

01 RrecordId

[转至元数据结尾](#)

- 由 [郑生俊](#) 创建, 最后修改于 [九月 12, 2018](#)

[转至元数据起始](#)

- [一 介绍](#)
 - [1.产生](#)
 - [2.说明](#)
- [二 RrecordId存储](#)
 - [1. Hbase表结构](#)
 - [2. 举例](#)
- [三 RrecordId使用场景](#)

一 介绍

Hive支持ACID的一个基础就是RrecordId, RrecordId是一个**记录**和**对该记录操作历史**的唯一标识。

1.产生

在向ACID表插入数据时，RrecordId由Hive产生，由 $\${TransactionID} + \${BucketID} + \${RowId}$ 组成，其中RowId是一个事务内针对每条插入的记录从零开始自增的ID。
在更新操作时，必须由客户端提供之前插入数据时，对应的RrecordId。

2.说明

之所以说RrecordId是对记录和记录操作历史的唯一标示，因为它包含记录的主键和对记录增删改操作对应的事务信息。

从上面的组成看应该会有以下疑问：

对记录增删改操作对应的事务信息就是 $\${TransactionID}$ ，那记录的主键信息呢？在Mysql中记录的主键信息是由表的字段组成，并不是事务内从零开始自增的RowId

因此在这里我们需要做额外操作：

将事务内从零开始自增的RowId，和Mysql中的主键信息关联并记录到Hbase中，在更新时用到。

二 RrecordId存储

1. Hbase表结构

rowKey	f:recordId	f2:lastUpdateTime
{database_name}. {table_name}_\${business_id}	transactionId_bucketId_RowId	记录中的更新时间毫秒数

business_id : Mysql主键的值（通常情况只有一个主键ID，取ID这一列的值。多个主键的情况，按照主键列名排序，然后取值用下划线 _ 拼接起来）

bucketId: 我们设计所有的表只有一个bucket，因此bucketId固定为0；

2. 举例

Mysql表：Testdb.TestTable 中的数据

ID1(primary_key)	ID2(primary_key)	col3
ID1(primary_key)	ID2(primary_key)	col3
id1	id2	xxx
id11	id22	xxx
id33	id33	xxx

如果在transaction id为1 的事务中插入到了hive表，对应Hbase中存储的的RecordId信息：

rowKey	f:recordId	f:lastUpdateTime
Testdb.TestTable_id1_id2	1_0_1	时间毫秒数
Testdb.TestTable_id11_id22	1_0_2	时间毫秒数
Testdb.TestTable_id33_id33	1_0_3	时间毫秒数

三 RrecordId使用场景

- 1. Update Delete时必须提供对应记录在插入时产生的RrecordId
- 2. 数据检查时根据RowKey规则，拼出记录对应的rowKey，从Hbase中获取记录的更新时间作为数据版本。