IDUG

IDUG VIRTUAL

2021 NA **Db2** Tech Conference

Virtual Indexes Explained

Ryszard Stepien, Broadcom

Db2 for z/OS

Agenda

- What is Virtual Index?
- Virtual Index Evolution
- Virtual Index Requirements
- Virtual Index Considerations
- Use Case

What is Virtual Index?

- Prototype of the actual index
- Helps evaluate indexes without actually creating them
- Simulates the impact of creating or dropping an index
- Virtual Index definitions reside in input explain tables
- Only affects the outcome of EXPLAIN PLAN
- Considered by the Db2 optimizer as part of the access path evaluation during EXPLAIN PLAN, along with existing catalog indexes
- No affect on BIND/REBIND or on SQL execution

Virtual Index Evolution (1|2)

- Retrofitted into Db2 Version 8 via APAR PK46687 (2008)
  - Db2 V8 CM used short name DSN_VIRTUAL_INDEX
  - Renamed in Db2 V8 NFM to DSN_VIRTUAL_INDEXES
- Introduced in Db2 9 (2008)
  - Managed in DSN_VIRTUAL_INDEDXES table
- Sample DDL in PDS *prefix.*SDSNSAMP member, along with other EXPLAIN tables
  - DSNTIJOS (Db2 V8 and 9)
  - DSNTESC (Db2 10, 11, and 12)
- Index types supported by DSN_VIRTUAL_INDEXES table:
  - Type-2 non-partitioned Indexes
  - Data Partitioned Secondary Indexes (DPSI)

Virtual Index Evolution (2|2)

- Db2 11 introduced a new virtual index table DSN_VIRTUAL_KEYTARGETS
  - Sample DDL in *prefix.*SDSNSAMP(DSNTESC)
- Index types supported by DSN_VIRTUAL_KEYTARGETS
  - XML indexes
  - Expression-based indexes

Virtual Index Requirements (1|11)

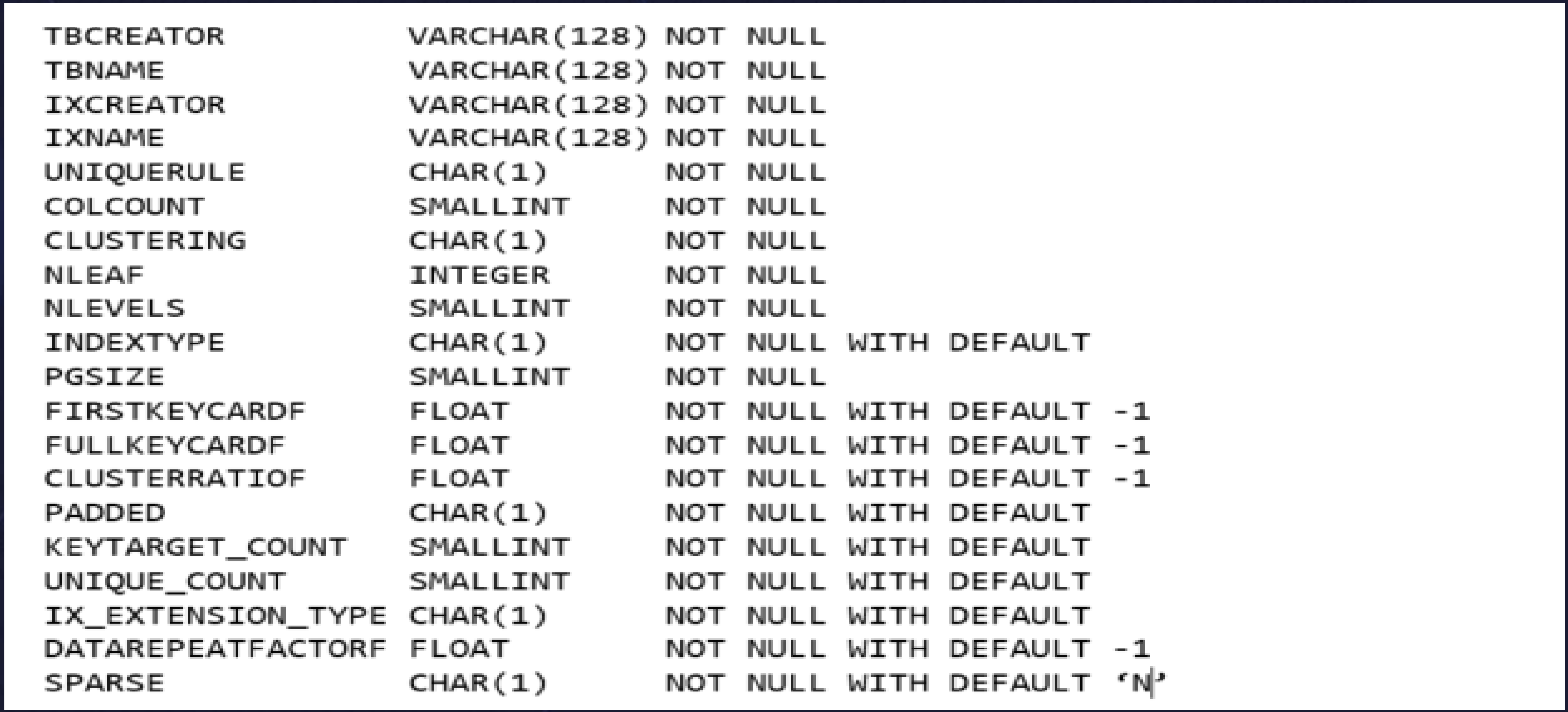
- Must have a set of EXPLAIN tables with the correct qualifier; i.e. PLAN_TABLE and DSN_STATEMNT_TABLE
  - Sample DDL in *prefix.*SDSNSAMP(DSNTESC)
- Must define the DSN_VIRTUAL_INDEXES and optionally the DSN_VIRTUAL_KEYTARGETS tables
  - These tables must have the same owner as the EXPLAIN tables. Otherwise, the optimizer ignores them.

Virtual Index Requirements (2|11)

- DSN_VIRTUAL_INDEXES table definitions are derived from related catalog tables:
  - SYSIBM.SYSINDEXES
  - SYSIBM.SYSKEYS
  - Plus a couple of new/unique columns

Virtual Index Requirements (3|11)

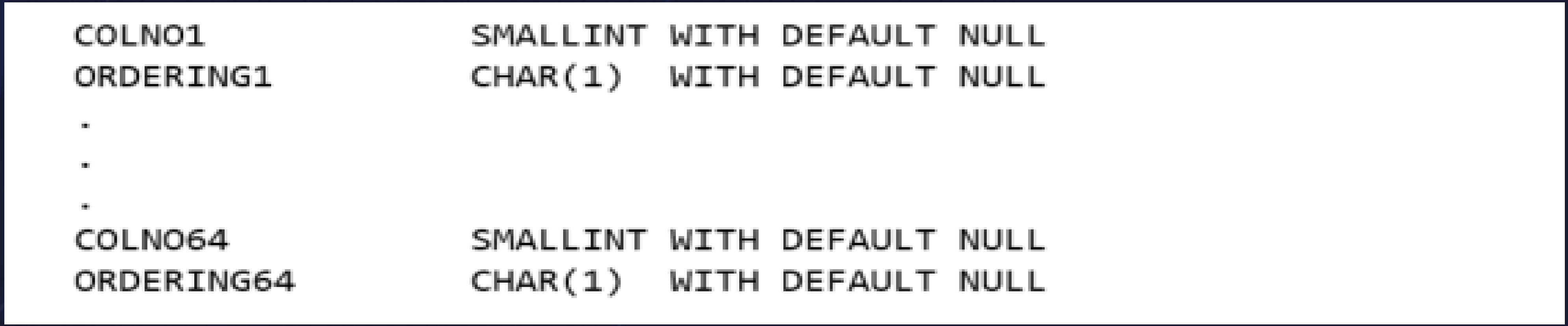
- SYSINDEXES columns in DSN_VIRTUAL_INDEXES:



```
TBCREATOR          VARCHAR(128) NOT NULL
TBNAME             VARCHAR(128) NOT NULL
IXCREATOR          VARCHAR(128) NOT NULL
IXNAME             VARCHAR(128) NOT NULL
UNIQUERULE         CHAR(1)      NOT NULL
COLCOUNT           SMALLINT     NOT NULL
CLUSTERING         CHAR(1)      NOT NULL
NLEAF              INTEGER      NOT NULL
NLEVELS            SMALLINT     NOT NULL
INDEXTYPE          CHAR(1)      NOT NULL WITH DEFAULT
PGSIZE             SMALLINT     NOT NULL
FIRSTKEYCARDF      FLOAT        NOT NULL WITH DEFAULT -1
FULLKEYCARDF       FLOAT        NOT NULL WITH DEFAULT -1
CLUSTERRATIOF      FLOAT        NOT NULL WITH DEFAULT -1
PADDED             CHAR(1)      NOT NULL WITH DEFAULT
KEYTARGET_COUNT    SMALLINT     NOT NULL WITH DEFAULT
UNIQUE_COUNT       SMALLINT     NOT NULL WITH DEFAULT
IX_EXTENSION_TYPE  CHAR(1)      NOT NULL WITH DEFAULT
DATAREPEATFACTORF  FLOAT        NOT NULL WITH DEFAULT -1
SPARSE             CHAR(1)      NOT NULL WITH DEFAULT ‘N’
```

Virtual Index Requirements (4|11)

