00
        / RTS.TOTALROWS)*100) > 5
    OR ((RTS.REORGUNCLUSTINS*1.00 / RTS.TOTALROWS)*100) > 10
    OR RTS.EXTENTS > 200
)
  AND TB.NAME IN ('XXXTB')
ORDER BY RTS.DBNAME,RTS.NAME,RTS.PARTITION
WITH UR;
```

Report: Tablespaces to Reorg (cont.)

- Space criteria:

```
OR (EXISTS (SELECT 1
            FROM SYSIBM.SYSTABLES TB
            WHERE TB.DBNAME = RTS.DBNAME
              AND TB.TSNAME = RTS.NAME
              AND TB.DBID   = RTS.DBID
              AND (TB.AVGROWLEN * RTS.TOTALROWS/4000 <
                    RTS.NPAGES*0.5
                  AND TB.AVGROWLEN > 0
                  OR TB.RECLENGTH * RTS.TOTALROWS/4000 <
                    RTS.NPAGES*0.5
                  )
           )
       AND RTS.NPAGES > 1
       AND RTS.NPAGES IS NOT NULL
       AND RTS.STATSLASTTIME IS NOT NULL

      )
```

Report: Indexes to reorg

```
SELECT SUBSTR(RIX.DBNAME,1,8) DBNAME
     ,SUBSTR(RIX.NAME,1,18)  IXNAME
     ,RIX.PARTITION          PART
     ,SUBSTR(IX.TBNAME,1,20) TBNAME
     ,DECIMAL((RIX.REORGLEAFFAR       * 100.00)
           / RIX.NACTIVE     , 7,2)
                        "LEAFFAR%"
     ,RIX.REORGNUMLEVELS              NUMLEVL
     ,RIX.TOTALENTRIES                TOTENTR
     ,RIX.EXTENTS                     EXT
     ,RIX.REORGINSERTS                INSERTS
     ,RIX.REORGDELETES                DELETES
     ,RIX.REORGLASTTIME
     ,RIX.STATSLASTTIME
 FROM SYSIBM.SYSINDEXSPACESTATS RIX
 LEFT JOIN SYSIBM.SYSINDEXES    IX
  ON IX.CREATOR = RIX.CREATOR
  AND IX.NAME    = RIX.NAME
  AND IX.DBNAME  = RIX.DBNAME
  AND IX.DBID    = RIX.DBID
WHERE RIX.CREATOR                          = PRODUSER'
  AND RIX.TOTALENTRIES                     > 0
  AND (   RIX.REORGMASSDELETE              > 0
      OR ( (RIX.REORGPSEUDODELETES * 1.00)
         / RIX.TOTALENTRIES) * 100 > 10
      OR ( (RIX.REORGLEAFFAR       * 1.00)
         / RIX.NACTIVE     ) * 100 > 10
      OR ( (RIX.REORGAPPENDINSERT  * 1.00)
         / RIX.TOTALENTRIES) * 100 > 10
      OR RIX.REORGNUMLEVELS                > 1
      OR RIX.EXTENTS                       > 200
     )
ORDER BY RIX.DBNAME
       ,RIX.NAME
       ,RIX.PARTITION
 WITH UR;
```

Identifying Obsolete Indexes

```
SELECT DBNAME,LASTUSED,SUBSTR(NAME,1,18) NAME
  ,REORGLASTTIME,TOTALENTRIES
  ,(SELECT SUBSTR(TBNAME,1,18)
   FROM SYSIBM.SYSINDEXES
   WHERE NAME = IXS.NAME
     AND CREATOR = 'DB2PROD'
   ) TBNAME
  ,COALESCE (
   (SELECT SUBSTR(DNAME,1,8)
   FROM SYSIBM.SYSPACKDEP
   WHERE BNAME = IXS.NAME
   FETCH FIRST 1 ROW ONLY
   ), 'NONE'
   ) ONE_PACKAGE
  ,(SELECT UNIQUERULE
     FROM SYSIBM.SYSINDEXES IX
     WHERE IX.NAME  = IXS.NAME
   )  UNIQUERULE
  , FOREIGN_KEY_INFO                          For referential constraints
FROM SYSIBM.SYSINDEXSPACESTATS IXS
WHERE (LASTUSED IS NULL
    OR LASTUSED <='01/01/2009'
     )
AND DBNAME LIKE 'XXXDB%'
AND TOTALENTRIES > 0
ORDER BY DBNAME,NAME,PARTITION
WITH UR;
```

RTS History Tables

- RTS_TS_HISTORY
- RTS_IX_HISTORY
- CAPTURE_TS column added to the end
- Loaded at 8 a.m. and 6 p.m. every day

Using RTS History tables

- Frequently reorged tablespaces/partitions

