DB2® 9 for z/OS Explain Tables

Download the EXPLAIN table DDL and these queries at WWW.DB2EXPERT.COM

Yevich, Lawson & Associates, Inc.

bmcsoftware

Use this query to view critical EXPLAIN information. Replace the COLLID and PROGNAME predicates with the package of interest, or use the query "as is" preceded by an EXPLAIN statement.

```
WITH MAXTIME (COLLID, PROGNAME, QUERYNO, BIND_TIME) AS
(SELECT COLLID, PROGNAME, QUERYNO, MAX(BIND_TIME)
 FROM   PLAN_TABLE
 WHERE  COLLID   = 'DSNESPCS'
 AND    PROGNAME = 'DSNESM68'
 GROUP  BY COLLID, PROGNAME, QUERYNO)

SELECT SUBSTR(P.PROGNAME, 1, 8) AS PROGNAME
     ,SUBSTR(DIGITS(P.QUERYNO), 6) CONCAT '-'
      CONCAT SUBSTR(DIGITS(P.QBLOCKNO), 4)
      CONCAT '-' CONCAT
      SUBSTR(DIGITS(P.PLANNO), 4) AS QQP
     ,SUBSTR(CHAR(P.METHOD), 1, 3) AS MTH
     ,SUBSTR(CHAR(F.PREDNO), 1, 3) AS P_NO
     ,SUBSTR(CHAR(P.MERGE_JOIN_COLS), 1, 3) AS MJC
     ,SUBSTR(P.CREATOR, 1, 8) AS TBCREATOR
     ,SUBSTR(P.TNAME, 1, 18) AS TBNAME
     ,SUBSTR(P.CORRELATION_NAME, 1, 8) AS CORR_NM
     ,DEC(D.ONECOMPROWS, 10, 1) AS ROWS_POST_FILTER
     ,P.ACCESSTYPE AS ATYP
     ,SUBSTR(P.ACCESSNAME, 1, 15) AS A_NM
     ,P.INDEXONLY AS IXO
     ,CHAR(P.MIXOPSEQ) AS MIX
     ,CHAR(P.MATCHCOLS) MCOL
     ,F.STAGE AS STAGE
     ,DEC(E.FILTER_FACTOR, 11, 10) AS FF
     ,E.BOOLEAN_TERM AS BT
     ,SUBSTR(E.TEXT, 1, 30) AS PRED_TEXT30
     ,P.SORTN_JOIN CONCAT P.SORTC_UNIQ CONCAT
      P.SORTC_JOIN CONCAT P.SORTC_ORDERBY CONCAT
      P.SORTC_GROUPBY AS NJ_CUJOG
     ,P.PREFETCH AS PF
     ,P.COLUMN_FN_EVAL AS CFE
     ,P.PAGE_RANGE AS PGRNG
     ,P.JOIN_TYPE AS JT
     ,P.QBLOCK_TYPE AS QB_TYP
     ,P.PARENT_QBLOCKNO AS P_QB
     ,P.TABLE_TYPE AS TB_TYP
     ,P.BIND_TIME AS B_TM

FROM PLAN_TABLE P
INNER JOIN
      MAXTIME   M
ON    M.COLLID    = P.COLLID
AND   M.PROGNAME  = P.PROGNAME
AND   M.QUERYNO   = P.QUERYNO
AND   M.BIND_TIME = P.BIND_TIME
LEFT JOIN
      DSN_FILTER_TABLE F
ON    M.COLLID    = F.COLLID
AND   M.PROGNAME  = F.PROGNAME
AND   M.QUERYNO   = F.QUERYNO
AND   P.QBLOCKNO  = F.QBLOCKNO
AND   P.PLANNO    = F.PLANNO
AND   M.BIND_TIME = F.EXPLAIN_TIME
and   P.ACCESSTYPE Not IN ('MX', 'MI', 'MU')
LEFT JOIN
      DSN_PREDICAT_TABLE E
ON    F.PROGNAME  = E.PROGNAME
AND   F.QUERYNO   = E.QUERYNO
AND   F.QBLOCKNO  = E.QBLOCKNO
AND   F.PREDNO    = E.PREDNO
AND   M.BIND_TIME = E.EXPLAIN_TIME
LEFT JOIN
TABLE  (SELECT MIN(X.ONECOMPROWS) AS ONECOMPROWS
        FROM   DSN_DETCOST_TABLE X
        WHERE  M.PROGNAME  = X.PROGNAME
        AND    M.QUERYNO   = X.QUERYNO
        AND    P.QBLOCKNO  = X.QBLOCKNO
        AND    P.PLANNO    = X.PLANNO
        AND    M.BIND_TIME = X.EXPLAIN_TIME) AS D
ON 1=1
ORDER  BY PROGNAME, B_TM, QQP, MIX, F.PREDNO  ;
```

