ACOUG
All China Oracle User Group
中国 Oracle 用户组

2016
ACOUG Asia Tour

从分区到Sharding
数据库核心业务表的分区设计

杨廷琨

个人介绍

- 杨廷琨(yangtingkun)
  - Oracle ACE Director
  - ITPUB数据库管理区版主
  - ACOUG核心会员
  - 参与编写《Oracle数据库性能优化》、《Oracle DBA手记》和《Oracle DBA手记3》
  - 十六年的一线DBA经验
  - 个人BLOG中积累了2500篇原创技术文章
  - 云和恩墨CTO

目录

- 分区与Sharding
- 分区概述
- 分区设计案例
- 12.2分区新特性


11.11
全球狂欢节
天猫·2015
为什么说双11是门技术活


订单创建 / 秒
140,000
单 / 秒
80,000
42,000
14,000
3,200
1,000
400
支付笔数 / 秒
85,900
笔 / 秒
38,000
15,000
3,850
1,200
500
200
2015
2014
2013
2012
2011
2010
2009

10.2 压力测试

	Per Second	Per Transaction
Redo size:	23,757,440.30	2,513.89
Logical reads:	268,364.75	28.40
Block changes:	99,948.39	10.58
Physical reads:	5.02	0.00
Physical writes:	1,693.37	0.18
User calls:	1.93	0.00
Parses:	27.53	0.00
Hard parses:	1.81	0.00
Sorts:	5.16	0.00
Logons:	0.05	0.00
Executes:	47,303.23	5.01
Transactions:	9,450.48	

11.2 真实环境

	Per Second	Per Transaction	Per Exec	Per Call
DB Time(s):	92.0	0.0	0.00	0.00
DB CPU(s):	21.2	0.0	0.00	0.00
Redo size (bytes):	27,576,579.2	1,460.3		
Logical read (blocks):	379,436.5	20.1		
Block changes:	158,659.1	8.4		
Physical read (blocks):	5,973.3	0.3		
Physical write (blocks):	6,955.8	0.4		
Read IO requests:	4,837.9	0.3		
Write IO requests:	4,605.4	0.2		
Read IO (MB):	46.7	0.0		
Write IO (MB):	54.3	0.0		
User calls:	60,341.9	3.2		
Parses (SQL):	29,585.5	1.6		
Hard parses (SQL):	2.3	0.0		
SQL Work Area (MB):	0.4	0.0		
Logons:	4.1	0.0		
Executes (SQL):	29,671.6	1.6		
Rollbacks:	0.0	0.0		
Transactions:	18,884.0			

12.1 Exadata压测

	Per Second	Per Transaction	Per Exec	Per Call
DB Time(s):	55.3	0.0	0.00	0.00
DB CPU(s):	30.8	0.0	0.00	0.00
Background CPU(s):	2.3	0.0	0.00	0.00
Redo size (bytes):	176,438,164.5	6,781.1		
Logical read (blocks):	2,691,588.2	103.5		
Block changes:	992,503.4	38.2		
Physical read (blocks):	496.4	0.0		
Physical write (blocks):	3,914.2	0.2		
Read IO requests:	496.2	0.0		
Write IO requests:	561.7	0.0		
Read IO (MB):	3.9	0.0		
Write IO (MB):	30.6	0.0		
IM scan rows:	0.0	0.0		
Session Logical Read IM:				
Global Cache blocks received:	8,295.1	0.3		
Global Cache blocks served:	8,358.2	0.3		
User calls:	28,851.9	1.1		
Parses (SQL):	739.1	0.0		
Hard parses (SQL):	0.4	0.0		
SQL Work Area (MB):	25.1	0.0		
Logons:	0.4	0.0		
Executes (SQL):	176,367.2	6.8		
Rollbacks:	1,249.3	0.1		
Transactions:	26,019.1			

