如何在MySQL中分配innodb_buffer_pool_size

innodb_buffer_pool_size是整个MySQL服务器最重要的变量。

1. 为什么需要innodb buffer pool?

在MySQL5.5之前，广泛使用的和默认的存储引擎是MyISAM。MyISAM使用操作系统缓存来缓存数据。InnoDB需要innodb buffer pool中处理缓存。所以非常需要有足够的InnoDB buffer pool空间。

2. MySQL InnoDB buffer pool 里包含什么?

- **数据缓存**

InnoDB数据页面

- **索引缓存**

索引数据

- **缓冲数据**

脏页（在内存中修改尚未刷新(写入)到磁盘的数据）

- **内部结构**

如自适应哈希索引，行锁等。

3. 如何设置innodb_buffer_pool_size?

innodb_buffer_pool_size默认大小为128M。最大值取决于CPU的架构。在32-bit平台上，最大值为2**32 -1,在64-bit平台上最大值为2**64-1。当缓冲池大小大于1G时，将innodb_buffer_pool_instances设置大于1的值可以提高服务器的可扩展性。

大的缓冲池可以减小多次磁盘I/O访问相同的表数据。在专用数据库服务器上，可以将缓冲池大小设置为服务器物理内存的80%。

3.1 配置缓冲池大小时，请注意以下潜在问题

- 物理内存争用可能导致操作系统频繁的paging
- InnoDB为缓冲区和control structures保留了额外的内存，因此总分配空间比指定的缓冲池大小大约大10%。
- 缓冲池的地址空间必须是连续的，这在带有在特定地址加载的DLL的Windows系统上可能是一个问题。
- 初始化缓冲池的时间大致与其大小成比例。在具有大缓冲池的实例上，初始化时间可能很长。要减少初始化时间，可以在服务器关闭时保存缓冲池状态，并在服务器启动时将其还原。
    - innodb_buffer_pool_dump_pct：指定每个缓冲池最近使用的页面读取和转储的百分比。 范围是1到100。默认值是25。例如，如果有4个缓冲池，每个缓冲池有100个page，并且innodb_buffer_pool_dump_pct设置为25，则dump每个缓冲池中最近使用的25个page。

    - innodb_buffer_pool_dump_at_shutdown：默认启用。指定在MySQL服务器关闭时是否记录在InnoDB缓冲池中缓存的页面，以便在下次重新启动时缩短预热过程。
    - innodb_buffer_pool_load_at_startup：默认启用。指定在MySQL服务器启动时，InnoDB缓冲池通过加载之前保存的相同页面自动预热。 通常与innodb_buffer_pool_dump_at_shutdown结合使用。

增大或减小缓冲池大小时，将以chunk的形式执行操作。chunk大小由innodb_buffer_pool_chunk_size配置选项定义，默认值为128 MB。

缓冲池大小必须始终等于或者是innodb_buffer_pool_chunk_size * innodb_buffer_pool_instances的倍数。

如果将缓冲池大小更改为不等于或等于innodb_buffer_pool_chunk_size * innodb_buffer_pool_instances的倍数的值，

则缓冲池大小将自动调整为等于或者是innodb_buffer_pool_chunk_size * innodb_buffer_pool_instances的倍数的值。

innodb_buffer_pool_size可以动态设置，允许在不重新启动服务器的情况下调整缓冲池的大小。 可以通过状态变量Innodb_buffer_pool_resize_status报告在线调整缓冲池大小操作的状态。

```
mysql> show status like 'Innodb_buffer_pool_resize%';
+-----------------------------------+-------+
| Variable_name                     | Value |
+-----------------------------------+-------+
| Innodb_buffer_pool_resize_status  |       |
+-----------------------------------+-------+
```

3.2 配置示例

在以下示例中，innodb_buffer_pool_size设置为3G，innodb_buffer_pool_instances设置为8。innodb_buffer_pool_chunk_size默认值为128M。