Build indexes on your EXPLAIN tables for improved performance of your EXPLAIN queries. An index on COLLID, PROGNAME, QUERYNO, and BIND_TIME (or EXPLAIN_TIME) will help these queries.

Plan Table
Contains information about access paths for queries that were explained or hints.

		D
QUERYNO	INTEGER	1
QBLOCKNO	SMALLINT	
APPLNAME	VARCHAR(24)	
PROGNAME	VARCHAR(128)	
PLANNO	SMALLINT	
METHOD	SMALLINT	
CREATOR	VARCHAR(128)	
TNAME	VARCHAR(128)	
TABNO	SMALLINT	
ACCESSTYPE	CHAR(2)	
MATCHCOLS	SMALLINT	
ACCESSCREATOR	VARCHAR(128)	
ACCESSNAME	VARCHAR(128)	
INDEXONLY	CHAR(1)	
SORTN_UNIQ	CHAR(1)	
SORTN_JOIN	CHAR(1)	
SORTN_ORDERBY	CHAR(1)	
SORTN_GROUPBY	CHAR(1)	
SORTC_UNIQ	CHAR(1)	
SORTC_JOIN	CHAR(1)	
SORTC_ORDERBY	CHAR(1)	
SORTC_GROUPBY	CHAR(1)	
TSLOCKMODE	CHAR(3)	
TIMESTAMP	CHAR(16)	
REMARKS	VARCHAR(762)	
PREFETCH	CHAR(1)	
COLUMN_FN_EVAL	CHAR(1)	
MIXOPSEQ	SMALLINT	
VERSION	VARCHAR(64)	
COLLID	VARCHAR(128)	
ACCESS_DEGREE	SMALLINT	
ACCESS_PGROUP_ID	SMALLINT	
JOIN_DEGREE	SMALLINT	
JOIN_PGROUP_ID	SMALLINT	
SORTC_PGROUP_ID	SMALLINT	
SORTN_PGROUP_ID	SMALLINT	
PARALLELISM_MODE	CHAR(1)	
MERGE_JOIN_COLS	SMALLINT	
CORRELATION_NAME	CHAR(18)	
PAGE_RANGE	CHAR(1)	
JOIN_TYPE	CHAR(1)	
GROUP_MEMBER	VARCHAR(24)	
IBM_SERVICE_DATA	VARCHAR(254)	
WHEN_OPTIMIZE	CHAR(1)	
QBLOCK_TYPE	CHAR(6)	
BIND_TIME	TIMESTAMP	2
OPTHINT	VARCHAR(128)	
HINT_USED	VARCHAR(128)	
PRIMARY_ACCESSTYPE	CHAR(1)	
PARENT_QBLOCK	SMALLINT	
TABLE_TYPE	CHAR(1)	
TABLE_ENCODE	CHAR(1)	
TABLE_SCCSID	SMALLINT	
TABLE_MCCSID	SMALLINT	
TABLE_DCCSID	SMALLINT	
ROUTINE_ID	INTEGER	
CTREF	SMALLINT	
STMTTOKEN	VARCHAR(240)	
PARENT_PLANNO	SMALLINT	

DSN_PREDICAT_TABLE
Contains information about all the predicates in a query.