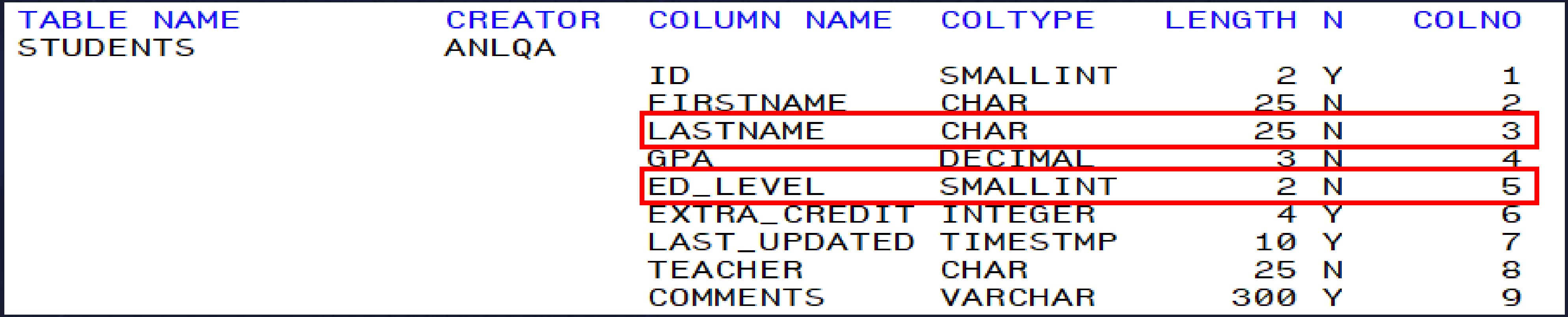
- SYSKEYS columns in DSN_VIRTUAL_INDEXES:
  - COLNO*1-64* – Indicates the column number in the index key
  - ORDERING*1-64* – Indicates the column order in the index key
    - A – Ascending
    - D – Descending



```
COLNO1              SMALLINT WITH DEFAULT NULL
ORDERING1           CHAR(1)  WITH DEFAULT NULL
.
.
.
COLNO64             SMALLINT WITH DEFAULT NULL
ORDERING64          CHAR(1)  WITH DEFAULT NULL
```

Virtual Index Requirements (5|11)

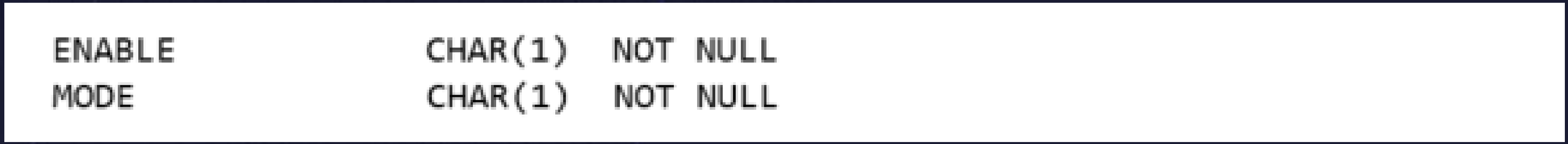
- Setting key columns
  - COLNO1 = 3 & ORDERING1 = A|D
  - COLNO2 = 5 & ORDERING2 = A|D



```
TABLE NAME              CREATOR    COLUMN NAME     COLTYPE    LENGTH N    COLNO
STUDENTS                ANLQA
                                   ID              SMALLINT        2 Y        1
                                   FIRSTNAME       CHAR           25 N        2
                                   LASTNAME        CHAR           25 N        3
                                   GPA             DECIMAL         3 N        4
                                   ED_LEVEL        SMALLINT        2 N        5
                                   EXTRA_CREDIT    INTEGER         4 Y        6
                                   LAST_UPDATED    TIMESTMP       10 Y        7
                                   TEACHER         CHAR           25 N        8
                                   COMMENTS        VARCHAR       300 Y        9
```

Virtual Index Requirements (6|11)

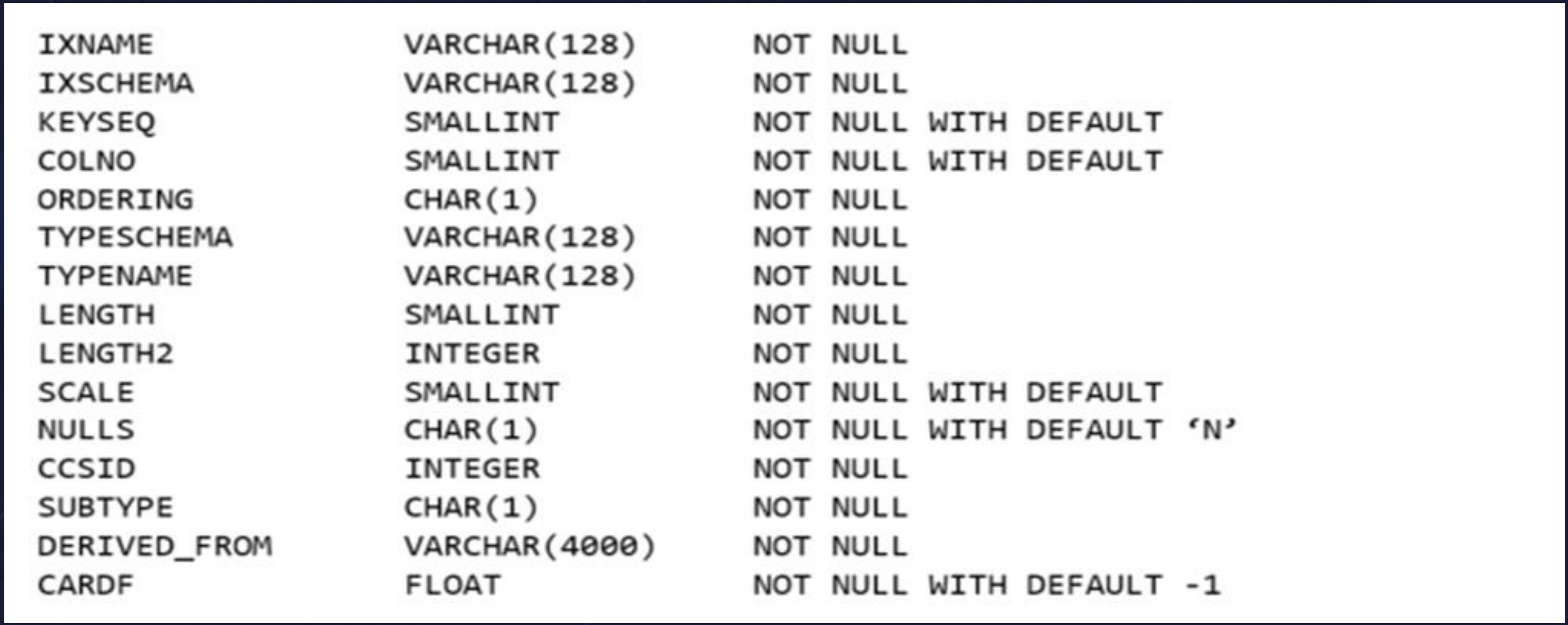
- DSN_VIRTUAL_INDEXES new/unique columns:
  - ENABLE – Indicates whether the virtual index should be considered by the optimizer
    - Y – Use this index (index definition must be valid or the index gets ignored)
    - N – Do not use this index
  - MODE – Indicates whether the index is being create or dropped
    - C – Create this index (new index)
    - D – Drop this index (old index)



```
ENABLE                  CHAR(1)   NOT NULL
MODE                    CHAR(1)   NOT NULL
```

Virtual Index Requirements (7|11)

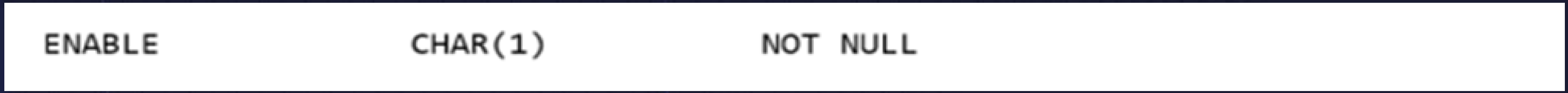
- DSN_VIRTUAL_KEYTARGETS definition shadows SYSIBM.SYSKEYTARGETS



```
IXNAME           VARCHAR(128)     NOT NULL
IXSCHEMA         VARCHAR(128)     NOT NULL
KEYSEQ           SMALLINT         NOT NULL WITH DEFAULT
COLNO            SMALLINT         NOT NULL WITH DEFAULT
ORDERING         CHAR(1)          NOT NULL
TYPESCHEMA       VARCHAR(128)     NOT NULL
TYPENAME         VARCHAR(128)     NOT NULL
LENGTH           SMALLINT         NOT NULL
LENGTH2          INTEGER          NOT NULL
SCALE            SMALLINT         NOT NULL WITH DEFAULT
NULLS            CHAR(1)          NOT NULL WITH DEFAULT ‘N’
CCSID            INTEGER          NOT NULL
SUBTYPE          CHAR(1)          NOT NULL
DERIVED_FROM     VARCHAR(4000)    NOT NULL
CARDF            FLOAT            NOT NULL WITH DEFAULT -1
```

Virtual Index Requirements (8|11)

- DSN_VIRTUAL_KEYTARGETS unique column:
  - ENABLE – Indicates whether the virtual index should be considered by the optimizer
    - Y – Use this index (index definition must be valid or the index gets ignored)
    - N – Do not use this index