3G是有效的innodb_buffer_pool_size值，因为3G是innodb_buffer_pool_instances = 8 * innodb_buffer_pool_chunk_size = 128M的倍数

```
# mysqld --innodb_buffer_pool_size=3G --innodb_buffer_pool_instances=8 &

mysql> show variables like 'innodb_buffer_pool%';

+-------------------------------------+----------------+
| Variable_name                       | Value          |
+-------------------------------------+----------------+
| innodb_buffer_pool_chunk_size       | 134217728      |
| innodb_buffer_pool_dump_at_shutdown | ON             |
| innodb_buffer_pool_dump_now         | OFF            |
| innodb_buffer_pool_dump_pct         | 25             |
| innodb_buffer_pool_filename         | ib_buffer_pool |
| innodb_buffer_pool_instances        | 8              |
| innodb_buffer_pool_load_abort       | OFF            |
| innodb_buffer_pool_load_at_startup    | ON             |
| innodb_buffer_pool_load_now           | OFF            |
| innodb_buffer_pool_size               | 3221225472     |
+---------------------------------------+----------------+
10 rows in set (0.01 sec)
```

在以下示例中，innodb_buffer_pool_size设置为3G，innodb_buffer_pool_instances设置为16，innodb_buffer_pool_chunk_size为128M。
3G不是有效的innodb_buffer_pool_size值，因为3G不是innodb_buffer_pool_instances = 16 * innodb_buffer_pool_chunk_size = 128M的倍数，可以看出innodb_buffer_pool_size的值自动调整到4GB

```
# mysqld --innodb_buffer_pool_size=3G --innodb_buffer_pool_instances=16 &

mysql> show variables like '%innodb_buffer_pool%';
+---------------------------------------+----------------+
| Variable_name                         | Value          |
+---------------------------------------+----------------+
| innodb_buffer_pool_chunk_size         | 134217728      |
| innodb_buffer_pool_dump_at_shutdown   | ON             |
| innodb_buffer_pool_dump_now           | OFF            |
| innodb_buffer_pool_dump_pct           | 25             |
| innodb_buffer_pool_filename           | ib_buffer_pool |
| innodb_buffer_pool_instances          | 16             |
| innodb_buffer_pool_load_abort         | OFF            |
| innodb_buffer_pool_load_at_startup    | ON             |
| innodb_buffer_pool_load_now           | OFF            |
| innodb_buffer_pool_size               | 4294967296     |
+---------------------------------------+----------------+
10 rows in set (0.01 sec)
```

3.3 在线调整InnoDB缓冲池大小

```
mysql> SET GLOBAL innodb_buffer_pool_size = 3221225472
```

3.4 监控在线缓冲池调整进度

```
mysql> SHOW STATUS WHERE Variable_name='InnoDB_buffer_pool_resize_status';
+----------------------------------+----------------------------------------------------+
| Variable_name                    | Value                                              |
+----------------------------------+----------------------------------------------------+
| Innodb_buffer_pool_resize_status | Completed resizing buffer pool at 180824 15:05:03. |
+----------------------------------+----------------------------------------------------+
```

缓冲池大小调整进度也记录在服务器错误日志中。以下为增大时，记录的日志

```
2018-08-24T07:05:03.819049Z 2 [Note] InnoDB: Requested to resize buffer pool. (new size: 3221225472 bytes)
2018-08-24T07:05:03.819141Z 0 [Note] InnoDB: Resizing buffer pool from 2684354560 to 3221225472 (unit=134217728).
2018-08-24T07:05:03.819155Z 0 [Note] InnoDB: Disabling adaptive hash index.
2018-08-24T07:05:03.824902Z 0 [Note] InnoDB: disabled adaptive hash index.
2018-08-24T07:05:03.824933Z 0 [Note] InnoDB: Withdrawing blocks to be shrunken.
2018-08-24T07:05:03.824940Z 0 [Note] InnoDB: Latching whole of buffer pool.
2018-08-24T07:05:03.824959Z 0 [Note] InnoDB: buffer pool 0 : resizing with chunks 5 to 6.
2018-08-24T07:05:03.839564Z 0 [Note] InnoDB: buffer pool 0 : 1 chunks (8192 blocks) were added.
2018-08-24T07:05:03.839594Z 0 [Note] InnoDB: buffer pool 1 : resizing with chunks 5 to 6.
2018-08-24T07:05:03.848910Z 0 [Note] InnoDB: buffer pool 1 : 1 chunks (8192 blocks) were added.
2018-08-24T07:05:03.849046Z 0 [Note] InnoDB: buffer pool 2 : resizing with chunks 5 to 6.
2018-08-24T07:05:03.856711Z 0 [Note] InnoDB: buffer pool 2 : 1 chunks (8192 blocks) were added.
2018-08-24T07:05:03.856741Z 0 [Note] InnoDB: buffer pool 3 : resizing with chunks 5 to 6.
2018-08-24T07:05:03.864867Z 0 [Note] InnoDB: buffer pool 3 : 1 chunks (8192 blocks) were added.
2018-08-24T07:05:03.864902Z 0 [Note] InnoDB: Completed to resize buffer pool from 2684354560 to 3221225472.
2018-08-24T07:05:03.864915Z 0 [Note] InnoDB: Re-enabled adaptive hash index.
2018-08-24T07:05:03.864935Z 0 [Note] InnoDB: Completed resizing buffer pool at 180824 15:05:03.
```

