**常见Oracle HINT的用法**

1. /\*+ALL\_ROWS\*/

　　表明对语句块选择基于开销的优化方法,并获得最佳吞吐量,使资源消耗最小化.

　　例如:

　　SELECT /\*+ALL+\_ROWS\*/ EMP\_NO,EMP\_NAM,DAT\_IN FROM BSEMPMS WHERE EMP\_NO='SCOTT';

　　2. /\*+FIRST\_ROWS\*/

　　表明对语句块选择基于开销的优化方法,并获得最佳响应时间,使资源消耗最小化.

　　例如:

　　SELECT /\*+FIRST\_ROWS\*/ EMP\_NO,EMP\_NAM,DAT\_IN FROM BSEMPMS WHERE EMP\_NO='SCOTT';

　　3. /\*+CHOOSE\*/

　　表明如果数据字典中有访问表的统计信息,将基于开销的优化方法,并获得最佳的吞吐量;

　　表明如果数据字典中没有访问表的统计信息,将基于规则开销的优化方法;

　　例如:

　　SELECT /\*+CHOOSE\*/ EMP\_NO,EMP\_NAM,DAT\_IN FROM BSEMPMS WHERE EMP\_NO='SCOTT';

　　4. /\*+RULE\*/

　　表明对语句块选择基于规则的优化方法.

　　例如:

　　SELECT /\*+ RULE \*/ EMP\_NO,EMP\_NAM,DAT\_IN FROM BSEMPMS WHERE EMP\_NO='SCOTT';

　　5. /\*+FULL(TABLE)\*/

　　表明对表选择全局扫描的方法.

　　例如:

　　SELECT /\*+FULL(A)\*/ EMP\_NO,EMP\_NAM FROM BSEMPMS A WHERE EMP\_NO='SCOTT';

　　6. /\*+ROWID(TABLE)\*/

　　提示明确表明对指定表根据ROWID进行访问.

　　例如:

　　SELECT /\*+ROWID(BSEMPMS)\*/ \* FROM BSEMPMS WHERE ROWID>='AAAAAAAAAAAAAA'

　　AND EMP\_NO='SCOTT';

　　7. /\*+CLUSTER(TABLE)\*/

　　提示明确表明对指定表选择簇扫描的访问方法,它只对簇对象有效.

　　例如:

　　SELECT /\*+CLUSTER \*/ BSEMPMS.EMP\_NO,DPT\_NO FROM BSEMPMS,BSDPTMS

　　WHERE DPT\_NO='TEC304' AND BSEMPMS.DPT\_NO=BSDPTMS.DPT\_NO;

　　8. /\*+INDEX(TABLE INDEX\_NAME)\*/

　　表明对表选择索引的扫描方法.

　　例如:

　　SELECT /\*+INDEX(BSEMPMS SEX\_INDEX) USE SEX\_INDEX BECAUSE THERE ARE FEWMALE BSEMPMS \*/ FROM BSEMPMS WHERE SEX='M';

　　9. /\*+INDEX\_ASC(TABLE INDEX\_NAME)\*/

　　表明对表选择索引升序的扫描方法.

　　例如:

　　SELECT /\*+INDEX\_ASC(BSEMPMS PK\_BSEMPMS) \*/ FROM BSEMPMS WHERE DPT\_NO='SCOTT';

　　10. /\*+INDEX\_COMBINE\*/

　　为指定表选择位图访问路经,如果INDEX\_COMBINE中没有提供作为参数的索引,将选择出位图索引的布尔组合方式.

　　例如:

　　SELECT /\*+INDEX\_COMBINE(BSEMPMS SAL\_BMI HIREDATE\_BMI)\*/ \* FROM BSEMPMS

　　WHERE SAL<5000000 AND HIREDATE

　　11. /\*+INDEX\_JOIN(TABLE INDEX\_NAME)\*/

　　提示明确命令优化器使用索引作为访问路径.

　　例如:

　　SELECT /\*+INDEX\_JOIN(BSEMPMS SAL\_HMI HIREDATE\_BMI)\*/ SAL,HIREDATE

　　FROM BSEMPMS WHERE SAL<60000;

　　12. /\*+INDEX\_DESC(TABLE INDEX\_NAME)\*/

　　表明对表选择索引降序的扫描方法.

　　例如:

　　SELECT /\*+INDEX\_DESC(BSEMPMS PK\_BSEMPMS) \*/ FROM BSEMPMS WHERE DPT\_NO='SCOTT';

　　13. /\*+INDEX\_FFS(TABLE INDEX\_NAME)\*/

　　对指定的表执行快速全索引扫描,而不是全表扫描的办法.

　　例如:

　　SELECT /\*+INDEX\_FFS(BSEMPMS IN\_EMPNAM)\*/ \* FROM BSEMPMS WHERE DPT\_NO='TEC305';

　　14. /\*+ADD\_EQUAL TABLE INDEX\_NAM1,INDEX\_NAM2,...\*/

　　提示明确进行执行规划的选择,将几个单列索引的扫描合起来.