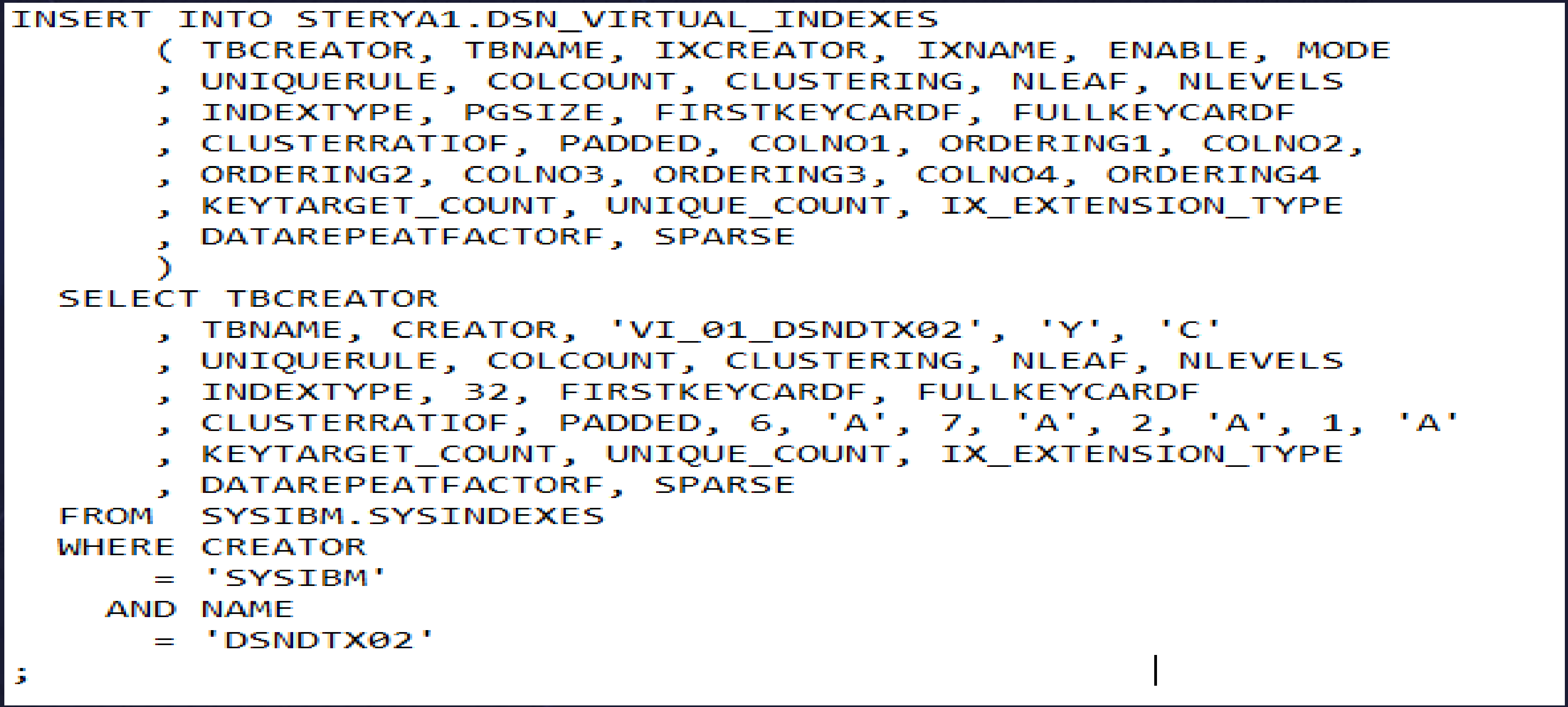
ENABLE	CHAR(1)	NOT NULL

Virtual Index Requirements (9|11)

- These tables were put in place in support of the vendor tools
  - Perhaps your vendor already supports virtual indexes (lucky you)
- If you want to play with it yourself, handle with care
  - Use caution when manually inserting or updating the virtual index tables because these tables may be used by vendor tools
  - Db2 optimizer ignores invalid index definitions
  - Estimating some of the virtual index values may be challenging i.e. NLEAF, NLEVELS, FIRSTKEYCARDF, FULLKEYCARDS, CLUSTERRATIOF, PGSIZE
    - Some help is available in the Db2 Administration Guide. Refer to “Calculating the space required for an index”
  - Check constrains help prevent invalid data
  - Write a REXX script to automate the process of populating data in the table

Virtual Index Requirements (10|11)

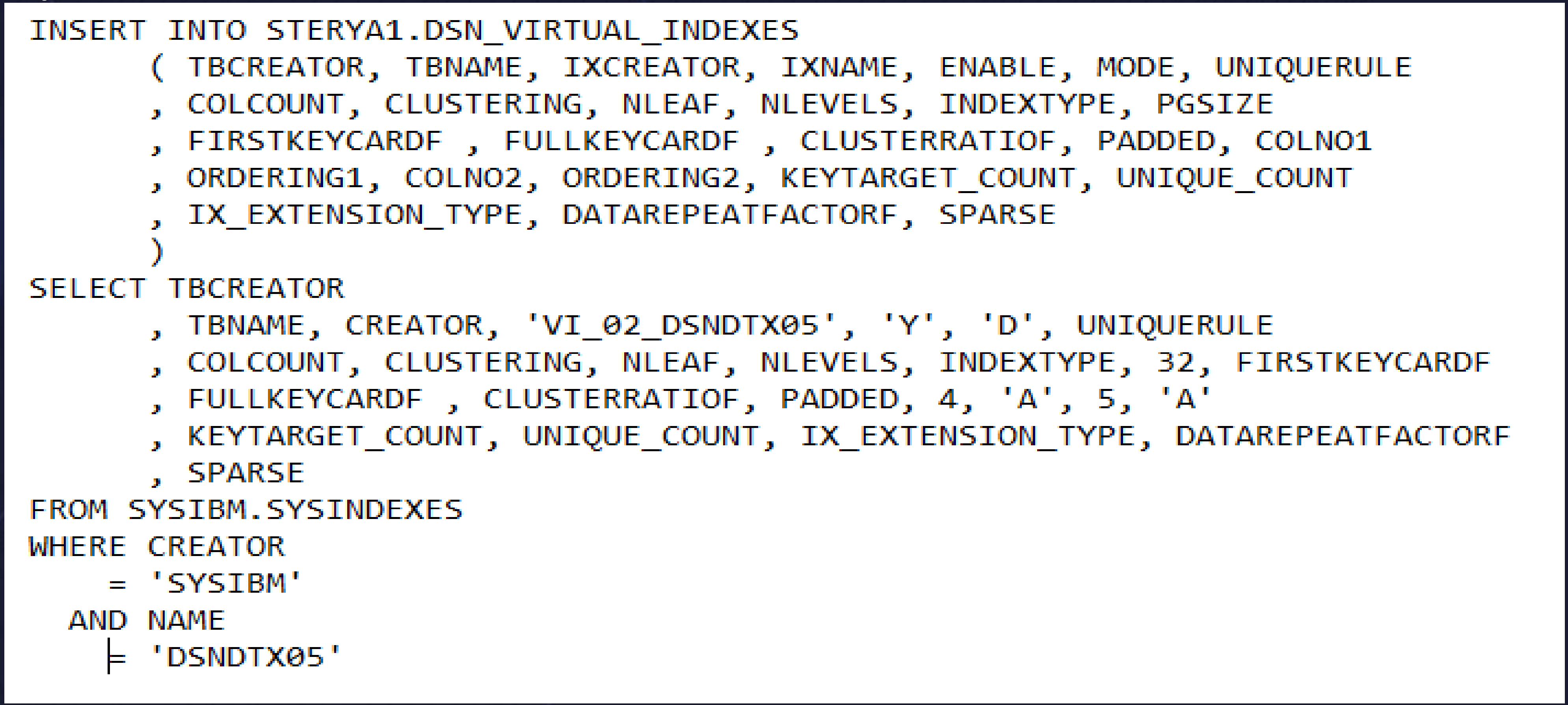
- Example of templating an existing index in the Db2 catalog for virtual index create



```
INSERT INTO STERYA1.DSN_VIRTUAL_INDEXES
      ( TBCREATOR, TBNAME, IXCREATOR, IXNAME, ENABLE, MODE
      , UNIQUERULE, COLCOUNT, CLUSTERING, NLEAF, NLEVELS
      , INDEXTYPE, PGSIZE, FIRSTKEYCARDF, FULLKEYCARDF
      , CLUSTERRATIOF, PADDED, COLNO1, ORDERING1, COLNO2,
      , ORDERING2, COLNO3, ORDERING3, COLNO4, ORDERING4
      , KEYTARGET_COUNT, UNIQUE_COUNT, IX_EXTENSION_TYPE
      , DATAREPEATFACTORF, SPARSE
      )
  SELECT TBCREATOR
      , TBNAME, CREATOR, 'VI_01_DSNDTX02', 'Y', 'C'
      , UNIQUERULE, COLCOUNT, CLUSTERING, NLEAF, NLEVELS
      , INDEXTYPE, 32, FIRSTKEYCARDF, FULLKEYCARDF
      , CLUSTERRATIOF, PADDED, 6, 'A', 7, 'A', 2, 'A', 1, 'A'
      , KEYTARGET_COUNT, UNIQUE_COUNT, IX_EXTENSION_TYPE
      , DATAREPEATFACTORF, SPARSE
  FROM  SYSIBM.SYSINDEXES
  WHERE CREATOR
      = 'SYSIBM'
    AND NAME
      = 'DSNDTX02'
;
```

Virtual Index Requirements (11|11)

- Example of templating an existing index in the Db2 catalog for virtual index drop



```
INSERT INTO STERYA1.DSN_VIRTUAL_INDEXES
      ( TBCREATOR, TBNAME, IXCREATOR, IXNAME, ENABLE, MODE, UNIQUERULE
      , COLCOUNT, CLUSTERING, NLEAF, NLEVELS, INDEXTYPE, PGSIZE
      , FIRSTKEYCARDF , FULLKEYCARDF , CLUSTERRATIOF, PADDED, COLNO1
      , ORDERING1, COLNO2, ORDERING2, KEYTARGET_COUNT, UNIQUE_COUNT
      , IX_EXTENSION_TYPE, DATAREPEATFACTORF, SPARSE
      )
SELECT TBCREATOR
      , TBNAME, CREATOR, 'VI_02_DSNDTX05', 'Y', 'D', UNIQUERULE
      , COLCOUNT, CLUSTERING, NLEAF, NLEVELS, INDEXTYPE, 32, FIRSTKEYCARDF
      , FULLKEYCARDF , CLUSTERRATIOF, PADDED, 4, 'A', 5, 'A'
      , KEYTARGET_COUNT, UNIQUE_COUNT, IX_EXTENSION_TYPE, DATAREPEATFACTORF
      , SPARSE
FROM SYSIBM.SYSINDEXES
WHERE CREATOR
    = 'SYSIBM'
  AND NAME
    = 'DSNDTX05'
```

Virtual Index Consideration

- Try an existing index with more, less, or different columns
- Test altered statistics on an index
- See what happens if an index is dropped
- Determine the impact of adding an index
- Etc.