以下为减小时，记录的日志

```
2018-08-24T07:10:20.666816Z 2 [Note] InnoDB: Requested to resize buffer pool. (new size: 2684354560 bytes)
2018-08-24T07:10:20.666880Z 0 [Note] InnoDB: Resizing buffer pool from 3221225472 to 2684354560 (unit=134217728).
2018-08-24T07:10:20.666889Z 0 [Note] InnoDB: Disabling adaptive hash index.
2018-08-24T07:10:20.673416Z 0 [Note] InnoDB: disabled adaptive hash index.
2018-08-24T07:10:20.673508Z 0 [Note] InnoDB: Withdrawing blocks to be shrunken.
2018-08-24T07:10:20.673519Z 0 [Note] InnoDB: buffer pool 0 : start to withdraw the last 8192 blocks.
2018-08-24T07:10:20.678441Z 0 [Note] InnoDB: buffer pool 0 : withdrawing blocks. (8192/8192)
2018-08-24T07:10:20.678521Z 0 [Note] InnoDB: buffer pool 0 : withdrew 8192 blocks from free list. Tried to relocate 0 pages (8192/8192).
2018-08-24T07:10:20.678919Z 0 [Note] InnoDB: buffer pool 0 : withdrawn target 8192 blocks.
2018-08-24T07:10:20.678977Z 0 [Note] InnoDB: buffer pool 1 : start to withdraw
```

```
the last 8192 blocks.
2018-08-24T07:10:20.681644Z 0 [Note] InnoDB: buffer pool 1 : withdrawing blocks. (8192/8192)
2018-08-24T07:10:20.682168Z 0 [Note] InnoDB: buffer pool 1 : withdrew 8192 blocks from free list. Tried to relocate 0 pages (8192/8192).
2018-08-24T07:10:20.682235Z 0 [Note] InnoDB: buffer pool 1 : withdrawn target 8192 blocks.
2018-08-24T07:10:20.682254Z 0 [Note] InnoDB: buffer pool 2 : start to withdraw the last 8192 blocks.
2018-08-24T07:10:20.686560Z 0 [Note] InnoDB: buffer pool 2 : withdrawing blocks. (8192/8192)
2018-08-24T07:10:20.686917Z 0 [Note] InnoDB: buffer pool 2 : withdrew 8192 blocks from free list. Tried to relocate 0 pages (8192/8192).
2018-08-24T07:10:20.687002Z 0 [Note] InnoDB: buffer pool 2 : withdrawn target 8192 blocks.
2018-08-24T07:10:20.687010Z 0 [Note] InnoDB: buffer pool 3 : start to withdraw the last 8192 blocks.
2018-08-24T07:10:20.690038Z 0 [Note] InnoDB: buffer pool 3 : withdrawing blocks. (8192/8192)
2018-08-24T07:10:20.690373Z 0 [Note] InnoDB: buffer pool 3 : withdrew 8192 blocks from free list. Tried to relocate 0 pages (8192/8192).
2018-08-24T07:10:20.690433Z 0 [Note] InnoDB: buffer pool 3 : withdrawn target 8192 blocks.
2018-08-24T07:10:20.690479Z 0 [Note] InnoDB: Latching whole of buffer pool.
2018-08-24T07:10:20.690498Z 0 [Note] InnoDB: buffer pool 0 : resizing with chunks 6 to 5.
2018-08-24T07:10:20.693293Z 0 [Note] InnoDB: buffer pool 0 : 1 chunks (8192 blocks) were freed.
2018-08-24T07:10:20.693357Z 0 [Note] InnoDB: buffer pool 1 : resizing with chunks 6 to 5.
2018-08-24T07:10:20.695947Z 0 [Note] InnoDB: buffer pool 1 : 1 chunks (8192 blocks) were freed.
2018-08-24T07:10:20.696011Z 0 [Note] InnoDB: buffer pool 2 : resizing with chunks 6 to 5.
2018-08-24T07:10:20.698977Z 0 [Note] InnoDB: buffer pool 2 : 1 chunks (8192 blocks) were freed.
2018-08-24T07:10:20.699288Z 0 [Note] InnoDB: buffer pool 3 : resizing with chunks 6 to 5.
2018-08-24T07:10:20.702088Z 0 [Note] InnoDB: buffer pool 3 : 1 chunks (8192 blocks) were freed.
2018-08-24T07:10:20.702398Z 0 [Note] InnoDB: Completed to resize buffer pool from 3221225472 to 2684354560.
2018-08-24T07:10:20.702413Z 0 [Note] InnoDB: Re-enabled adaptive hash index.
2018-08-24T07:10:20.703896Z 0 [Note] InnoDB: Completed resizing buffer pool at 180824 15:10:20.
```

