CBASE 分布式数据库参考指南

1 日志参考

CBASE在运行过程中会自动生成日志。维护工程师通过查看和分析日志，可以了解 CBASE的启动和运行状态。

1.1 日志概述

主要介绍日志清单、日志级别、查看日志的方法和收集日志的方法等。

1.1.1 日志清单

列出了 CBASE运行日志和操作日志的清单。CBASE日志清单如表1-1 所示。

表 1-1 日志清单

服务类型	日志名称	日志路径	说明
RootServer	rootserver.log	RootServer 所在CBase服务器的“~/oceanbase/log”目录下。	记录RootServer 启动过程以及启动后的运行情况。
1、2、3、...	RootServer 所在CBase服务器的“~/oceanbase/data/rs_commitlog”目录下。	RootServer 的commit 日志，记录RootServer的Tablet 信息和schema 信息。	
UpdateServer	updateserver.log	UpdateServer 所在CBase服务器的“~/oceanbase/log”目录下。	UpdateServer 所在CBase服务器的“~/oceanbase/log”目录下。
1、2、3、...	UpdateServer 所在CBase服务器的“~/oceanbase/data/ups_commitlog”目录下。	UpdateServer 所在CBase服务器的“~/oceanbase/data/ups_commitlog”目录下。	
MergeServer	mergeserver.log	MergeServer 所在CBase服务器的“~/oceanbase/log”目录下。	记录CBase启动过程以及启动后的运行情况。
ChunkServer	chunkserver.log	所在 CBase服务器的“~/oceanbase/log”目录下。	记录ChunkServer 启动过程以及启动后的运行情况。
-	rs_admin.log	执行 rs_admin 命令的CBase服务器的“~/oceanbase”目录下。	记录bin/rs_admin 的执行日志。

1.1.2 日志级别

介绍 CBASE日志的级别及每类级别的含义。当启动或运行异常时，可维护工程师可以通过日志级别和内容分析和定位异常原因。CBASE的运行日志中的级别及含义如表 1-2 所示。日志级别从高到底排列。

表 1-2 日志级别及含义

日志级别	含义
ERROR	严重错误，用于记录系统的故障信息，且必须进行故障排除，否则系统不可用。
WARN	警告，用于记录可能会出现的潜在错误。
INFO	提示，用于记录系统运行的当前状态，该信息为正常信息。
DEBUG	调试信息，用于调试时更详细的了解系统运行状态，包括当前调用的函数名、参数、变量、函数调用返回值等。

1.1.3 查看日志

介绍了查看 CBASE日志的方法。本文以查看 RootServer 为例，介绍查看 CBASE日志方法。假设 RootServer 所在的 CBASE服务器 IP 为“10.10.10.2”，查看日志的步骤如下：

1. 登录 CBASE服务器（10.10.10.2）。
2. 执行以下命令，进入日志文件所在的目录。

```
cd ~/oceanbase/log
```

3. 执行以下命令，查看已写入的日志记录。

```
more rootserver.log
```

或者执行以下命令，查看正在写入的日志记录。

tail rootserver.log

1.1.4 收集日志

介绍了收集 CBASE日志的方法。建议日志收集命名格式为：“日志名称_收集日期.tar”。本文以收集 RootServer 和 UpdateServer 日志为例，介绍收集 CBASE日志的方法。假设 RootServer 和 UpdateServer 合设的 CBASE服务器 IP 为“10.10.10.2”，收集 CBASE日志的操作步骤如下：