　　例如:

　　SELECT /\*+INDEX\_FFS(BSEMPMS IN\_DPTNO,IN\_EMPNO,IN\_SEX)\*/ \* FROM BSEMPMS WHERE EMP\_NO='SCOTT' AND DPT\_NO='TDC306';

　　15. /\*+USE\_CONCAT\*/

　　对查询中的WHERE后面的OR条件进行转换为UNION ALL的组合查询.

　　例如:

　　SELECT /\*+USE\_CONCAT\*/ \* FROM BSEMPMS WHERE DPT\_NO='TDC506' AND SEX='M';

　　16. /\*+NO\_EXPAND\*/

　　对于WHERE后面的OR 或者IN-LIST的查询语句,NO\_EXPAND将阻止其基于优化器对其进行扩展.

　　例如:

　　SELECT /\*+NO\_EXPAND\*/ \* FROM BSEMPMS WHERE DPT\_NO='TDC506' AND SEX='M';

　　17. /\*+NOWRITE\*/

　　禁止对查询块的查询重写操作.

　　18. /\*+REWRITE\*/

　　可以将视图作为参数.

　　19. /\*+MERGE(TABLE)\*/

　　能够对视图的各个查询进行相应的合并.

　　例如:

　　SELECT /\*+MERGE(V) \*/ A.EMP\_NO,A.EMP\_NAM,B.DPT\_NO FROM BSEMPMS A (SELET DPT\_NO

　　,AVG(SAL) AS AVG\_SAL FROM BSEMPMS B GROUP BY DPT\_NO) V WHERE A.DPT\_NO=V.DPT\_NO

　　AND A.SAL>V.AVG\_SAL;

　　20. /\*+NO\_MERGE(TABLE)\*/

　　对于有可合并的视图不再合并.

　　例如:

　　SELECT /\*+NO\_MERGE(V) \*/ A.EMP\_NO,A.EMP\_NAM,B.DPT\_NO FROM BSEMPMS A (SELECT DPT\_NO,AVG(SAL) AS AVG\_SAL FROM BSEMPMS B GROUP BY DPT\_NO) V WHERE A.DPT\_NO=V.DPT\_NO AND A.SAL>V.AVG\_SAL;

　　21. /\*+ORDERED\*/

　　根据表出现在FROM中的顺序,ORDERED使ORACLE依此顺序对其连接.

　　例如:

　　SELECT /\*+ORDERED\*/ A.COL1,B.COL2,C.COL3 FROM TABLE1 A,TABLE2 B,TABLE3 C WHERE A.COL1=B.COL1 AND B.COL1=C.COL1;

　　22. /\*+USE\_NL(TABLE)\*/

　　将指定表与嵌套的连接的行源进行连接,并把指定表作为内部表.

　　例如:

　　SELECT /\*+ORDERED USE\_NL(BSEMPMS)\*/ BSDPTMS.DPT\_NO,BSEMPMS.EMP\_NO,BSEMPMS.EMP\_NAM FROM BSEMPMS,BSDPTMS WHERE BSEMPMS.DPT\_NO=BSDPTMS.DPT\_NO;

　　23. /\*+USE\_MERGE(TABLE)\*/

　　将指定的表与其他行源通过合并排序连接方式连接起来.

　　例如:

　　SELECT /\*+USE\_MERGE(BSEMPMS,BSDPTMS)\*/ \* FROM BSEMPMS,BSDPTMS WHERE BSEMPMS.DPT\_NO=BSDPTMS.DPT\_NO;

　　24. /\*+USE\_HASH(TABLE)\*/

　　将指定的表与其他行源通过哈希连接方式连接起来.

　　例如:

　　SELECT /\*+USE\_HASH(BSEMPMS,BSDPTMS)\*/ \* FROM BSEMPMS,BSDPTMS WHERE BSEMPMS.DPT\_NO=BSDPTMS.DPT\_NO;

　　25. /\*+DRIVING\_SITE(TABLE)\*/

　　强制与ORACLE所选择的位置不同的表进行查询执行.

　　例如:

　　SELECT /\*+DRIVING\_SITE(DEPT)\*/ \* FROM BSEMPMS,DEPT@BSDPTMS WHERE BSEMPMS.DPT\_NO=DEPT.DPT\_NO;

　　26. /\*+LEADING(TABLE)\*/

　　将指定的表作为连接次序中的首表.

　　27. /\*+CACHE(TABLE)\*/

　　当进行全表扫描时,CACHE提示能够将表的检索块放置在缓冲区缓存中最近最少列表LRU的最近使用端

　　例如:

　　SELECT /\*+FULL(BSEMPMS) CAHE(BSEMPMS) \*/ EMP\_NAM FROM BSEMPMS;

　　28. /\*+NOCACHE(TABLE)\*/

　　当进行全表扫描时,CACHE提示能够将表的检索块放置在缓冲区缓存中最近最少列表LRU的最近使用端

　　例如:

　　SELECT /\*+FULL(BSEMPMS) NOCAHE(BSEMPMS) \*/ EMP\_NAM FROM BSEMPMS;

　　29. /\*+APPEND\*/

　　直接插入到表的最后,可以提高速度.

　　insert /\*+append\*/ into test1 select \* from test4 ;

　　30. /\*+NOAPPEND\*/

　　通过在插入语句生存期内停止并行模式来启动常规插入.

　　insert /\*+noappend\*/ into test1 select \* from test4 ;