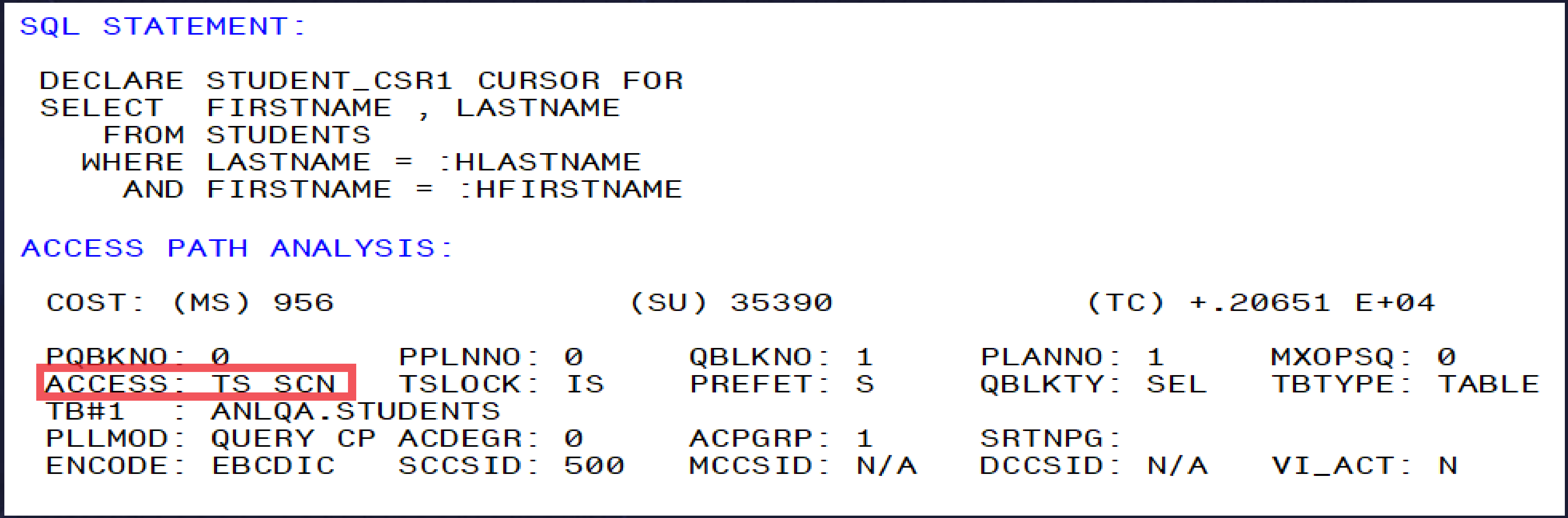
Use Case (1|13)

A case to illustrate the use of virtual indexes in solving SQL performance issues.

- Problem description
  - Recent spike in CPU and high GETPAGE count has been reported
- Troubleshoot the problem
  - You suspect the problem might be related to a new program package that was recently implemented
  - DB2 Explain shows an inefficient tablespace scan
- Explore potential solution
  - Implement a virtual index
- Asses the effectiveness of the candidate solution
  - Re-Explain the package and compare the access path results
- Implement permanent solution

Use Case (2|13)

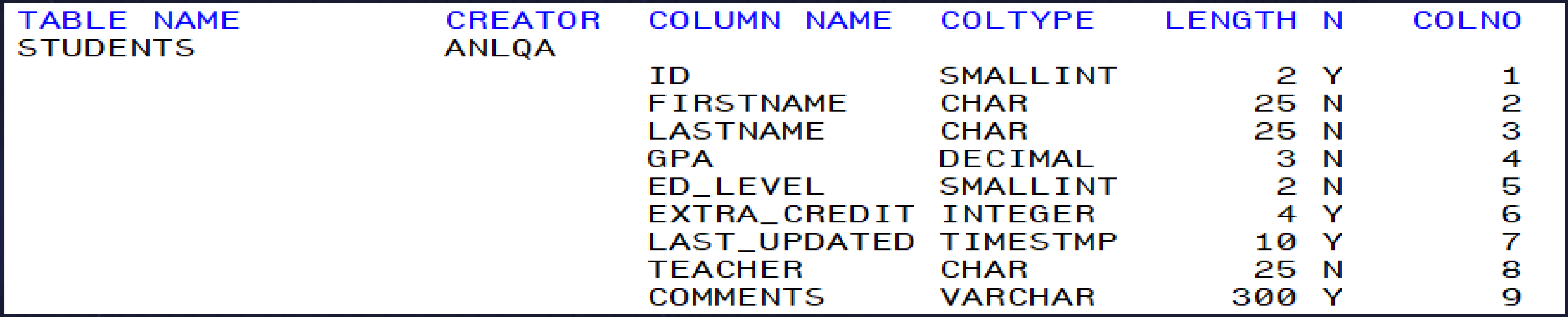
- EXPLAIN package reveals the problem



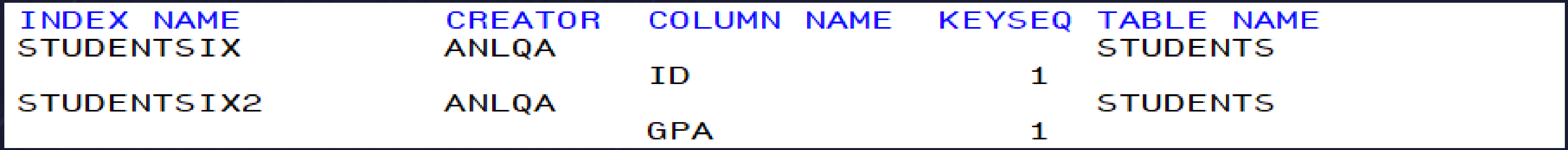
```
SQL STATEMENT:

 DECLARE STUDENT_CSR1 CURSOR FOR
 SELECT  FIRSTNAME , LASTNAME
    FROM STUDENTS
   WHERE LASTNAME = :HLASTNAME
     AND FIRSTNAME = :HFIRSTNAME

ACCESS PATH ANALYSIS:

 COST: (MS) 956              (SU) 35390            (TC) +.20651 E+04

 PQBKNO: 0        PPLNNO: 0     QBLKNO: 1     PLANNO: 1     MXOPSQ: 0
 ACCESS: TS SCN   TSLOCK: IS    PREFET: S     QBLKTY: SEL   TBTYPE: TABLE
 TB#1  : ANLQA.STUDENTS
 PLLMOD: QUERY CP ACDEGR: 0     ACPGRP: 1     SRTNPG:
 ENCODE: EBCDIC   SCCSID: 500   MCCSID: N/A   DCCSID: N/A   VI_ACT: N
```

Use Case (3|13)

- Identify problem object



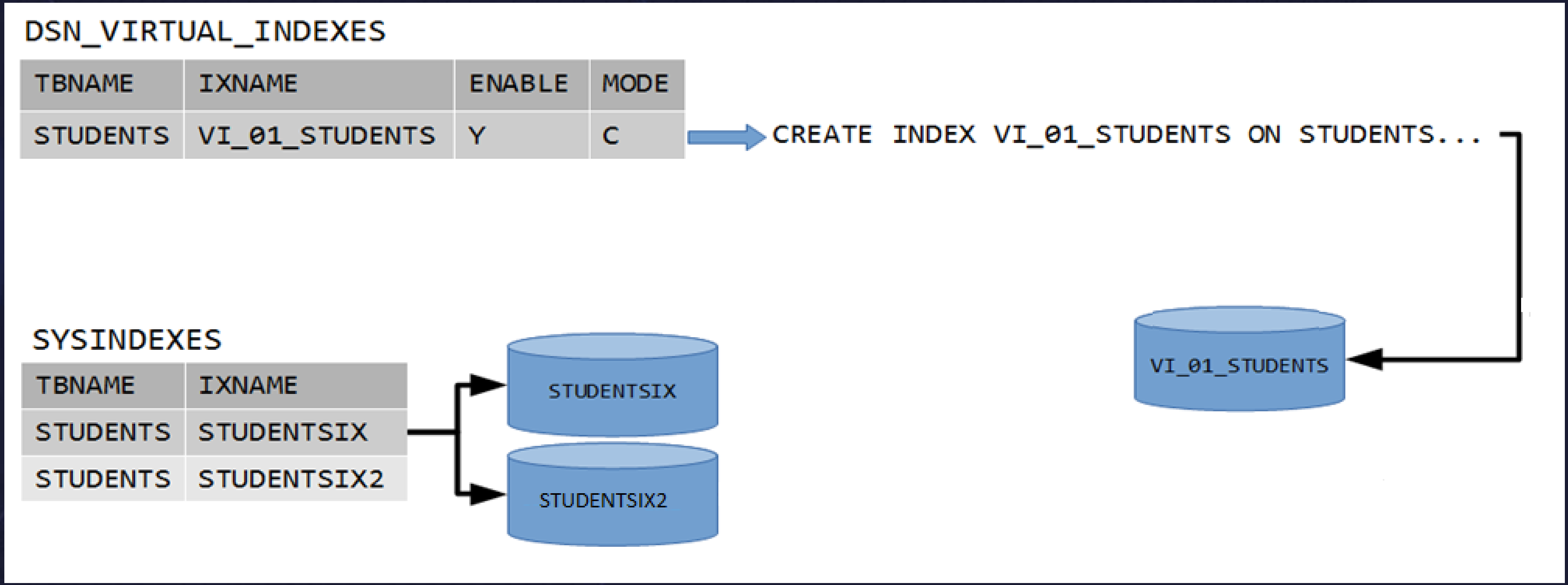
```
TABLE NAME         CREATOR  COLUMN NAME  COLTYPE   LENGTH N   COLNO
STUDENTS           ANLQA
                            ID           SMALLINT       2 Y       1
                            FIRSTNAME    CHAR          25 N       2
                            LASTNAME     CHAR          25 N       3
                            GPA          DECIMAL        3 N       4
                            ED_LEVEL     SMALLINT       2 N       5
                            EXTRA_CREDIT INTEGER        4 Y       6
                            LAST_UPDATED TIMESTMP      10 Y       7
                            TEACHER      CHAR          25 N       8
                            COMMENTS     VARCHAR      300 Y       9
```



```
INDEX NAME         CREATOR  COLUMN NAME  KEYSEQ TABLE NAME
STUDENTSIX         ANLQA                        STUDENTS
                            ID               1
STUDENTSIX2        ANLQA                        STUDENTS
                            GPA              1
```