	Per Second	Per Transaction	Per Exec	Per Call
DB Time(s):	55.3	0.0	0.00	0.00
DB CPU(s):	32.3	0.0	0.00	0.00
Background CPU(s):	2.2	0.0	0.00	0.00
Redo size (bytes):	184,175,416.1	6,824.1		
Logical read (blocks):	2,824,663.0	104.7		
Block changes:	1,031,202.8	38.2		
Physical read (blocks):	518.3	0.0		
Physical write (blocks):	3,994.5	0.2		
Read IO requests:	477.3	0.0		
Write IO requests:	452.8	0.0		
Read IO (MB):	4.1	0.0		
Write IO (MB):	31.2	0.0		
IM scan rows:	0.0	0.0		
Session Logical Read IM:				
Global Cache blocks received:	8,427.0	0.3		
Global Cache blocks served:	8,426.2	0.3		
User calls:	29,840.7	1.1		
Parses (SQL):	780.8	0.0		
Hard parses (SQL):	0.2	0.0		
SQL Work Area (MB):	26.3	0.0		
Logons:	0.4	0.0		
Executes (SQL):	182,637.0	6.8		
Rollbacks:	1,370.3	0.1		
Transactions:	26,989.1			

Transactions: 26,193.5

Transactions: 26,202.4

Transactions: 26,358.1

Transactions: 25,769.9

Transactions: 26,494.1

Transactions: 25,262.8

Oracle vs. 天猫

版本	描述	事务数	天猫交易

Sharding

- Oracle Sharding
  - 数据垂直分区到多个独立的数据库中
  - 线性扩展
  - 自动部署
  - 自动Rebalance和Resharding
  - 支持HASH、RANGE、LIST和复合方式的自动数据分区

目录

- 分区与Sharding
- 分区概述
- 分区设计案例
- 12.2分区新特性

分区概述

- 定义
  - 根据内部定义的规则，将一张表的数据拆分到多个数据段中。
  - 对应用透明，程序可以不做任何额外调整
  - 可以通过分区列上的条件访问指定分区的数据，也可以通过分区扩展语句显式的访问
  - 分区级别上提供删除，截断，迁移，索引的能力

分区概述

- 分区的优点
  - 可维护性
  - 可用性增强
  - OLTP：降低共享资源争用
  - OLAP：提升查询性能

Oracle分区演进历史

版本	功能	性能	管理性
Oracle 8.0	范围分区 本地、全局范围索引	静态分区剪裁	基本维护功能：ADD、DROP、EXCHANGE
Oracle 8.1	哈希分区 范围哈希分区	分区智能连接 动态分区剪裁	扩展维护功能：MERGE
Oracle 9.0	列表分区		全局索引维护
Oracle 9.2	范围列表分区	快速分区SPLIT	
Oracle 10.1	全局哈希索引分区		本地索引维护
Oracle 10.2	单表允许1百万分区	多维度分区剪裁	快速DROP TABLE
Oracle 11.1	虚拟列分区 多重复合分区方式 参考分区		间隔分区 分区建议 增量统计信息收集
Oracle 11.2	HASH复合分区 扩展参考分区	“AND”分区剪裁	多分支执行
Oracle 12.1	间隔参考分区	多个分区的分区维护操作 异步全局索引维护	在线分区MOVE 级联TRUNCATE 部分分区索引

目录

- 分区与Sharding
- 分区概述
- 分区设计案例
- 12.2分区新特性

范围分区

- 适用场景
  - 时间属性
  - 关注近期数据
- 最佳实践
  - 数据生命周期和访问范围
  - 明确时间条件限定
  - 定期清理过期分区
  - 非主键优先本地索引
- 优势
  - 数据分布平均，分区数量可控
  - DDL清理过期数据
  - 查询仅访问个别分区
  - 表大小相对稳定
  - 索引大小相对稳定

