MySQL "replace into" 的坑

2014-10-09 17:00 by 默念默, 6008 阅读, 0 评论, 收藏, 编辑

MySQL 对 SQL 有很多扩展，有些用起来很方便，但有一些被误用之后会有性能问题，还会有一些意料之外的副作用，比如 REPLACE INTO。

比如有这样一张表：

```
CREATE TABLE `auto` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `k` int(10) unsigned NOT NULL,
  `v` varchar(100) DEFAULT NULL,
  `extra` varchar(200) DEFAULT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `uk_k` (`k`)
) ENGINE=InnoD
```

auto 表有一个自增的 id 字段作为主键，字段 k 有 UNIQUE KEY 做唯一性约束。写入几条记录之后会是这样：

```
xupeng@diggle7:3600(dba_m) [dba] mysql> INSERT INTO auto (k, v, extra) VALUES (1, '1', 'extra 1'), (2, '2', 'extra 2'), (3, '3', 'extra 3');
Query OK, 3 rows affected (0.01 sec)
Records: 3  Duplicates: 0  Warnings: 0

xupeng@diggle7:3600(dba_m) [dba] mysql> SHOW CREATE TABLE auto\G
*************************** 1. row ***************************
       Table: auto
Create Table: CREATE TABLE `auto` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `k` int(10) unsigned NOT NULL,
  `v` varchar(100) DEFAULT NULL,
  `extra` varchar(200) DEFAULT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `uk_k` (`k`)
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=latin1
1 row in set (0.01 sec)

xupeng@diggle7:3600(dba_m) [dba] mysql> SELECT * FROM auto;
+----+---+------+---------+
| id | k | v    | extra   |
+----+---+------+---------+
|  1 | 1 | 1    | extra 1 |
|  2 | 2 | 2    | extra 2 |
|  3 | 3 | 3    | extra 3 |
+----+---+------+---------+
3 rows in set (0.00 sec)
```

在 slave 节点上是和 master 一致的：

```
xupeng@diggle8:3600(dba_s) [dba] mysql> SELECT * FROM auto;
+----+---+------+---------+
| id | k | v    | extra   |
```

About

昵称：默念默
园龄：5年11个月
粉丝：5
关注：22
+加关注

SEARCH

最新评论

日历

< 2016年8月 >

日	一	二	三	四	五	六
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

随笔分类

Asp.Net(18)
C#(6)
JS(1)
Linq(2)
SQLSERVER(15)

推荐排行榜

1. httpModules与httpHandlers(2)
2. vs2010 快捷键大全(2)
3. SQL中列转行(2)
4. 扩展性很好的一个分页存储过程(2)
5. MySQL "replace into" 的坑(2)

阅读排行榜

1. 用Quartz处理定时执行的任务(23913)
2. MySQL "replace into" 的坑(6008)
3. Linq读取XML(3246)
4. @Html.Partial,@Html.Action,@Html.RenderPartial,@Html.RenderAction区别．(转)(2731)
5. 在自定义HttpHandler中无法使用Session(1219)

随笔档案

2016年2月(1)
2015年10月(1)
2015年7月(1)
2015年6月(1)
2015年3月(1)
2015年1月(1)
2014年11月(4)
2014年10月(1)
2014年7月(5)
2014年6月(2)
2014年5月(1)
2014年4月(2)
2014年3月(1)
2013年9月(2)
2013年8月(1)
2013年7月(1)
2013年5月(1)
2012年12月(2)
2012年11月(1)
2012年10月(1)
2012年9月(1)
2012年8月(1)
2012年7月(1)
2012年5月(1)
2011年12月(1)
2011年11月(4)
2011年10月(9)
2011年9月(3)
2011年5月(1)
2011年2月(1)
2011年1月(1)
2010年10月(1)

```
+----+---+------+---------+
|  1 | 1 | 1    | extra 1 |
|  2 | 2 | 2    | extra 2 |
|  3 | 3 | 3    | extra 3 |
+----+---+------+---------+
3 rows in set (0.00 sec)

xupeng@diggle8:3600(dba_s) [dba] mysql> SHOW CREATE TABLE auto\G
*************************** 1. row ***************************
       Table: auto
Create Table: CREATE TABLE `auto` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `k` int(10) unsigned NOT NULL,
  `v` varchar(100) DEFAULT NULL,
  `extra` varchar(200) DEFAULT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `uk_k` (`k`)
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=latin1
1 row in set (0.00 sec)
```

可以看到，写入三条记录之后，auto 表的 AUTO_INCREMENT 增长为 4，也就是说下一条不手工为 id 指定值的记录，id 字段的值会是 4。

接下来使用 REPLACE INTO 来写入一条记录：

```
xupeng@diggle7:3600(dba_m) [dba] mysql> REPLACE INTO auto (k, v)
VALUES (1, '1-1');
Query OK, 2 rows affected (0.01 sec)

xupeng@diggle7:3600(dba_m) [dba] mysql> SELECT * FROM auto;
+----+---+------+---------+
| id | k | v    | extra   |
+----+---+------+---------+
|  2 | 2 | 2    | extra 2 |
|  3 | 3 | 3    | extra 3 |
|  4 | 1 | 1-1  | NULL    |
+----+---+------+---------+
3 rows in set (0.00 sec)

xupeng@diggle7:3600(dba_m) [dba] mysql> SHOW CREATE TABLE auto\G
*************************** 1. row ***************************
       Table: auto
Create Table: CREATE TABLE `auto` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `k` int(10) unsigned NOT NULL,
  `v` varchar(100) DEFAULT NULL,
  `extra` varchar(200) DEFAULT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `uk_k` (`k`)
) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=latin1
1 row in set (0.00 sec)
```