Use Case (4|13)

- Define virtual index on LASTNAME and ED_LEVEL key columns



DSN_VIRTUAL_INDEXES

TBNAME	IXNAME	ENABLE	MODE
STUDENTS	VI_01_STUDENTS	Y	C

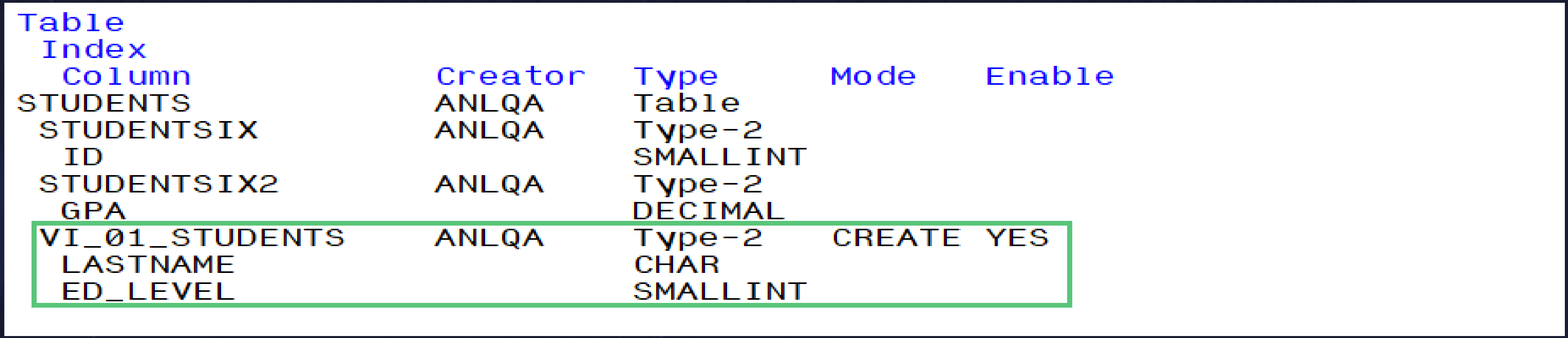
```
CREATE INDEX VI_01_STUDENTS ON STUDENTS...
```

SYSINDEXES

TBNAME	IXNAME
STUDENTS	STUDENTSIX
STUDENTS	STUDENTSIX2

Use Case (5|13)

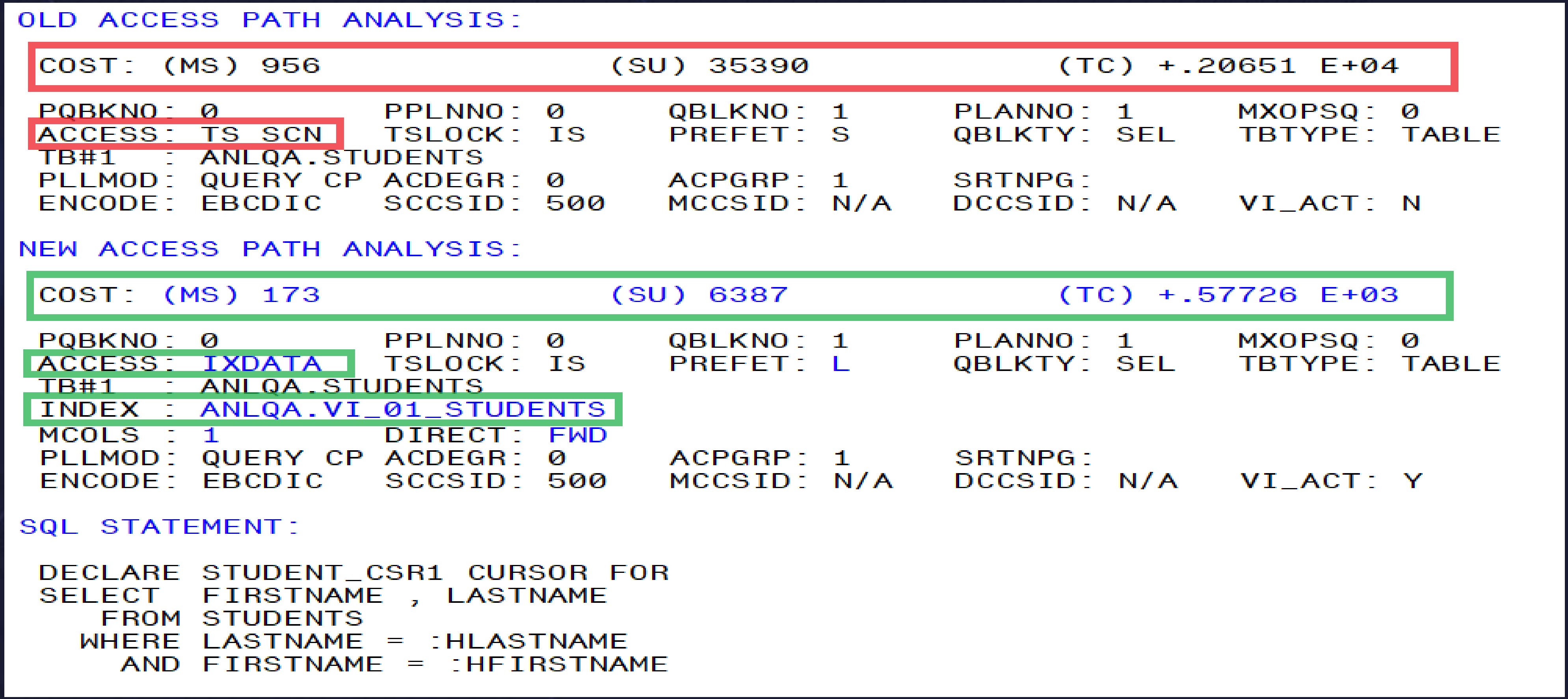
- Our new virtual index is enabled for consideration by the optimizer



```
Table
 Index
  Column                 Creator   Type      Mode    Enable
STUDENTS                 ANLQA     Table
 STUDENTSIX              ANLQA     Type-2
  ID                               SMALLINT
 STUDENTSIX2             ANLQA     Type-2
  GPA                              DECIMAL
 VI_01_STUDENTS          ANLQA     Type-2    CREATE  YES
  LASTNAME                         CHAR
  ED_LEVEL                         SMALLINT
```

Use Case (6|13)