范围分区

- 面临挑战
  - 分区清理逻辑复杂
  - DELETE效率低下
  - DELETE无法释放空间
  - 数据量迅速膨胀
  - 分区数量不断增长

- 解决方案
  - INSERT + EXCHANGE方式
  - MERGE分区减少分区数量
  - 避免DELETE效率低下
  - 避免空间无法释放

范围分区

```
SQL> SELECT TEMPORARY, COUNT(*) FROM T_PART PARTITION (P3) GROUP BY TEMPORARY;

T   COUNT(*)
- ----------
Y         30
N      87261


SQL> LOCK TABLE T_PART PARTITION (P3) IN EXCLUSIVE MODE;

Table(s) Locked.

SQL> INSERT INTO T_INTER SELECT * FROM T_PART PARTITION (P3) WHERE TEMPORARY = 'Y';

30 rows created.

SQL> ALTER TABLE T_PART EXCHANGE PARTITION P3 WITH TABLE T_INTER;

Table altered.

SQL> SELECT TEMPORARY, COUNT(*) FROM T_PART PARTITION (P3) GROUP BY TEMPORARY;

T   COUNT(*)
- ----------
Y         30
```

范围分区

- 面临挑战
  - 子表不包含分区时间列
  - 主子表时间列字段含义不同
  - 数据清理破坏主子表依赖关系
  - 子表需要冗余主表分区字段
  - 主外键约束禁止DDL操作

- 解决方案
  - 主表时间字段范围分区
  - 子表外键建立参考分区
  - 参考分区不会破坏主外键依赖关系

参考分区

- 适用场景
  - 主子表采用相同的数据清理策略
  - 子表没有合适的分区字段
  - 主子表经常关联访问

```
SQL> CREATE TABLE T_PRIMARY (ID NUMBER PRIMARY KEY, NAME VARCHAR2(128), CREATED DATE)
  2  PARTITION BY RANGE (CREATED)
  3  (PARTITION P1 VALUES LESS THAN (TO_DATE('201512', 'YYYYMM')),
  4  PARTITION P2 VALUES LESS THAN (MAXVALUE));

Table created.

SQL> CREATE TABLE T_FOREIGN (ID NUMBER, FID NUMBER NOT NULL, NAME VARCHAR2(128),
  2  CONSTRAINT FK_FID FOREIGN KEY (FID) REFERENCES T_PRIMARY (ID))
  3  PARTITION BY REFERENCE (FK_FID);

Table created.
```

哈希分区

- 适用场景
  - 没有时间属性
  - 缺少区分数据的业务字段
- 最佳实践
  - 分区键值列选择重复度不高的字段
  - 分区数量应为2的幂
  - 多创建全局索引
  - 哈希分区索引可解决索引热点块问题
- 优势
  - 分区没有业务特点的数据
  - 数据均匀分布
  - 有效解决递增索引的热点块问题

哈希全局分区索引

- 批量插入导致索引热点块争用
- RAC全局等待加重热点块问题
- 逆键索引不支持范围扫描

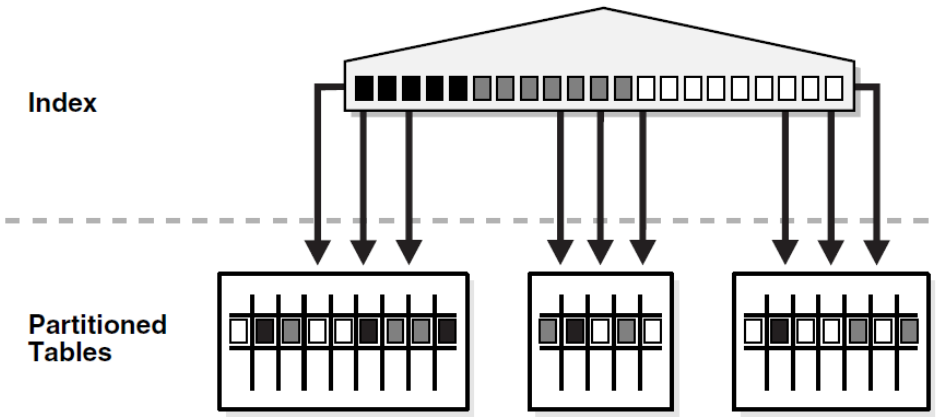
```
SQL> CREATE TABLE T_PART (ID NUMBER, NAME VARCHAR2(30), CREATED DATE);

Table created.

SQL> CREATE INDEX IND_PART_CREATED ON T_PART(CREATED) GLOBAL
  2  PARTITION BY HASH (CREATED)
  3  PARTITIONS 32;

Index created.
```