		D
QUERYNO	INTEGER	1
QBLOCKNO	SMALLINT	
APPLNAME	VARCHAR(24)	
PROGNAME	VARCHAR(128)	
PREDNO	INTEGER	
TYPE	CHAR(8)	
LEFT_HAND_SIDE	VARCHAR(128)	
LEFT_HAND_PNO	INTEGER	
LHS_TABNO	SMALLINT	
LHS_QBNO	SMALLINT	
RIGHT_HAND_SIDE	VARCHAR(128)	
RIGHT_HAND_PNO	INTEGER	
RHS_TABNO	SMALLINT	
RHS_QBNO	SMALLINT	
FILTER_FACTOR	FLOAT	
BOOLEAN_TERM	CHAR(1)	
SEARCHARG	CHAR(1)	
AFTER_JOIN	CHAR(1)	
ADDED_PRED	CHAR(1)	
REDUNDANT_PRED	CHAR(1)	
DIRECT_ACCESS	CHAR(1)	
KEYFIELD	CHAR(1)	
EXPLAIN_TIME	TIMESTAMP	2
CATEGORY	SMALLINT	
CATEGORY_B	SMALLINT	
PRED_ENCODE	CHAR(1)	
PRED_CCSID	SMALLINT	
PRED_MCCSID	SMALLINT	
MARKER	CHAR(1)	
PARENT_PNO	INTEGER	
NEGATION	CHAR(1)	
LITERALS	VARCHAR(128)	
CLAUSE	CHAR(8)	
GROUP_MEMBER	VARCHAR(24)	

DSN_FILTER_TABLE
Contains information about how predicates are used during query processing.

		D
QUERYNO	INTEGER	1
QBLOCKNO	SMALLINT	
PLANNO	SMALLINT	
APPLNAME	VARCHAR(24)	
PROGNAME	VARCHAR(128)	
COLLID	VARCHAR(128)	
ORDERNO	INTEGER	
PREDNO	INTEGER	
STAGE	CHAR(9)	
ORDERCLASS	INTEGER	
EXPLAIN_TIME	TIMESTAMP	2
MIXOPSEQ	SMALLINT	
REEVAL	CHAR(1)	
GROUP_MEMBER	VARCHAR(24)	

The separate queries here will produce the plan table information in one result set, and the additional information about the predicates in a separate result set. It will not produce the multiple rows for compound predicates (the query in the blue box on the left will produce multiple rows).

```
SELECT SUBSTR(P.PROGNAME, 1, 8) AS PROGNAME
     ,SUBSTR(DIGITS(P.QUERYNO), 6) CONCAT '-'
      CONCAT SUBSTR(DIGITS(P.QBLOCKNO), 4)
      CONCAT '-' CONCAT
      SUBSTR(DIGITS(P.PLANNO), 4) AS QQP
     ,SUBSTR(CHAR(P.METHOD), 1, 3) AS MTH
     ,SUBSTR(CHAR(P.MERGE_JOIN_COLS), 1, 3) AS MJC
     ,SUBSTR(P.CREATOR, 1, 8) AS TBCREATOR
     ,SUBSTR(P.TNAME, 1, 18) AS TBNAME
     ,SUBSTR(P.CORRELATION_NAME, 1, 8) AS CORR_NM
     ,DEC(D.ONECOMPROWS, 10, 1) AS ROWS_POST_FILTER
     ,P.ACCESSTYPE AS ATYP
     ,SUBSTR(P.ACCESSNAME, 1, 15) AS A_NM
     ,P.INDEXONLY AS IXO
     ,CHAR(P.MIXOPSEQ) AS MIX
     ,CHAR(P.MATCHCOLS) MCOL
     ,P.SORTN_JOIN CONCAT P.SORTC_UNIQ CONCAT
      P.SORTC_JOIN CONCAT P.SORTC_ORDERBY CONCAT
      P.SORTC_GROUPBY AS NJ_CUJOG
     ,P.PREFETCH AS PF
     ,P.COLUMN_FN_EVAL AS CFE
     ,P.PAGE_RANGE AS PGRNG
     ,P.JOIN_TYPE AS JT
     ,P.QBLOCK_TYPE AS QB_TYP
     ,P.PARENT_QBLOCKNO AS P_QB
     ,P.TABLE_TYPE AS TB_TYP
     ,P.BIND_TIME AS B_TM
FROM   PLAN_TABLE P
LEFT JOIN TABLE  (SELECT MIN(X.ONECOMPROWS)
AS ONECOMPROWS
       FROM   DSN_DETCOST_TABLE X
       WHERE  P.PROGNAME  = X.PROGNAME
       AND    P.QUERYNO   = X.QUERYNO
       AND    P.QBLOCKNO  = X.QBLOCKNO
       AND    P.PLANNO    = X.PLANNO
       AND    P.BIND_TIME = X.EXPLAIN_TIME) AS D
ON 1=1
ORDER  BY PROGNAME, B_TM, QQP, MIX;
SELECT SUBSTR(F.PROGNAME, 1, 8) AS PROGNAME
     ,SUBSTR(DIGITS(F.QUERYNO), 6) CONCAT '-'
      CONCAT SUBSTR(DIGITS(F.QBLOCKNO), 4)
      CONCAT '-' CONCAT
      SUBSTR(DIGITS(F.PLANNO), 4) AS QQP,
      PREDNO, STAGE, EXPLAIN_TIME AS B_TM
FROM   DSN_FILTER_TABLE F
ORDER  BY PROGNAME, B_TM, QQP, PREDNO;

SELECT SUBSTR(P.PROGNAME, 1, 8) AS PROGNAME
     ,SUBSTR(DIGITS(P.QUERYNO), 6) CONCAT '-'
      CONCAT SUBSTR(DIGITS(P.QBLOCKNO), 4) AS QQ,
      PREDNO, FILTER_FACTOR, BOOLEAN_TERM AS BT,
      JOIN, AFTER_JOIN AS AJ, ADDED_PRED AS AP,
      REDUNDANT_PRED AS RP,
      KEYFIELD, TEXT, EXPLAIN_TIME AS B_TM
FROM   DSN_PREDICAT_TABLE P
ORDER  BY PROGNAME, B_TM, QQ, PREDNO;
```