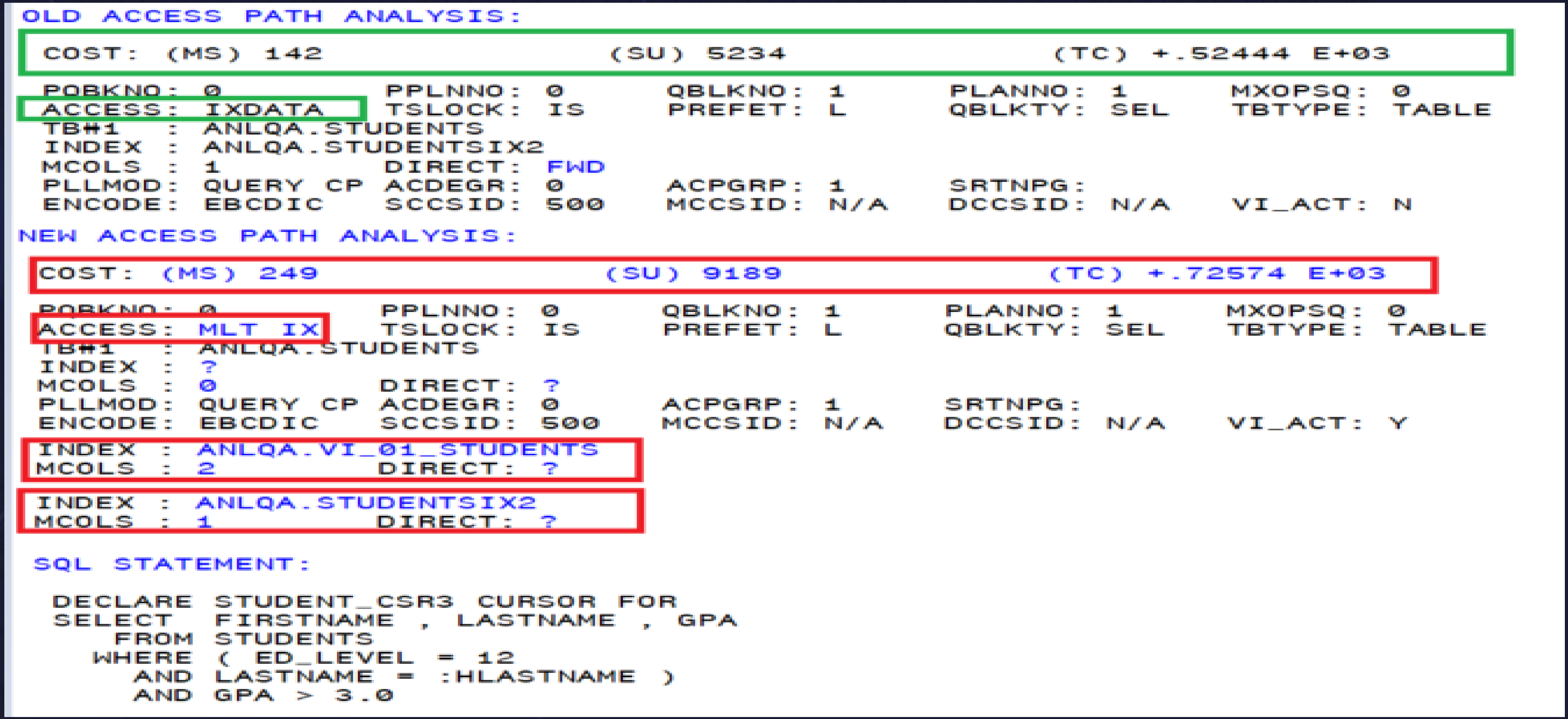
- Re-EXPLAIN shows positive effect of the virtual index on access path



```
OLD ACCESS PATH ANALYSIS:

 COST: (MS) 956                (SU) 35390                (TC) +.20651 E+04

 PQBKNO: 0       PPLNNO: 0     QBLKNO: 1     PLANNO: 1     MXOPSQ: 0
 ACCESS: TS SCN  TSLOCK: IS    PREFET: S     QBLKTY: SEL   TBTYPE: TABLE
 TB#1   : ANLQA.STUDENTS
 PLLMOD: QUERY CP ACDEGR: 0    ACPGRP: 1     SRTNPG:
 ENCODE: EBCDIC  SCCSID: 500   MCCSID: N/A   DCCSID: N/A   VI_ACT: N

NEW ACCESS PATH ANALYSIS:

 COST: (MS) 173                (SU) 6387                 (TC) +.57726 E+03

 PQBKNO: 0       PPLNNO: 0     QBLKNO: 1     PLANNO: 1     MXOPSQ: 0
 ACCESS: IXDATA  TSLOCK: IS    PREFET: L     QBLKTY: SEL   TBTYPE: TABLE
 TB#1   : ANLQA.STUDENTS
 INDEX  : ANLQA.VI_01_STUDENTS
 MCOLS  : 1      DIRECT: FWD
 PLLMOD: QUERY CP ACDEGR: 0    ACPGRP: 1     SRTNPG:
 ENCODE: EBCDIC  SCCSID: 500   MCCSID: N/A   DCCSID: N/A   VI_ACT: Y

SQL STATEMENT:

 DECLARE STUDENT_CSR1 CURSOR FOR
 SELECT  FIRSTNAME , LASTNAME
    FROM STUDENTS
   WHERE LASTNAME  = :HLASTNAME
     AND FIRSTNAME  = :HFIRSTNAME
```

Use Case (7|13)

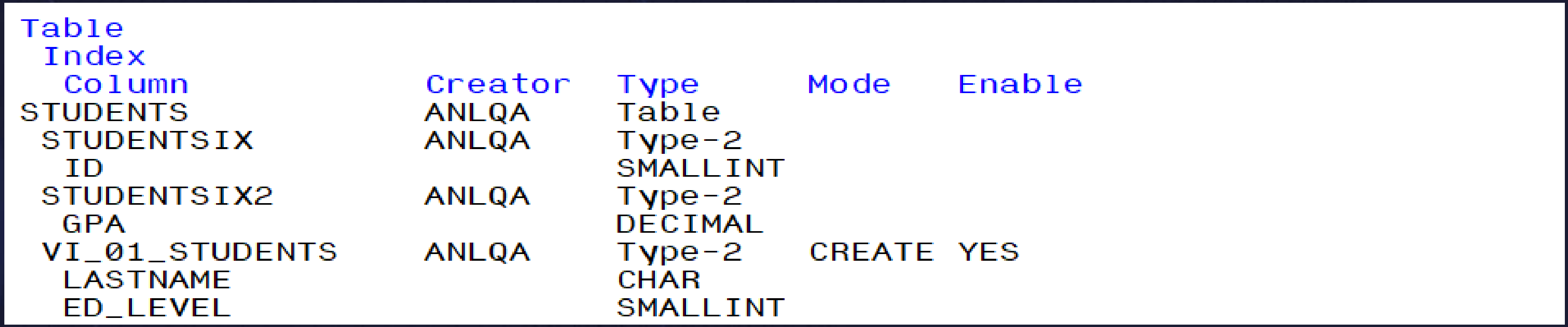
- But the new IX causes access path degradation for another SQL statement



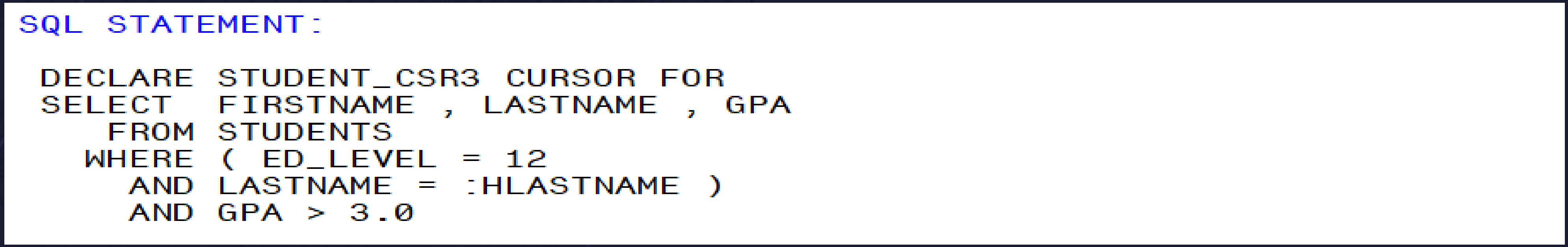
```
OLD ACCESS PATH ANALYSIS:
 COST: (MS) 142                    (SU) 5234                     (TC) +.52444 E+03
 PQBKNO: 0          PPLNNO: 0      QBLKNO: 1      PLANNO: 1      MXOPSQ: 0
 ACCESS: IXDATA     TSLOCK: IS     PREFET: L      QBLKTY: SEL    TBTYPE: TABLE
 TB#1   : ANLQA.STUDENTS
 INDEX  : ANLQA.STUDENTSIX2
 MCOLS  : 1         DIRECT: FWD
 PLLMOD: QUERY CP   ACDEGR: 0      ACPGRP: 1      SRTNPG:
 ENCODE: EBCDIC     SCCSID: 500    MCCSID: N/A    DCCSID: N/A    VI_ACT: N
NEW ACCESS PATH ANALYSIS:
 COST: (MS) 249                    (SU) 9189                     (TC) +.72574 E+03
 PQBKNO: 0          PPLNNO: 0      QBLKNO: 1      PLANNO: 1      MXOPSQ: 0
 ACCESS: MLT IX     TSLOCK: IS     PREFET: L      QBLKTY: SEL    TBTYPE: TABLE
 TB#1   : ANLQA.STUDENTS
 INDEX  : ?
 MCOLS  : 0         DIRECT: ?
 PLLMOD: QUERY CP   ACDEGR: 0      ACPGRP: 1      SRTNPG:
 ENCODE: EBCDIC     SCCSID: 500    MCCSID: N/A    DCCSID: N/A    VI_ACT: Y
 INDEX  : ANLQA.VI_01_STUDENTS
 MCOLS  : 2         DIRECT: ?
 INDEX  : ANLQA.STUDENTSIX2
 MCOLS  : 1         DIRECT: ?

 SQL STATEMENT:

  DECLARE STUDENT_CSR3 CURSOR FOR
  SELECT  FIRSTNAME , LASTNAME , GPA
     FROM STUDENTS
    WHERE ( ED_LEVEL = 12
      AND LASTNAME = :HLASTNAME )
      AND GPA > 3.0
```

Use Case (8|13)

- Review all indexes and the affected SQL statement



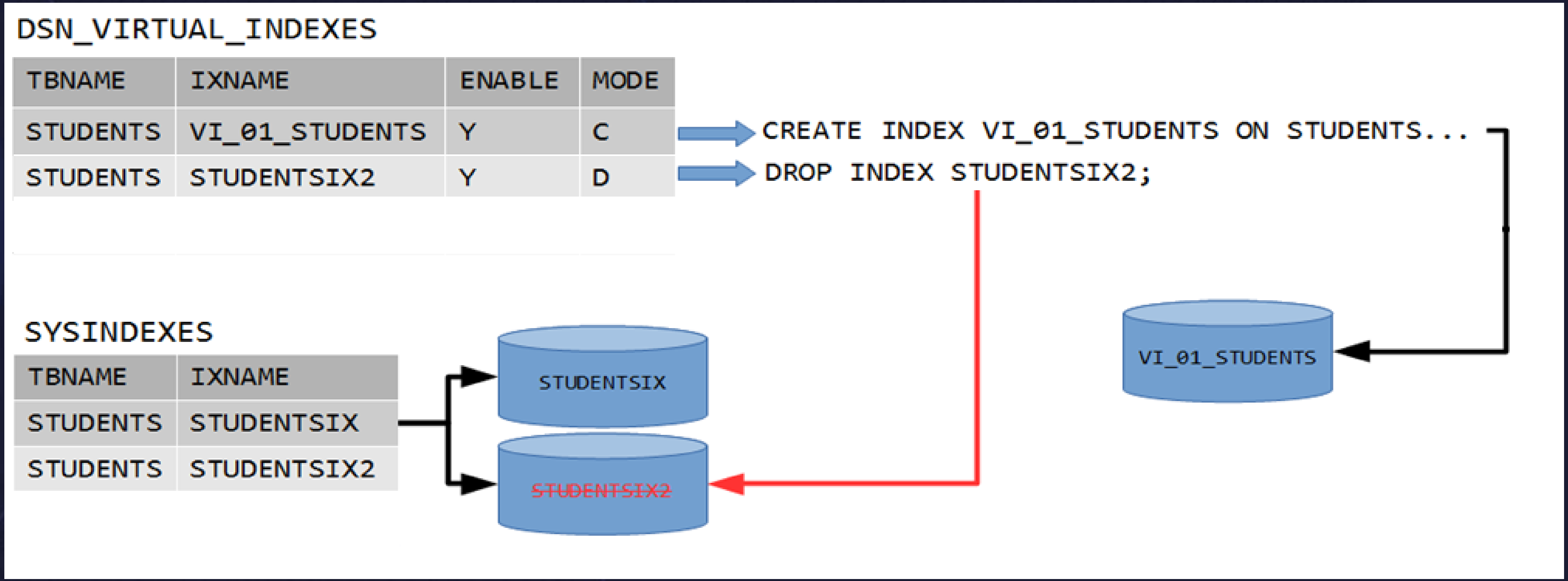
```
Table
 Index
  Column           Creator  Type      Mode    Enable
STUDENTS           ANLQA    Table
 STUDENTSIX        ANLQA    Type-2
  ID                        SMALLINT
 STUDENTSIX2       ANLQA    Type-2
  GPA                       DECIMAL
 VI_01_STUDENTS    ANLQA    Type-2    CREATE  YES
  LASTNAME                  CHAR
  ED_LEVEL                  SMALLINT
```



```
SQL STATEMENT:

 DECLARE STUDENT_CSR3 CURSOR FOR
 SELECT  FIRSTNAME , LASTNAME , GPA
    FROM STUDENTS
   WHERE ( ED_LEVEL = 12
     AND LASTNAME = :HLASTNAME )
     AND GPA > 3.0
```