列表分区

- 适用场景
  - 地域，类型等有限的业务属性
  - 访问一种或几种业务属性
  - 通过业务属性可平均拆分数据
- 最佳实践
  - 地区字段是常见候选
  - 数据分布和访问方式确定分区键值划分
  - 设定DEFAULT分区
- 优势
  - 分区方式和业务匹配度更好
  - 数据如何在分区中存放的选择度更高
  - 分区键值与分区的对应更加明确

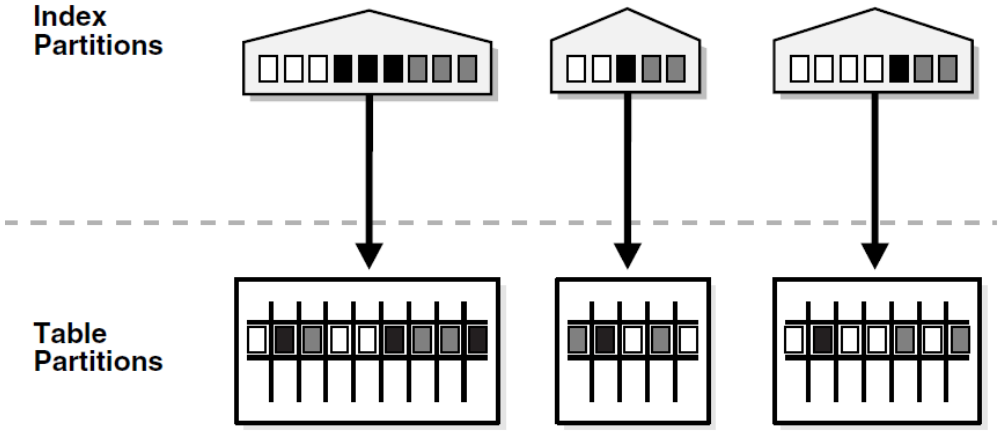
索引选择

- 全局索引
  - 索引扫描不会导致逻辑读增加
  - 大部分分区维护操作会导致索引失效
  - 对于并行执行没有帮助



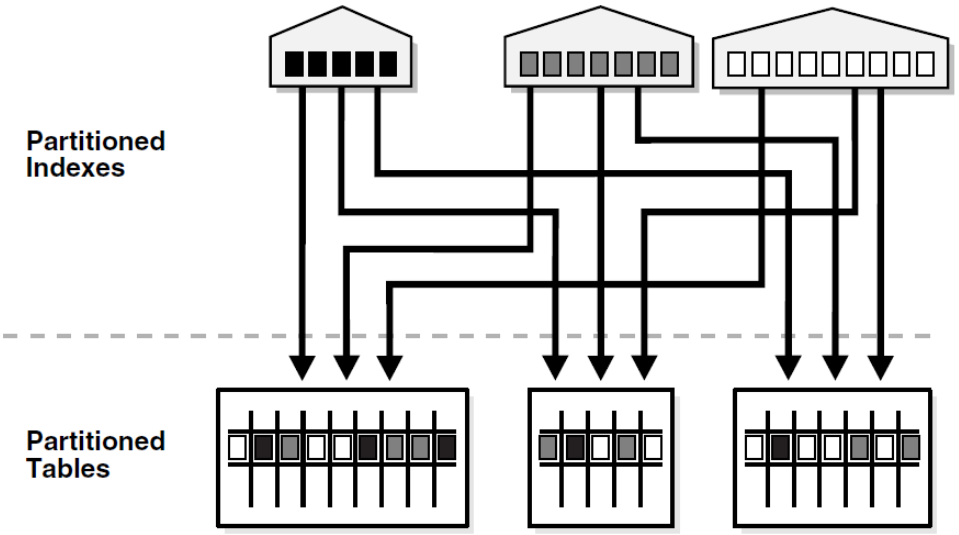
索引选择

- 本地索引
  - 大部分分区维护操作不会导致索引失效
  - 支持并行扫描和创建
  - 唯一索引必须包含分区键值列
  - 未限定分区的查询将扫描全部索引分区