DSN_PGROUP_TABLE
Contains information about the parallel groups in a query.

		D
QUERYNO	INTEGER	1
QBLOCKNO	SMALLINT	
PLANNAME	VARCHAR(24)	
COLLID	VARCHAR(128)	
PROGNAME	VARCHAR(128)	
EXPLAIN_TIME	TIMESTAMP	2
VERSION	VARCHAR(122)	
GROUPID	SMALLINT	
FIRSTPLAN	SMALLINT	
LASTPLAN	SMALLINT	
CPUCOST	REAL	
IOCOST	REAL	
BESTTIME	REAL	
DEGREE	SMALLINT	
MODE	CHAR(1)	
REASON	SMALLINT	
LOCALCPU	SMALLINT	
TOTALCPU	SMALLINT	
FIRSTBASE	SMALLINT	
LARGETS	CHAR(1)	
PARTKIND	CHAR(1)	
GROUPTYPE	CHAR(3)	
ORDER	CHAR(1)	
STYLE	CHAR(4)	
RANGEKIND	CHAR(1)	
NKEYCOLS	SMALLINT	
LOWBOUND	VARCHAR(40)	
HIGHBOUND	VARCHAR(40)	
LOWKEY	VARCHAR(40	
HIGHKEY	VARCHAR(40)	
FIRSTPAGE	CHAR(4)	
LASTPAGE	CHAR(4)	
GROUP_MEMBER	VARCHAR(24)	
HOST_REASON	SMALLINT	
PARA_TYPE	CHAR(4)	
PART_INNER	CHAR(1)	
GRNU_KEYRNG	CHAR(1)	
OPEN_KEYRNG	CHAR(1))	

DSN_PTASK_TABLE
Contains information about the parallel tasks in a query.

		D
QUERYNO	INTEGER	1
QBLOCKNO	SMALLINT	
APPLNAME	VARCHAR(24)	
PROGNAME	VARCHAR(128)	
LPTNO	SMALLINT	
KEYCOLID	SMALLINT	
DPSI	CHAR(1)	
LPTLOKEY	VARCHAR(40)	
LPTHIKEY	VARCHAR(40)	
LPTLOPAG	CHAR(4)	
LPTLHIPAG	CHAR(4)	
LPTLOPG#	CHAR(4)	
LPTHIPG#	CHAR(4)	
LPTLOPT#	SMALLINT	
KEYCOLDT	SMALLINT	
KEYCOLPREC	SMALLINT	
KEYCOLSCAL	SMALLINT	
EXPLAIN_TIME	TIMESTAMP	2
GROUP_MEMBER	VARCHAR(24	

DSN_SORT_TABLE
Contains information about sort operations required for a query.