Use Case (9|13)

- Evaluate drop of an existing indexes



DSN_VIRTUAL_INDEXES

TBNAME	IXNAME	ENABLE	MODE
STUDENTS	VI_01_STUDENTS	Y	C
STUDENTS	STUDENTSIX2	Y	D

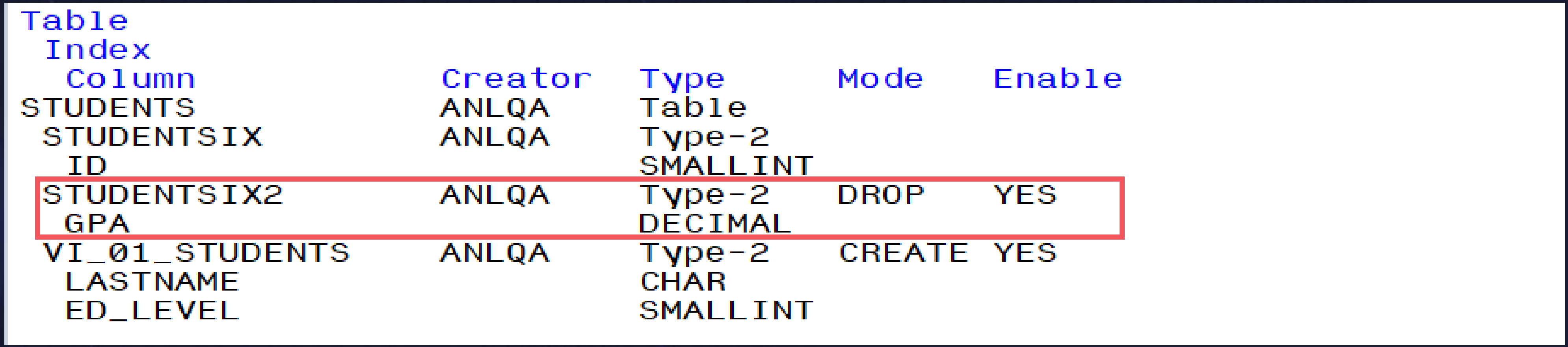
```
CREATE INDEX VI_01_STUDENTS ON STUDENTS...
DROP INDEX STUDENTSIX2;
```

SYSINDEXES

TBNAME	IXNAME
STUDENTS	STUDENTSIX
STUDENTS	STUDENTSIX2

Use Case (10|13)

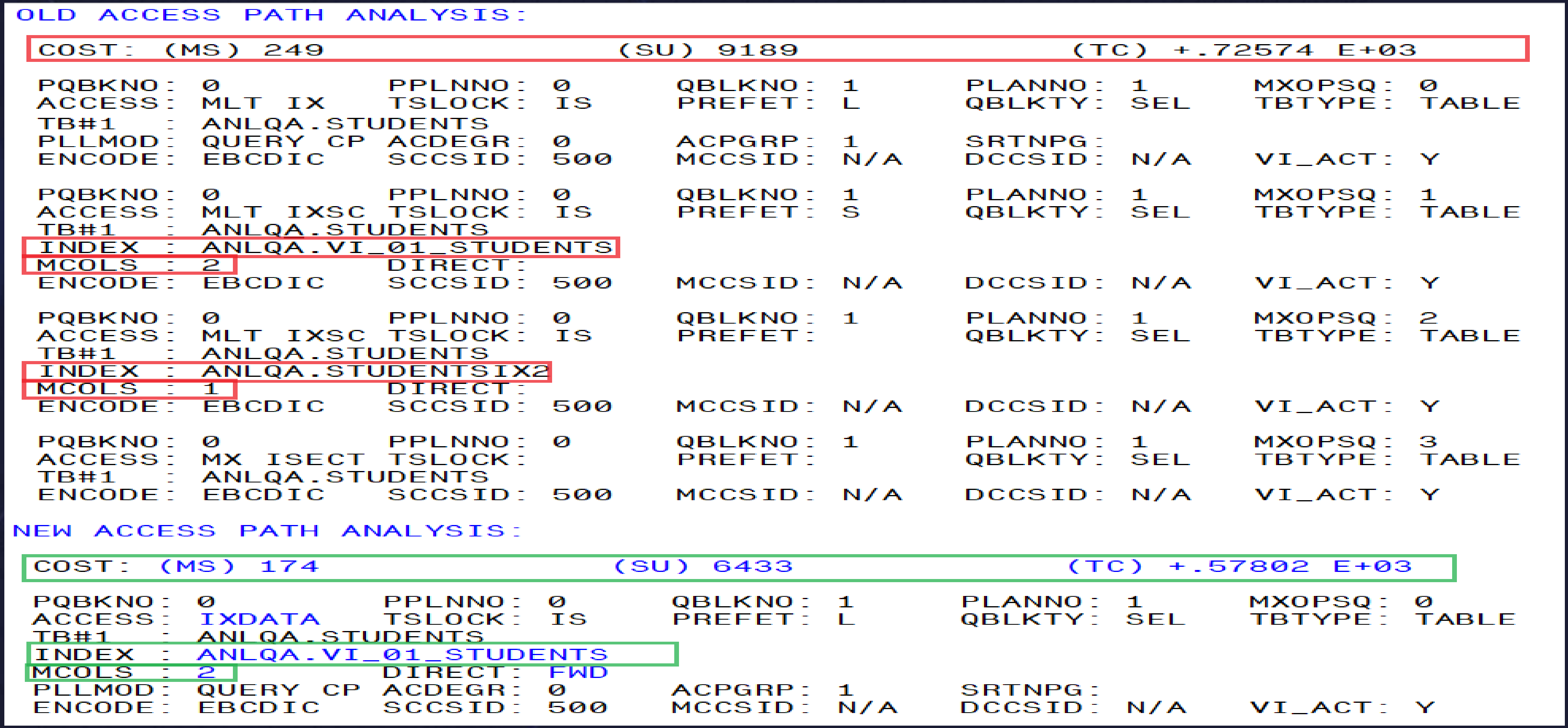
- Add the existing index STUDENTSIX2 to DSN_VIRTUAL_INDXES for drop evaluation



```
Table
 Index
  Column            Creator  Type     Mode   Enable
STUDENTS            ANLQA    Table
 STUDENTSIX         ANLQA    Type-2
  ID                         SMALLINT
 STUDENTSIX2        ANLQA    Type-2   DROP   YES
  GPA                        DECIMAL
 VI_01_STUDENTS     ANLQA    Type-2   CREATE YES
  LASTNAME                   CHAR
  ED_LEVEL                   SMALLINT
```

Use Case (11|13)

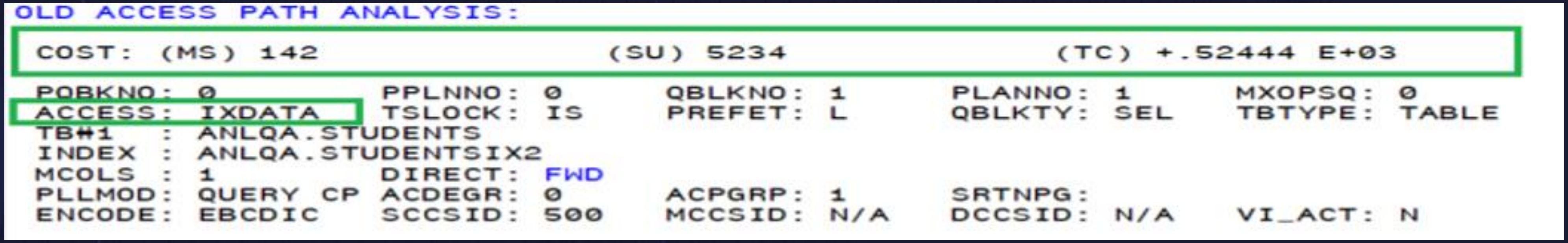
- EXPLAIN output shows improvement with of new access