4. 配置的innodb_buffer_pool_size是否合适?

当前配置的innodb_buffer_pool_size是否合适，可以通过分析InnoDB缓冲池的性能来验证。
可以使用以下公式计算InnoDB缓冲池性能：

```
Performance = innodb_buffer_pool_reads / innodb_buffer_pool_read_requests * 100
```

innodb_buffer_pool_reads：表示InnoDB缓冲池无法满足的请求数。需要从磁盘中读取。

innodb_buffer_pool_read_requests：表示从内存中读取逻辑的请求数。

例如，在我的服务器上，检查当前InnoDB缓冲池的性能：

```
root@localhost [(none)] 15:35:31>show status like 'innodb_buffer_pool_read%';
+---------------------------------------+-------------+
| Variable_name                         | Value       |
+---------------------------------------+-------------+
| Innodb_buffer_pool_read_ahead_rnd     | 0           |
| Innodb_buffer_pool_read_ahead         | 0           |
| Innodb_buffer_pool_read_ahead_evicted | 0           |
| Innodb_buffer_pool_read_requests      | 4029033624  |
| Innodb_buffer_pool_reads              | 91661       |
+---------------------------------------+-------------+
5 rows in set (0.00 sec)


Performance = 91661 / 4029033624 * 100 = 0.0022750120389663
```

意味着InnoDB可以满足缓冲池本身的大部分请求。从磁盘完成读取的百分比非常小。因此无需增加innodb_buffer_pool_size值。

4.1 什么时候减小innodb_buffer_pool_size?

在专用MySQL服务器上，多余的innodb_buffer内存不会有问题，但是当使用共享服务器时，可能会有性能影响。因为空闲内存对其他程序和操作系统很有用。

可以使用SHOW ENGINE INNODB STATUS\G命令检查内存状态：

```
mysql> show engine innodb status\G

...
Total large memory allocated 26386366464
Dictionary memory allocated 23826297
Buffer pool size   1572672
Free buffers       8192
Database pages     1553364
Old database pages 573246
Modified db pages  36
Pending reads      0
Pending writes: LRU 0, flush list 0, single page 0
Pages made young 881819, not young 18198964
0.02 youngs/s, 0.05 non-youngs/s
Pages read 681064, created 2749237, written 3988300
0.02 reads/s, 0.12 creates/s, 11.50 writes/s
Buffer pool hit rate 1000 / 1000, young-making rate 0 / 1000 not 0 / 1000
```

```
Pages read ahead 0.00/s, evicted without access 0.00/s, Random read ahead 0.00/s
LRU len: 1553364, unzip_LRU len: 0
I/O sum[5152]:cur[0], unzip sum[0]:cur[0]
...
```

Free buffers :表示有多少空闲buffer。如果 此值长时间都较高，则可以考虑减小InnoDB缓冲池大小。

InnoDB buffer pool 命中率：

```
InnoDB buffer pool 命中率 = innodb_buffer_pool_read_requests /
(innodb_buffer_pool_read_requests + innodb_buffer_pool_reads ) * 100
```