```
SELECT DBNAME
    ,NAME
    ,(SELECT SUBSTR(NAME,1,20)
      FROM SYSIBM.SYSTABLES TB
      WHERE TB.TSNAME   = TH1.NAME
        AND TB.DBNAME   = TH1.DBNAME
        AND TB.TYPE     = 'T'
      FETCH FIRST 1 ROW ONLY
     )  TBNAME
    ,PARTITION
    ,REORGLASTTIME
FROM SODDBA.RTS_TS_HIST TH1
WHERE TH1.REORGLASTTIME IS NOT NULL
 AND TH1.REORGLASTTIME      > '2011-08-01-00.00.00.000000'
 AND TH1.CAPTURE_TS         > '2011-09-12-00.00.00.000000'          Last CAPTURE_TS
 AND EXISTS (SELECT 1
          FROM SODDBA.RTS_TS_HIST TH2
          WHERE TH2.DBNAME = TH1.DBNAME
            AND TH2.NAME   = TH1.NAME
            AND TH2.PARTITION = TH1.PARTITION
            AND (TH2.REORGLASTTIME + 28 DAYS) >
                TH1.REORGLASTTIME
            AND TH2.REORGLASTTIME < TH1.REORGLASTTIME
            AND TH2.CAPTURE_TS > '2011-08-01-00.00.00.000000'
         )
ORDER BY DBNAME
      ,NAME
      ,PARTITION
WITH UR;
```

Using RTS History tables (cont.)

- Frequently reorged indexes/partitions

```
SELECT DBNAME
    ,SUBSTR(NAME,1,10)    NAME
    ,(SELECT SUBSTR(TBNAME,1,20)
      FROM SYSIBM.SYSINDEXES IX
      WHERE IX.NAME     = TH1.NAME
        AND IX.CREATOR  = TH1.CREATOR
      FETCH FIRST 1 ROW ONLY
     )  TBNAME
    ,PARTITION
    ,REORGLASTTIME
FROM SODDBA.RTS_IX_HIST TH1
WHERE TH1.REORGLASTTIME IS NOT NULL
 AND TH1.REORGLASTTIME > '2011-08-01-00.00.00.000000'
 AND TH1.CAPTURE_TS > '2011-09-12-00.00.00.000000'
 AND EXISTS (SELECT 1
          FROM SODDBA.RTS_IX_HIST TH2
          WHERE TH2.DBNAME = TH1.DBNAME
            AND TH2.NAME   = TH1.NAME
            AND TH2.PARTITION = TH1.PARTITION
            AND (TH2.REORGLASTTIME + 28 DAYS) >
                TH1.REORGLASTTIME
            AND TH2.REORGLASTTIME < TH1.REORGLASTTIME
            AND TH2.CAPTURE_TS > '2011-08-01-00.00.00.000000'
         )
ORDER BY DBNAME
      ,NAME
      ,PARTITION
OPTIMIZE FOR 1 ROW
WITH UR;
```

Using RTS History tables (cont.)

- Object Growth Report (1% within 24 hours)

```
SELECT TH1.DBNAME
     ,TH1.NAME
     ,TH1.PARTITION
     ,(SELECT SUBSTR(NAME,1,20)
       FROM SYSIBM.SYSTABLES TB
       WHERE TB.TSNAME = TH1.NAME
         AND TB.DBNAME = TH1.DBNAME
         AND TB.TYPE   = 'T'
       FETCH FIRST 1 ROW ONLY
      ) TBNAME
     ,(TH1.TOTALROWS + TH1.REORGINSERTS - TH1.REORGDELETES)
        TOTAL_LAST
     ,(TH2.TOTALROWS + TH2.REORGINSERTS - TH2.REORGDELETES)
        TOTAL_PREV
     ,TH1.UPDATESTATSTIME
     ,TH2.UPDATESTATSTIME
FROM SODDBA.RTS_TS_HIST   TH1
  JOIN SODDBA.RTS_TS_HIST TH2
  ON TH1.DBNAME   = TH2.DBNAME
 AND TH1.NAME     = TH2.NAME
 AND TH1.PARTITION  = TH2.PARTITION
 AND TH1.REORGLASTTIME IS NOT NULL
 AND TH2.REORGLASTTIME IS NOT NULL
 AND TH1.TOTALROWS  >  0
 AND TH2.TOTALROWS  >  0
WHERE TH1.CAPTURE_TS > '2011-09-12-00.00.00.000000'
  AND TH2.CAPTURE_TS > '2011-09-11-00.00.00.000000'
  AND TH2.CAPTURE_TS <> TH1.CAPTURE_TS
  AND ((TH1.TOTALROWS + TH1.REORGINSERTS - TH1.REORGDELETES) -
        (TH2.TOTALROWS + TH2.REORGINSERTS - TH2.REORGDELETES)) >
       (TH2.TOTALROWS * TH2.REORGINSERTS - TH2.REORGDELETES) * 0.01
AND TH1.DBNAME LIKE 'XXXDB%'
ORDER BY TH1.DBNAME
      ,TH1.NAME
      ,TH1.PARTITION
WITH UR;
```