		D
QUERYNO	INTEGER	1
QBLOCKNO	SMALLINT	
PLANNO	SMALLINT	
APPLNAME	VARCHAR(24)	
PROGNAME	VARCHAR(128)	
COLLID	VARCHAR(128)	
SORTC	CHAR(5)	
SORTN	CHAR(5)	
SORTNO	SMALLINT	
KEYSIZE	SMALLINT	
ORDERCLASS	INTEGER	
EXPLAIN_TIME	TIMESTAMP	2
GROUP_MEMBER	VARCHAR(24)	

DSN_DETCOST_TABLE
Contains information about detailed cost estimation of the mini-plans in a query.

		D
QUERYNO	INTEGER	1
QBLOCKNO	SMALLINT	
APPLNAME	VARCHAR(24)	
PROGNAME	VARCHAR(128)	
PLANNO	SMALLINT	
OPENIO	FLOAT(4)	
OPENCPU	FLOAT(4)	
OPENCOST	FLOAT(4)	
DMIO	FLOAT(4)	
DMCPU	FLOAT(4)	
DMTOT	FLOAT(4)	
SUBIO	FLOAT(4)	
SUBCOST	FLOAT(4)	
BASEIO	FLOAT(4)	
BASECPU	FLOAT(4)	
BASETOT	FLOAT(4)	
ONECOMPROWS	FLOAT(4)	
IMLEAF	FLOAT(4)	
IMIO	FLOAT(4)	
IMPREFH	CHAR(2)	
IMMPRED	INTEGER	
IMFF	FLOAT(4)	
IMSRPRED	INTEGER	
IMFFADJ	FLOAT(4)	
IMSCANCST	FLOAT(4)	
IMREDSORT	FLOAT(4)	
IMMERGCST	FLOAT(4)	
IMCPU	FLOAT(4)	
IMTOT	FLOAT(4)	
IMSEQNO	SMALLINT	
DMPREFEH	CHAR(2)	
DMCLUDIO	FLOAT(4)	
DMCLUDIO	FLOAT(4)	
DMPREDS	INTEGER	
DMSROWS	FLOAT(4)	
DMSCANCST	FLOAT(4)	
DMROWS	FLOAT(4)	
DMCOLS	SMALLINT	
RDSROWCST	FLOAT(4)	
DMPAGECST	FLOAT(4)	
DMDATAIO	FLOAT(4)	
DMDATACPU	FLOAT(4)	
DMDATATOT	FLOAT(4)	
RDSROW	FLOAT(4)	
SNCOLS	SMALLINT	
SNROWS	FLOAT(4)	
SNRUNS	FLOAT(4)	
SNMERGES	FLOAT(4)	
SNIOCOST	FLOAT(4)	
SNCPUCOST	FLOAT(4)	
SNCOST	FLOAT(4)	
SNCSANIO	FLOAT(4)	
SNSCANCPU	FLOAT(4)	
SNCCOLS	FLOAT(4)	
SCROWS	FLOAT(4)	
SRECSZ	INTEGER	
SCPAGES	FLOAT(4)	
SCRUNS	FLOAT(4)	
SCMERGES	FLOAT(4)	
SCIOCOST	FLOAT(4)	
SCCPUCOST	FLOAT(4)	
SCCOST	FLOAT(4)	
SCSCANIO	FLOAT(4)	
SCSCANCPU	FLOAT(4)	
SCSCANCOST	FLOAT(4)	
COMPCARD	FLOAT(4)	
COMPIOCOST	FLOAT(4)	
COMPCPUCOST	FLOAT(4)	
COMPCOST	FLOAT(4)	
JOINCOLS	SMALLINT	
EXPLAIN_TIME	TIMESTAMP	2
GROUP_MEMBER	VARCHAR(24)	

DSN_SORTKEY_TABLE
Contains information about sort keys for all the sorts required by a query.

		D
QUERYNO	INTEGER	1
QBLOCKNO	SMALLINT	
PLANNO	SMALLINT	
APPLNAME	VARCHAR(24)	
PROGNAME	VARCHAR(128)	
COLLID	VARCHAR(128)	
SORTNO	SMALLINT	
ORDERNO	SMALLINT	
EXPTYPE	CHAR(3)	
TEXT	VARCHAR(128)	
TABNO	SMALLINT	
COLNO	SMALLINT	
DATATYPE	CHAR(18)	
LENGTH	INTEGER	
CCSID	INTEGER	
ORDERCLASS	INTEGER	
EXPLAIN_TIME	TIMESTAMP	2
GROUP_MEMBER	VARCHAR(24)	

DSN_PGRANGE_TABLE
Contains information about qualified partitions for all page range scans in a query.