此值低于99%，则可以考虑增加innodb_buffer_pool_size。

5. InnoDB缓冲池状态变量有哪些?

可以运行以下命令进行查看：

```
root@localhost [(none)] 16:00:31>show global status like
'%innodb_buffer_pool_pages%';
+----------------------------------+--------+
| Variable_name                    | Value  |
+----------------------------------+--------+
| Innodb_buffer_pool_pages_data    | 457    |
| Innodb_buffer_pool_pages_dirty   | 0      |
| Innodb_buffer_pool_pages_flushed | 36     |
| Innodb_buffer_pool_pages_free    | 163363 |
| Innodb_buffer_pool_pages_misc    | 0      |
| Innodb_buffer_pool_pages_total   | 163820 |
+----------------------------------+--------+
6 rows in set (0.00 sec)
```

说明：

- **Innodb_buffer_pool_pages_data**

InnoDB缓冲池中包含数据的页数。 该数字包括脏页面和干净页面。 使用压缩表时，报告的Innodb_buffer_pool_pages_data值可能大于Innodb_buffer_pool_pages_total（Bug＃59550）。

- **Innodb_buffer_pool_pages_dirty**

显示在内存中修改但尚未写入数据文件的InnoDB缓冲池数据页的数量（脏页刷新）。

- **Innodb_buffer_pool_pages_flushed**

表示从InnoDB缓冲池中刷新脏页的请求数。

- **Innodb_buffer_pool_pages_free**

显示InnoDB缓冲池中的空闲页面

- **Innodb_buffer_pool_pages_misc**

InnoDB缓冲池中的页面数量很多，因为它们已被分配用于管理开销，例如行锁或自适应哈希索引。此值也可以计算为Innodb_buffer_pool_pages_total - Innodb_buffer_pool_pages_free - Innodb_buffer_pool_pages_data。

- **Innodb_buffer_pool_pages_total**

InnoDB缓冲池的总大小，以page为单位。

- **innodb_buffer_pool_reads**

表示InnoDB缓冲池无法满足的请求数。需要从磁盘中读取。

- **innodb_buffer_pool_read_requests**

它表示从内存中逻辑读取的请求数。

- **innodb_buffer_pool_wait_free**

通常，对InnoDB缓冲池的写入发生在后台。 当InnoDB需要读取或创建页面并且没有可用的干净页面时，InnoDB首先刷新一些脏页并等待该操作完成。 此计数器计算这些等待的实例。 如果已正确设置innodb_buffer_pool_size，则此值应该很小。如果大于0，则表示InnoDb缓冲池太小。

- **innodb_buffer_pool_write_request**

表示对缓冲池执行的写入次数。

6. InnoDB缓冲池当前使用了多少实际GB内存?

通过将缓冲池中可用的数据与InnoDB页面（InnoDB缓冲池单位）大小相乘，我们可以发现InnoDB缓冲池此时正在使用的实际内存。

```
set @ibpdata = (select variable_value from information_schema.global_status
where variable_name = 'innodb_buffer_pool_pages_data');

ERROR 3167 (HY000): The 'INFORMATION_SCHEMA.GLOBAL_STATUS' feature is disabled;
see the documentation for 'show_compatibility_56'

#从MySQL 5.7.6开始，GLOBAL_STATUS表中提供的信息从Performance Schema获取

mysql> set @ibpdata = (select variable_value from
performance_schema.global_status where variable_name =
'innodb_buffer_pool_pages_data');

mysql> select @ibpdata;
+-----------+
| @ibpdata |
+-----------+
| 568       |
+-----------+
1 row in set (0.00 sec)


mysql> set @idbpgsize = (select variable_value from
performance_schema.global_status where variable_name = 'innodb_page_size');

mysql> select @idbpgsize;
+-------------+
| @idbpgsize |
+-------------+
| 16384       |
+-------------+
1 row in set (0.00 sec)
```

```
mysql> set @ibpsize = @ibpdata * @idbpgsize / (1024*1024*1024);
Query OK, 0 rows affected (0.00 sec)

mysql> select @ibpsize;
+-----------------+
| @ibpsize        |
+-----------------+
| 0.0086669921875 |
+-----------------+
1 row in set (0.00 sec)
```