1. 登录 CBASE服务器（10.10.10.2）。
2. 执行以下命令，进入日志文件所在的目录。

cd ~/oceanbase/log

3. 执行以下命令，收集日志。

tar cvf rs_and_ups_130809.tar rootserver.log updateserver.log

说明：如果需要收集当前目录下的所有日志，可执行 tar cvf 日志名称 .tar * 命令 。

1.2 启动和运行日志

记录 CBASE各 Server 启动过程以及启动后的运行情况。这是我们在日常工作中主要查看的日志。

* 日志说明

CBASE启动和运行日志的名称、路径以及说明如表 表 1-3 所示。

表 1-3 日志说明

服务类型	日志名称	日志路径	说明
RootServer	rootserver.log	RootServer 所在CBase服务器的“~/oceanbase/log”目录下。	记录 RootServer启动过程以及启动后的运行情况。
UpdateServer	updateserver.log	UpdateServer 所在CBase服务器的“~/oceanbase/log”目录下。	记录UpdateServer 启动过程以及启动后的运行情况。
MergeServer	mergeserver.log	MergeServer 所在CBase服务器的“~/oceanbase/log”目录下。	记录MergeServer 启动过程以及启动后的运行情况。
ChunkServer	chunkserver.log	ChunkServer 所在CBase服务器的“~/oceanbase/log”目录下。	记录ChunkServer 启动过程以及启动后的运行情况。

* 日志格式

日志记录主要由五部分组成：记录时间、日志级别、文件名:行号、线程 ID 和日志内容。

* 日志样例

[2013-08-09 10:02:18.339314] INFO ob_root_worker.cpp:3159 [139702581581568] start to boot strap

解释：OceanBase 集群进行“boot srtap”初始化。

1.3 commit 日志

记录 RootServer 的 Tablet 信息和 schema 信息和 UpdateServer 的 mutator 信息。UpdateServer 每次进行更新操作时，均会产生一条 commit log，并追加到已有的 commit log 序列之后。UpdateServer 通过同步 commit log 实现主备复制。

* 日志说明

CBASE 的 commit 日志的名称、路径和说明如表 表 1-4 所示。

表 1-4 日志说明

服务类型	日志名称	日志路径	说明
RootServer	1、2、3、…	RootServer 所在CBase 服务器的“~/oceanbase/data/rs_commitlog”目录下。	RootServer 的commit 日志，记录RootServer 的Tablet 信息和schema 信息。
UpdateServer	1、2、3、…	UpdateServer 所在CBase 服务器的“~/oceanbase/data/ups_commitlog”目录下。	UpdateServer 的commit 日志，记录UpdateServer 的mutator 信息。

1.4 rs_admin 日志

记录在初始化CBASE过程中运行 rs_admin 的日志。

* 日志说明

CBASE 的 rs_admin 日志的名称、路径和说明如表 表 1-5 所示。

表1-5 日志说明

日志名称	日志路径	说明
rs_admin.log	执行 rs_admin 命令的CBase 服务器的“~/oceanbase”目录下。	记录 bin/rs_admin 的执行日志。