		D
QUERYNO	INTEGER	1
QBLOCKNO	SMALLINT	
RANGE	SMALLINT	
FIRSTPART	SMALLINT	
LASTPART	SMALLINT	
NUMPARTS	SMALLINT	
EXPLAIN_TIME	TIMESTAMP	2
GROUP_MEMBER	VARCHAR(24)	

DSN_VIEWREF_TABLE
Contains information about all the views and materialized query tables used to process a query.

		D
QUERYNO	INTEGER	1
APPLNAME	VARCHAR(24)	
PROGNAME	VARCHAR(128)	
VERSION	VARCHAR(122)	
CREATOR	VARCHAR(128)	
NAME	VARCHAR(128)	
TYPE	CHAR(1)	
MQTUSE	SMALLINT	
EXPLAIN_TIME	TIMESTAMP	2
GROUP_MEMBER	VARCHAR(24)	

DSN_QUERY_TABLE
Contains information about an SQL statement and displays the statement before and after query transformation in XML.

		D
QUERYNO	INTEGER	1
TYPE	CHAR(8)	
QUERY_STAGE	CHAR(8)	
SEQNO	INTEGER	
NODE_DATA	CLOB(2M)	
EXPLAIN_TIME	TIMESTAMP	2
QUERY_ROWID	ROWID	
GROUP_MEMBER	VARCHAR(24)	
HASHKEY	INTEGER	
HASH_PRED	CHAR(1)	

DSN_STATEMNT_TABLE
Contains information about the estimated cost of specified SQL statements

		D
QUERYNO	INTEGER	1
APPLNAME	VARCHAR(24)	
PROGNAME	VARCHAR(128)	
COLLID	VARCHAR(128)	
GROUP_MEMBER	VARCHAR(24)	
EXPLAIN_TIME	TIMESTAMP	2
STMT_TYPE	CHAR(6)	
COST_CATEGORY	CHAR(1)	
PROCMS	INTEGER	
PROCSU	INTEGER	
REASON	VARCHAR(254)	
STMT_ENCODE	CHAR(1)	
TOTAL_COST	FLOAT	

DSN_FUNCTION_TABLE
Contains descriptions of functions that are used in specified SQL statements.

		D
QUERYNO	INTEGER	1
QBLOCKNO	INTEGER	
APPLNAME	VARCHAR(24)	
PROGNAME	VARCHAR(128)	
COLLID	VARCHAR(128)	
GROUP_MEMBER	VARCHAR(24)	
EXPLAIN_TIME	TIMESTAMP	2
SCHEMA_NAME	VARCHAR(128)	
FUNCTION_NAME	VARCHAR(128)	
SPEC_FUNC_NAME	VARCHAR(128)	
FUNCTION_TYPE	CHAR(2)	
VIEW_CREATOR	VARCHAR(128)	
VIEW_NAME	VARCHAR(128)	
PATH	VARCHAR(2048)	
FUNCTION_TEXT	VARCHAR(1500)	

DSN_STATEMENT_CACHE_TABLE
Contains information about the SQL statements in the statement cache.

