MySQL表字段数量限制及行大小限制详情

目录

字段数量限制 Column Count Limits 行大小限制 Row Size Limits 8126B限制

Row Limits总结

字段数量限制 Column Count Limits

MySQL对于每个表的字段数量是严格控制在4096个,但是实际情况下的限制大小取决于以下因素 (也就是基本达不到4096就被限制了):

表的行最大的row size会限制字段数量,如果当前row size过大就不能加字段了
The storage requirements of individual columns constrain the number of columns that fit within a given maximum row size.

存储引擎本身也会对字段数量有限制,比如**Innodb**最大字段数量为1017 每张表都有一个后缀为**.frm**的表定义文件,表定义可能会影响字段数量大小

行大小限制 Row Size Limits

row size由以下因素影响:

MySQL本身对row size有硬性规定,不能超过65535bytes,即使存储引擎允许更大的size.而 BLOB和TEXT类型列只占行的9到12个字节,具体存储地方不在该行里;但是其他的比如varchar还 是算在行里的,这里要注意

Innodb存储引擎对于每行的大小一般限制为页大小的一半:页16KB,row size 8KB。另外对于不定长类型也有不同:If a row containing variable-length columns exceeds the InnoDB maximum row size, InnoDB selects variable-length columns for external off-page storage until the row fits within the InnoDB row size limit. The amount of data stored locally for variable-length columns that are stored off-page differs by row format. For more information, see InnoDB Row Formats.简单来说就是,对于像varchar这种不定长类型,如果这种类型长度超过了Innodb存储引擎规定的row size,那么Innodb会选择页外存储直到行大小符合Innodb存储引擎规定的row size。(但是即使这样也不能超过65535B,即65535B是包含不定长列中的内容的长度)

Different storage formats use different amounts of page header and trailer data, which affects the amount of storage available for rows. Row Size Limit Examples 65535B限制

看,由于**Latin1**编码1个字符是1个字节,总共不能超过65535个字符,超过就报错;而常用的 utf8mb4编码最多1个字符占用4个字节,所以当使用utf8mb4编码时,最多只能有 65535/4=16383个字符(实际测肯定会小点,因为还有字节去记录变长字段长度):

```
test> CREATE TABLE t (a VARCHAR(10000), b VARCHAR(6383)

DENGINE=InnoDB CHARACTER SET utf8mb4

[2022-07-21 15:09:51] [42000] [1118] Row size too large. The maximum row size for the used table type, not counting BLOBs, is 65535. This includes storage overhead, check the manual. You have to change some columns to TEXT or BLOBs test> CREATE TABLE t (a VARCHAR(10000), b VARCHAR(6382)

DENGINE=InnoDB CHARACTER SET utf8mb4

[2022-07-21 15:09:58] completed in 33 ms
```

根据提示,如果超过row size限制,可以使用TEXT or BLOBs类型,这个在row size中只占用9到12个字节。

8126B限制

65535B的限制主要针对不定长类型的限制,而定长类型的限制更为严格,像在Innodb存储引擎中,只能达到8KB多也就是页大小的一半(可以修改)。

innodb_strict_mode is enabled in the following example to ensure that InnoDB returns an error if the defined columns exceed the InnoDB row size limit. When innodb_strict_mode is disabled (the default), creating a table that uses REDUNDANT or COMPACT row format succeeds with a warning if the InnoDB row size limit is exceeded.

```
mysql> SET SESSION innodb_strict_mode=1;
mysql> CREATE TABLE t4 (
       c1 CHAR(255), c2 CHAR(255), c3 CHAR(255),
       c4 CHAR(255),c5 CHAR(255),c6 CHAR(255),
       c7 CHAR(255),c8 CHAR(255),c9 CHAR(255),
       c10 CHAR(255),c11 CHAR(255),c12 CHAR(255),
       c13 CHAR(255),c14 CHAR(255),c15 CHAR(255),
       c16 CHAR(255),c17 CHAR(255),c18 CHAR(255),
       c19 CHAR(255),c20 CHAR(255),c21 CHAR(255),
       c22 CHAR(255), c23 CHAR(255), c24 CHAR(255),
       c25 CHAR(255), c26 CHAR(255), c27 CHAR(255),
       c28 CHAR(255),c29 CHAR(255),c30 CHAR(255),
       c31 CHAR(255),c32 CHAR(255),c33 CHAR(255)
       ) ENGINE=InnoDB ROW FORMAT=COMPACT DEFAULT CHARSET latin1;
ERROR 1118 (42000): Row size too large (> 8126). Changing some columns to TEXT or
BLOB or using
ROW_FORMAT=DYNAMIC or ROW_FORMAT=COMPRESSED may help. In current row format, BLOB
prefix of 768
bytes is stored inline.
```

Row Limits总结

Row Limits 一般为 MySQL本身65535B限制和存储引擎8126B限制(默认Innodb)。

针对不定长类型,如varchar,一般首先收到限制的是MySQL本身65535B的限制。受不到存储引擎限制是因为,不定长类型如果长度超过8126B,会采用页外存储,也就是不定长类型的长度过长的话计入65535B而不计入8126B(大致可以这么理解)。

针对定长类型,首先会受到存储引擎8126B限制。

到此这篇关于MySQL表字段数量限制及行大小限制详情的文章就介绍到这了,更多相关MySQL行大小限制内容请搜索以前的文章或继续浏览下面的相关文章希望大家以后多多支持!