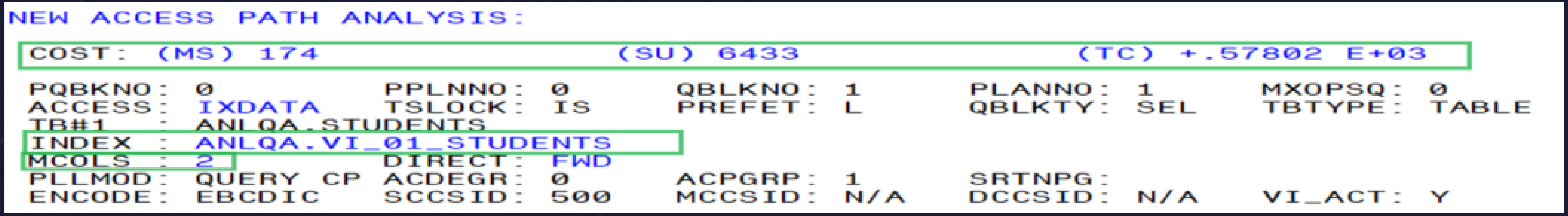
```
OLD ACCESS PATH ANALYSIS:
  COST:  (MS) 249                     (SU) 9189                      (TC)  +.72574  E+03

  PQBKNO:  0          PPLNNO:  0       QBLKNO:  1       PLANNO:  1       MXOPSQ:  0
  ACCESS:  MLT  IX    TSLOCK:  IS      PREFET:  L       QBLKTY:  SEL     TBTYPE:  TABLE
  TB#1    :  ANLQA.STUDENTS
  PLLMOD:  QUERY  CP  ACDEGR:  0       ACPGRP:  1       SRTNPG:
  ENCODE:  EBCDIC     SCCSID:  500     MCCSID:  N/A     DCCSID:  N/A     VI_ACT:  Y

  PQBKNO:  0          PPLNNO:  0       QBLKNO:  1       PLANNO:  1       MXOPSQ:  1
  ACCESS:  MLT  IXSC  TSLOCK:  IS      PREFET:  S       QBLKTY:  SEL     TBTYPE:  TABLE
  TB#1    :  ANLQA.STUDENTS
  INDEX   :  ANLQA.VI_01_STUDENTS
  MCOLS   :  2        DIRECT:
  ENCODE:  EBCDIC     SCCSID:  500     MCCSID:  N/A     DCCSID:  N/A     VI_ACT:  Y

  PQBKNO:  0          PPLNNO:  0       QBLKNO:  1       PLANNO:  1       MXOPSQ:  2
  ACCESS:  MLT  IXSC  TSLOCK:  IS      PREFET:          QBLKTY:  SEL     TBTYPE:  TABLE
  TB#1    :  ANLQA.STUDENTS
  INDEX   :  ANLQA.STUDENTSIX2
  MCOLS   :  1        DIRECT:
  ENCODE:  EBCDIC     SCCSID:  500     MCCSID:  N/A     DCCSID:  N/A     VI_ACT:  Y

  PQBKNO:  0          PPLNNO:  0       QBLKNO:  1       PLANNO:  1       MXOPSQ:  3
  ACCESS:  MX  ISECT  TSLOCK:          PREFET:          QBLKTY:  SEL     TBTYPE:  TABLE
  TB#1    :  ANLQA.STUDENTS
  ENCODE:  EBCDIC     SCCSID:  500     MCCSID:  N/A     DCCSID:  N/A     VI_ACT:  Y
NEW ACCESS PATH ANALYSIS:
  COST:  (MS) 174                     (SU) 6433                      (TC)  +.57802  E+03

  PQBKNO:  0          PPLNNO:  0       QBLKNO:  1       PLANNO:  1       MXOPSQ:  0
  ACCESS:  IXDATA     TSLOCK:  IS      PREFET:  L       QBLKTY:  SEL     TBTYPE:  TABLE
  TB#1    :  ANLQA.STUDENTS
  INDEX   :  ANLQA.VI_01_STUDENTS
  MCOLS   :  2        DIRECT:  FWD
  PLLMOD:  QUERY  CP  ACDEGR:  0       ACPGRP:  1       SRTNPG:
  ENCODE:  EBCDIC     SCCSID:  500     MCCSID:  N/A     DCCSID:  N/A     VI_ACT:  Y
```

Use Case (12|13)

- Performance with new access path is slightly degraded from the original access path, but it is an acceptable tradeoff for this test case



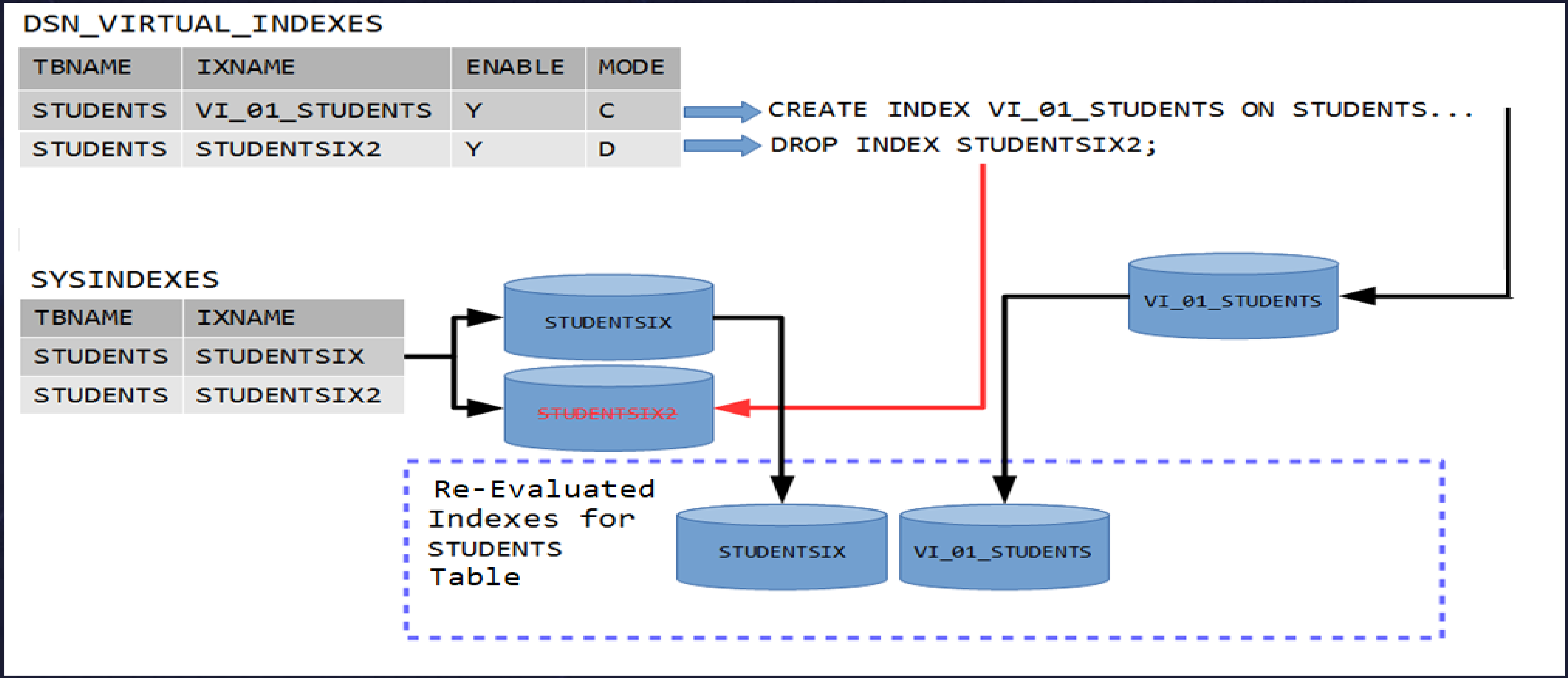
```
OLD ACCESS PATH ANALYSIS:
 COST: (MS) 142                (SU) 5234              (TC) +.52444 E+03
 PQBKNO: 0       PPLNNO: 0     QBLKNO: 1     PLANNO: 1     MXOPSQ: 0
 ACCESS: IXDATA  TSLOCK: IS    PREFET: L     QBLKTY: SEL   TBTYPE: TABLE
 TB#1  : ANLQA.STUDENTS
 INDEX : ANLQA.STUDENTSIX2
 MCOLS : 1       DIRECT: FWD
 PLLMOD: QUERY CP ACDEGR: 0    ACPGRP: 1     SRTNPG:
 ENCODE: EBCDIC  SCCSID: 500   MCCSID: N/A   DCCSID: N/A   VI_ACT: N
```



```
NEW ACCESS PATH ANALYSIS:
 COST: (MS) 174                (SU) 6433              (TC) +.57802 E+03
 PQBKNO: 0       PPLNNO: 0     QBLKNO: 1     PLANNO: 1     MXOPSQ: 0
 ACCESS: IXDATA  TSLOCK: IS    PREFET: L     QBLKTY: SEL   TBTYPE: TABLE
 TB#1  : ANLQA.STUDENTS
 INDEX : ANLQA.VI_01_STUDENTS
 MCOLS : 2       DIRECT: FWD
 PLLMOD: QUERY CP ACDEGR: 0    ACPGRP: 1     SRTNPG:
 ENCODE: EBCDIC  SCCSID: 500   MCCSID: N/A   DCCSID: N/A   VI_ACT: Y
```

Use Case (13|13)

- Virtual Index evaluation completed



DSN_VIRTUAL_INDEXES

TBNAME	IXNAME	ENABLE	MODE
STUDENTS	VI_01_STUDENTS	Y	C
STUDENTS	STUDENTSIX2	Y	D

CREATE INDEX VI_01_STUDENTS ON STUDENTS...

DROP INDEX STUDENTSIX2;

SYSINDEXES

TBNAME	IXNAME
STUDENTS	STUDENTSIX
STUDENTS	STUDENTSIX2

Thank You

Speaker: Ryszard Stepien
Company: Broadcom Inc.
Email Address: ryszard.stepien@broadcom.com

Session code:

Please fill out your session evaluation before leaving!!!!