		D
STMT_ID	INTEGER	1
STMT_TOKEN	VARCHAR(240)	
COLLID	VARCHAR(128)	
PROGRAM_NAME	VARCHAR(128)	
INV_DROPALT	CHAR(1)	
INV_REVOKE	CHAR(1)	
INV_LRU	CHAR(1)	
INV_RUNSTATS	CHAR(1)	
CACHED_TS	TIMESTAMP	
USERS	INTEGER	
COPIES	INTEGER	
LINES	INTEGER	
PRIMAUTH	VARCHAR(128)	
CURSQLID	VARCHAR(128)	
BIND_QUALIFIER	VARCHAR(128)	
BIND_ISO	CHAR(2)	
BIND_CDATA	CHAR(1)	
BIND_DYNRL	CHAR(1)	
BIND_DEGRE	CHAR(1)	
BIND_SQLRL	CHAR(1)	
BIND_CHOLD	CHAR(1)	
STAT_TS	TIMESTAMP	
STAT_EXEC	INTEGER	
STAT_GPAG	INTEGER	
STAT_SYNR	INTEGER	
STAT_WRIT	INTEGER	
STAT_EROW	INTEGER	
STAT_PROW	INTEGER	
STAT_SORT	INTEGER	
STAT_INDX	INTEGER	
STAT_RSCN	INTEGER	
STAT_PGRP	INTEGER	
STAT_ELAP	FLOAT	
STAT_CPU	FLOAT	
STAT_SUS_SYNIO	FLOAT	
STAT_SUS_LOCK	FLOAT	
STAT_SUS_SWIT	FLOAT	
STAT_SUS_GLCK	FLOAT	
STAT_SUS_OTHR	FLOAT	
STAT_SUS_OTHW	FLOAT	
STAT_RIDLIMT	INTEGER	
STAT_RIDSTOR	INTEGER	
EXPLAIN_TS	TIMESTAMP	1
SCHEMA	VARCHAR(128)	
STMT_TEXT	CLOB(2M)	
STMT_ROWID	ROWID	
BIND_RA_TOT	INTEGER	
BIND_RO_TYPE	CHAR(1)	

DSN_STRUCT_TABLE
Contains information about the query blocks in a query.

		D
QUERYNO	INTEGER	1
QBLOCKNO	SMALLINT	
APPLNAME	VARCHAR(24)	
PROGNAME	VARCHAR(128)	
PARENT	SMALLINT	
TIMES	FLOAT	
ROWCOUNT	INTEGER	
ATOPEN	CHAR(1)	
CONTEXT	CHAR(10)	
ORDERNO	SMALLINT	
DOATOPEN_PARENT	SMALLINT	
QBLOCK_TYPE	CHAR(6)	
EXPLAIN_TIME	TIMESTAMP	2
QUERY_STAGE	CHAR(8)	
GROUP_MEMBER	VARCHAR(24)	

Query 1:

```
EXPLAIN PLAN SET
QUERYNO=1 FOR

SELECT *

FROM   DSN8710.EMP

WHERE  EMPNO = ?

OR (EMPNO = ? AND

      LASTNAME >= ?)

OR EMPNO > ?

ORDER BY EMPNO, LASTNAME;
```

This example shows the result of adding a redundant Boolean term predicate to a query.

Query 1 contains the originally scrolling cursor, and query 2 the improved cursor with a redundant predicate.

Query 2:

```
EXPLAIN PLAN SET QUERYNO=
2 FOR

SELECT *

FROM DSN8710.EMP

WHERE  (EMPNO = ?

OR (EMPNO = ? AND

      LASTNAME >= ?)

OR EMPNO > ?)

AND EMPNO >= ?

ORDER BY EMPNO, LASTNAME;
```

Here on line 9 of the explain query output we can see the improvement in query performance with the additional Boolean term predicate resulting in an index match of 1 column.

The EXPLAIN query output demonstrates the improvement.

