# InnoDB源码目录剖析

InnoDB除了与MySQL层交互的部分为C++实现外，其余的基本都是以C语言实现。还有很少量的汇编语言。源码结构非常清晰。每个模块一个单独的文件夹，文件夹下为.c或者.cc文件。而.h文件统一放在include目录下。文件的命令规则为：模块名0子模块名。例如，关于B+树索引模块的文件命名为btr/btr0xxx.cc。文件名中，模块名部分就是文件夹的名字。有的源代码是将.h和.c放在一起，InnoDB这一点与其他的不同。.h文件在include目录下的文件名与c源程序文件夹中的.c或.cc的文件名对应。include中除了.h文件外还有.ic文件。这类文件为每个模块定义了内联函数。打开任意一个.ic文件，你会发现其中的函数都被UNIV\_INLINE修饰。UNIV\_INLINE在include/univ.i中定义：

#define UNIV\_INLINE static inline

InnoDB为了达到工业级的稳定性要求，做了很多努力，性能上也做了很多优化，这也伴随着大量的代码，如果不先从宏观上把握其结构而一头扎进源码中，就会有盲人摸象的感觉。因此先把InnoDB的各个模块画出来：

Macintosh HD:Users:xufubo:Desktop:InnoDB.pdf

最底层的是最基本的模块，文件管理封装了InnoDB对文件的各类操作。并发管理封装了各类mutex和latch，通用工具实现了一些基本的数据结构和算法，例如：链表、哈希表等。

途中至底向上看，第二和三层是InnoDB的核心，也是一个存储引擎都包含的各种模块。理解了它们基本就理解了InnoDB的运行机理和一个存储引擎的实现方法。

图中最上边的两层为接口层，通过这些接口与存储引擎内部进行通信。InnoDB可以不依赖于MySQL数据库，而作为一个嵌入式的数据库存在，因此还提供了嵌入式API的接口（这一部分不是我们学习的范畴）。

了解了InnoDB的整体结构，接下来看看InnoDB的源码目录。

对于MySQL 5.7.18，InnoDB的源码目录结构如下：

1、api：InnoDB可以作为SQL语义的存储引擎，同时也可以用于嵌入式，而不是通过SQL接口访问，该目录中就是这些API的实现。我们不做学习。

2、btr：B+树以及B+树索引等的实现。重点学习。

3、buf：任何一个存储引擎对内存的高效管理都是必不可少的，因为通过缓冲磁盘的数据减少磁盘IO，从而提高性能。该目录就是缓冲区的管理（缓冲区很大一部分都是用来缓冲磁盘的数据的）。重点学习。

4、data：影响行的次要实用程序集合。

5、dict：数据字典相关的程序。重点学习。

6、eval（EVALUATING）：SQL语句执行代价的评估。执行SQL存储过程是InnoDB的一个功能，但MySQL以其自己的方式处理存储过程，因此该目录中的程序不是非常重要。

7、fil：file的简写，数据库文件的读取和写入的实现，与底层文件IO例程协作完成文件的读写。底层的文件IO例程在os文件夹中的fil0fil.c中。重点学习。

8、fsp：file space的简写，作用与fil类似，也是用于文件的读写。重点学习。

9、fts：full text search的简写，全文检索相关的实现。

10、fut：file utility的简写，一些文件操作的使用例程。主要的文件操作和管理还是在fil、fsp、fts中。重点学习。

11、gis：R-tree相关的实现。

12、ha：哈希表的实现。重点学习。

13、handler：与MySQL层通信的接口。重点学习。

14、ibuf：之前的插入缓冲区，现在的change buffer的实现。重点学习。

15、include：包含了InnoDB实现需要的各种头文件。重点学习。

16、lock：事务相关的锁的操作实现。重点学习。

17、log：InnoDB所有日志相关的实现。例如重做日志。重点学习。

18、mach：mach0data.c中有两个小程序用于读取压缩的ulints（无符号长整数）

19、mem：通用内存池，即不在缓冲池（buf目录中的实现）和日志池（log目录中的实现）中的内存空间。重点学习。

20、mtr：*MINI-TRANSACTION*的简写。被大多数其他程序组调用的小型事务例程。可以理解成低级实用程序集。

21、os：这是一组其他模块在需要使用操作系统资源时可以调用的实用程序。其中的实现将不同的操作系统的函数进行了封装，为其他模块提供了同一的语义（因为，例如Linux和windows中的打开文件的函数就不一样，该目录中的代码会屏蔽这些不同）。重点学习。

22、page：InnoDB中的操作以page（InnoDB的页，不要和操作系统的页混淆）为单位，其中为页相关的实现。学习和研究的重点。重点学习。

23、pars：*PARSING*的简写，该目录实现的功能是：输入一个包含SQL语句的字符串，并输出一个内存中的解析树。EVALUATING（目录eval）程序将使用树进行SQL执行的代价评估。

通常的做法是使用Bison和Flex工具。 pars0grm.c是由Bison解析器从原始文件pars0grm.y生成的，而且lexyy.c是Flex生成的。

由于InnoDB本身是一个DBMS，因为它支持嵌入式的API。所以解析SQL语句很自然。但是在MySQL / InnoDB组合中，MySQL处理大部分的解析。因此这些文件并不重要。

24、que：程序que0que.c表面上是关于包含commit /rollback语句的存储过程的执行。我认为这对于一般的MySQL用户来说并不重要。

25、read：事务实现时需要使用的一组例程。

26、rem：record manager，记录管理。学习的重点。

27、row：InnoDB行的相关程序的实现，例如：InnoDB行与MySQL行的转换，与行有关的维护活动，例如插入、更新、删除等等。学习的重点

28、srv：这是服务器读取初始配置文件，依据配置拆分线程并进行访问的实现。也是学习和研究的重点

29、sync：现代操作系统和C语言函数库都会实现互斥原语，但是有时候不高效，自己定制实现反而高效，该目录中就是同步与互斥相关的实现，以及死锁等同步问题的处理。学习的重点。

30、trx：InnoDB事务的实现。学习和研究的重点

31、usr：一个用户可以有多个会话（会话是连接和断开连接之间发生的所有事情）。这是 InnoDB用于跟踪会话ID和服务器/客户端消息传递的位置。这通常是MySQL的工作。因此，对于学习InnoDB也不是很重要。

32、ut：一些试用程序，不重要。

作者：许富博

版权所有，文章以学习和交流为主，切勿用于商业用途。

限于本人水平有限，欢迎大家随时指正，联系方式：

[xufubobo@gmail.com](mailto:xufubobo@gmail.com)

[xufubobo@163.com](mailto:xufubobo@163.com)

[1332841493@qq.com](mailto:1332841493@qq.com)