索引选择

- 全局哈希分区索引
  - 具有分散热点数据的能力
  - 索引范围扫描将扫描全部索引分区



目录

- 分区与Sharding
- 分区概述
- 分区设计案例
- 12.2分区新特性

Oracle 12.2新特性

版本	功能	性能	管理性
Oracle 12.2	多列列表分区	多种在线分区操作	过滤分区维护操作
	自动列表分区	非分区表与分区表在线转换	只读分区
	间隔子分区	DDL操作减少游标失效	创建交换分区表
	外部表分区		

Oracle 12.2新特性

- 自动列表分区

```
SQL> CREATE TABLE T_LIST_AUTO (ID NUMBER, NAME VARCHAR2(30), TYPE VARCHAR2(30))
  2  PARTITION BY LIST (TYPE) AUTOMATIC
  3  (PARTITION P1 VALUES ('TABLE'));

Table created.

SQL> INSERT INTO T_LIST_AUTO VALUES (1, 'V_VIEW', 'VIEW');

1 row created.

SQL> SELECT TABLE_NAME, PARTITION_NAME, HIGH_VALUE FROM USER_TAB_PARTITIONS WHERE TABLE_NAME =
'T_LIST_AUTO';

TABLE_NAME      PARTITION_NAME  HIGH_VALUE
--------------- --------------- ------------------------------
T_LIST_AUTO     P1              'TABLE'
T_LIST_AUTO     SYS_P1366       'VIEW'
```

Oracle 12.2新特性

- 间隔子分区

```
SQL> CREATE TABLE T_INTER_SUBPART (ID NUMBER, CREATED DATE, TYPE VARCHAR2(30))
  2  PARTITION BY LIST (TYPE) SUBPARTITION BY RANGE (CREATED) INTERVAL (NUMTOYMINTERVAL(1, 'MONTH'))
  3  SUBPARTITION TEMPLATE (SUBPARTITION SP1 VALUES LESS THAN (TO_DATE('20151101', 'YYYYMMDD')))
  4  (PARTITION P1 VALUES ('TABLE'));

Table created.

SQL> INSERT INTO T_INTER_SUBPART VALUES (1, SYSDATE, 'TABLE');

1 row created.

SQL> SELECT SUBPARTITION_NAME, HIGH_VALUE FROM USER_TAB_SUBPARTITIONS WHERE TABLE_NAME =
'T_INTER_SUBPART';

SUBPARTITION_NAME    HIGH_VALUE
-------------------- --------------------------------------------------------------------------------
P1_SP1               TO_DATE(' 2015-11-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIA
SYS_SUBP1367         TO_DATE(' 2015-12-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIA
```

Oracle 12.2新特性

- 非分区表在线转换分区表

```
SQL> CREATE TABLE T_ONLINE_PART (ID NUMBER, NAME VARCHAR2(128), CREATED DATE);

Table created.

SQL> INSERT INTO T_ONLINE_PART SELECT ROWNUM, OBJECT_NAME, CREATED FROM DBA_OBJECTS;

94154 rows created.

SQL> COMMIT;

Commit complete.

SQL> ALTER TABLE T_ONLINE_PART MODIFY PARTITION BY RANGE (CREATED)
  2  (PARTITION P1 VALUES LESS THAN (TO_DATE('201512', 'YYYYMM')),
  3  PARTITION P2 VALUES LESS THAN (TO_DATE('201601', 'YYYYMM')),
  4  PARTITION PMAX VALUES LESS THAN (MAXVALUE)) ONLINE;

Table created.
```