PROGNAME	QQP	MTH	P_NO	MJC	TBCREATOR	TBNAME	CORR_NM	ROWS_POST_FILTER	ATYP	A_NM	IXO	MIX	MCOL	STAGE	FF	BT	PRED_TEXT30	NJ_CUJOG	PF	CFE	PGRNG	JT	QB_TYP	P_QB	TB_TYP	B_TM
DSNESM68	00001-01-01	0	2	---	DSN8710	EMP	--------	15.3	I	XEMP1	N	0	0	STAGE1	.0238095223	N	DSN8710.EMP.EMPNO=(EXPR)	NNNNN					SELECT	0	T	2007-09-19-10.35.47.090000
DSNESM68	00001-01-01	0	4	---	DSN8710	EMP	--------	15.3	I	XEMP1	N	0	0	STAGE1	.0238095223	N	DSN8710.EMP.EMPNO=(EXPR)	NNNNN					SELECT	0	T	2007-09-19-10.35.47.090000
DSNESM68	00001-01-01	0	5	---	DSN8710	EMP	--------	15.3	I	XEMP1	N	0	0	STAGE1	.3333333134	N	DSN8710.EMP.LASTNAME>=(EXPR)	NNNNN					SELECT	0	T	2007-09-19-10.35.47.090000
DSNESM68	00001-01-01	0	6	---	DSN8710	EMP	--------	15.3	I	XEMP1	N	0	0	STAGE1	.3333333134	N	DSN8710.EMP.EMPNO>(EXPR)	NNNNN					SELECT	0	T	2007-09-19-10.35.47.090000
DSNESM68	00002-01-01	0	3	---	DSN8710	EMP	--------	5.1	I	XEMP1	N	0	1	STAGE1	.0238095223	N	DSN8710.EMP.EMPNO=(EXPR)	NNNNN					SELECT	0	T	2007-09-19-10.35.47.100000
DSNESM68	00002-01-01	0	5	---	DSN8710	EMP	--------	5.1	I	XEMP1	N	0	1	STAGE1	.0238095223	N	DSN8710.EMP.EMPNO=(EXPR)	NNNNN					SELECT	0	T	2007-09-19-10.35.47.100000
DSNESM68	00002-01-01	0	6	---	DSN8710	EMP	--------	5.1	I	XEMP1	N	0	1	STAGE1	.3333333134	N	DSN8710.EMP.LASTNAME>=(EXPR)	NNNNN					SELECT	0	T	2007-09-19-10.35.47.100000
DSNESM68	00002-01-01	0	7	---	DSN8710	EMP	--------	5.1	I	XEMP1	N	0	1	STAGE1	.3333333134	N	DSN8710.EMP.EMPNO>(EXPR)	NNNNN					SELECT	0	T	2007-09-19-10.35.47.100000
DSNESM68	00002-01-01	0	8	---	DSN8710	EMP	--------	5.1	I	XEMP1	N	0	1	MATCHING	.3333333134	Y	DSN8710.EMP.EMPNO>=(EXPR)	NNNNN					SELECT	0	T	2007-09-19-10.35.47.100000

QQP	Query number, block, and plan step
MTH	Method
P_NO	Predicate number
MJC	Merge join columns
ATYP	Access type
A_NM	Access name
IXO	Index only
MIX	Multi-index access seq. number
MCOL	Matching columns
FF	Filter factor
BT	Boolean term predicate
NJ_CUJOG	Sorts
PF	Prefetch
CFE	Column function evaluation
PGRNG	Page range screening
JT	Join type
QB_TYP	Query block type
P_QB	Parent query block
TB_TYP	Table type
B_TM	Bind timestamp

Explain this statement, then run the query above to produce a report that looks like this.

```
SELECT A.FIRSTNME, A.LASTNAME, B.DEPTNAME

FROM   DSN8710.EMP A

INNER JOIN

DSN8710.DEPT B

ON A.WORKDEPT = B.DEPTNO

WHERE  A.EMPNO = '000010';
```

Number rows expected from the table after local predicates are applied

Column for each predicate indicating if it is an index matching, index screening, stage 1, or stage 2 predicate

Predicate filter factor

This column contains predicate text, including rewritten and generated predicates. You can expand the length in the explain query for more detail of large or compound predicates

Sort operations for the new table of a join, or for union (duplicate), join, order, or grouping of a composite table

PROGNAME	QQP	MTH	P_NO	MJC	TBCREATOR	TBNAME	CORR_NM	ROWS_POST_FILTER	ATYP	A_NM	IXO	MIX	MCOL	STAGE	FF	BT	PRED_TEXT30	NJ_CUJOG	PF	CFE	PGRNG	JT	QB_TYP	P_QB	TB_TYP	B_TM
DSNESM68	00003-01-01	0	2	----	DSN8710	EMP	A	1.0	I	XEMP1	N	0	1	MATCHING	0.0238095223	Y	A.EMPNO='000010'	NNNNN					SELECT	0	T	2007-08-13 15:47:24.150000
DSNESM68	00003-01-02	1	3	----	DSN8710	DEPT	B	14.0	I	XDEPT1	N	0	1	MATCHING	0.0714285969	Y	A.WORKDEPT=B.DEPTNO	NNNNN					SELECT	0	T	2007-08-13 15:47:24.150000

LEGEND

Explain Query Output
Explain Query Example
Column Names
Explain Table Layout
Queries Being Explained