可以看到 MySQL 说 “2 rows affected”，可是明明是只写一条记录，为什么呢？这是因为 MySQL 在执行 REPLACE INTO auto (k) VALUES (1) 时首先尝试 INSERT INTO auto (k) VALUES (1)，但由于已经存在一条 k=1 的记录，发生了 duplicate key error，于是 MySQL 会先删除已有的那条 k=1 即 id=1 的记录，然后重新写入一条新的记录。

这时候 slave 上出现了诡异的问题：

```
xupeng@diggle8:3600(dba_s) [dba] mysql> SHOW CREATE TABLE auto\G
*************************** 1. row ***************************
       Table: auto
Create Table: CREATE TABLE `auto` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `k` int(10) unsigned NOT NULL,
  `v` varchar(100) DEFAULT NULL,
  `extra` varchar(200) DEFAULT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `uk_k` (`k`)
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=latin1
```

可以知道，当前表内数据 id 字段的最大值是 4，AUTO_INCREMENT 应该为 5，但在 slave 上 AUTO_INCREMENT 却并未更新，这会有什么问题呢？把这个 slave 提升为 master 之后，由于 AUTO_INCREMENT 比实际的 next id 还要小，写入新记录时就会发生 duplicate key error，每次冲突之后 AUTO_INCREMENT += 1，直到增长为 max(id) + 1 之后才能恢复正常：

```
xupeng@diggle8:3600(dba_s) [dba] mysql> REPLACE INTO auto (k, v) VALUES (4, '4');
ERROR 1062 (23000): Duplicate entry '4' for key 'PRIMARY'
xupeng@diggle8:3600(dba_s) [dba] mysql> REPLACE INTO auto (k, v) VALUES (5, '5');
Query OK, 1 row affected (0.00 sec)

xupeng@diggle8:3600(dba_s) [dba] mysql> SELECT * FROM auto;
+----+---+------+---------+
| id | k | v    | extra   |
+----+---+------+---------+
|  2 | 2 | 2    | extra 2 |
|  3 | 3 | 3    | extra 3 |
|  4 | 1 | 1-1  | NULL    |
|  5 | 5 | 5    | NULL    |
+----+---+------+---------+
4 rows in set (0.00 sec)
```

没有预料到 MySQL 在数据冲突时实际上是删掉了旧记录，再写入新记录，这是使用 REPLACE INTO 时最大的一个误区，拿之前的例子来说，执行完 REPLACE INTO auto (k, v) VALUES (1, '1-1') 之后，由于新写入记录时并未给 extra 字段指定值，原记录 extra 字段的值就「丢失」了，而通常这并非是业务上所预期的，更常见的需求实际上是，当存在 k=1 的记录时，就把 v 字段的值更新为 '1-1'，其他未指定的字段则保持原状，而满足这一需求的 MySQL 方言是 INSERT INTO auto (k, v) VALUES (1, '1-1') ON DUPLICATE KEY UPDATE v=VALUES(v);

鉴于此，很多使用 REPLACE INTO 的场景，实际上需要的是 INSERT INTO ... ON DUPLICATE KEY UPDATE，在正确理解 REPLACE INTO 行为和副作用的前提下，谨慎使用 REPLACE INTO。

好文要顶 关注我 收藏该文

默念默
关注 - 22
粉丝 - 5
+加关注

2 0

« 上一篇：SQL Server 触发器
» 下一篇：jQuery小例子

分类：SQLSERVER

刷新评论 刷新页面 返回顶部

注册用户登录后才能发表评论，请 登录 或 注册，访问网站首页。

【推荐】50万行VC++源码：大型组态工控、电力仿真CAD与GIS源码库
【链接】融云阿里云大咖面对面 ： 教你玩转直播风口
【推荐】报表开发有捷径：快速设计轻松集成，数据可视化和交互
【推荐】一个月仅用630元赚取15000元，学会投资
【推荐】网易云信-一天开发一个微信，独创1对1技术顾问让开发加速

最新IT新闻：
· OPPO下半年出货挑战5000万部 华为宝座岌岌可危
· 周鸿祎说360目前没有考虑借壳 拆分业务是为了治疗大公司病
· Facebook开发青少年短视频应用，产品经理也才19岁
· 微信城市服务上线“挂号平台”，欲打破医疗信息孤岛
· 《神秘的程序们》54：如何奸诈得从卖软件上赚钱
» 更多新闻...

最新知识库文章：
· 程序猿媳妇儿注意事项
· 可是姑娘，你为什么要编程呢？
· 知其所以然（以算法学习为例）
· 如何给变量取个简短且无歧义的名字
· 编程的智慧
» 更多知识库文章...

历史上的今天：
2011-10-09 httpModules与httpHandlers
2011-10-09 asp.net中web.config配置节点大全详解