* 日志格式

日志记录主要由五部分组成：记录时间、日志级别、文件名:行号、线程 ID 和日志内容。

* 日志样例

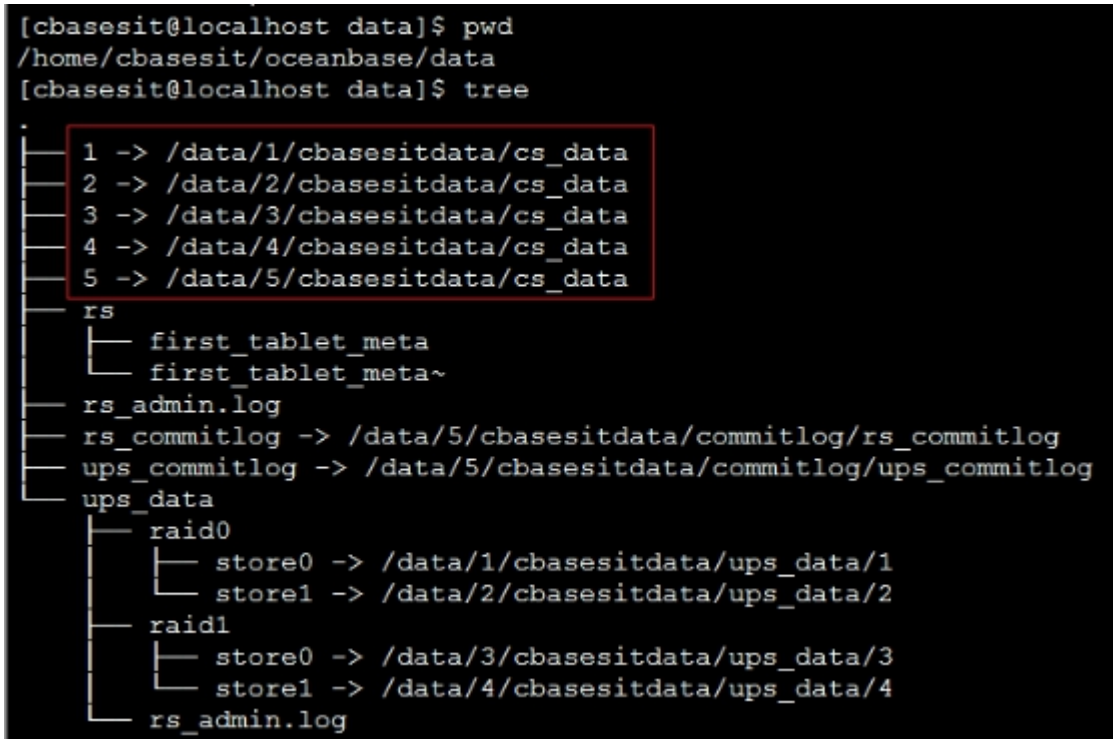
[2013-08-09 10:02:18.338467] INFO ob_base_client.cpp:70 [140524464404096] start io thread

解释：启动 io 线程。

2 数据目录与文件

CBASE的主数据目录路径在：~/oceanbase/data/

该目录的树形结构如下图所示：



CBASE数据库的数据分为基准数据和增量数据，基准数据由ChunkServer管理，增量数据由UpdateServer管理；另外还有一种比较重要的文件是commitlog文件，该文件相当于IBM/Oracle的redo log，在数据库重启或故障恢复时十分重要。

2.1 基准数据与ChunkServer

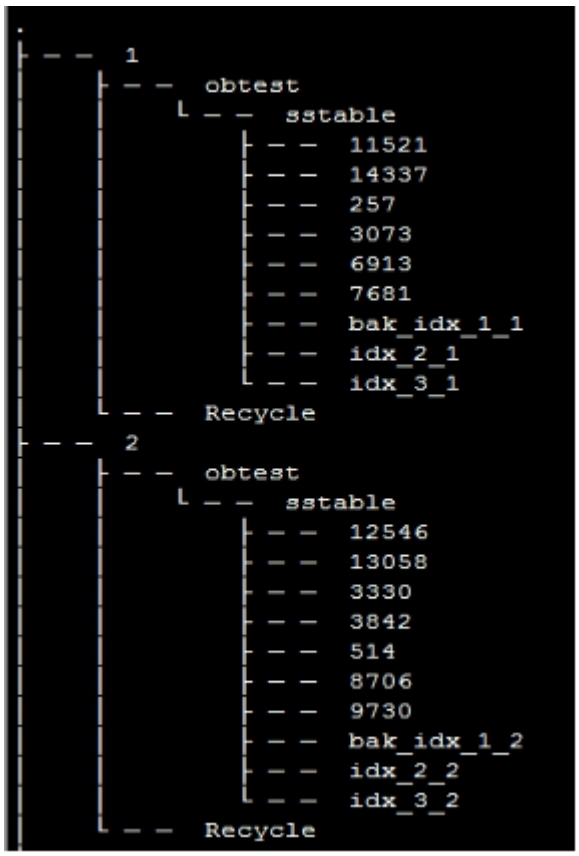
主数据目录下以数字命名的目录是ChunkServer存储基准数据的目录，生产环境上CS每个数据目录都会软连接到一个磁盘挂载目录，测试环境可根据机器磁盘资源软连接到适当的目录。目录结构如下所示：

CS每个数据目录下有两个目录：一个以app name命名，一个是Recycle目录，示例给出的app name是obtest。

CS会管理多个版本的基准数据，内存中维护两个版本。

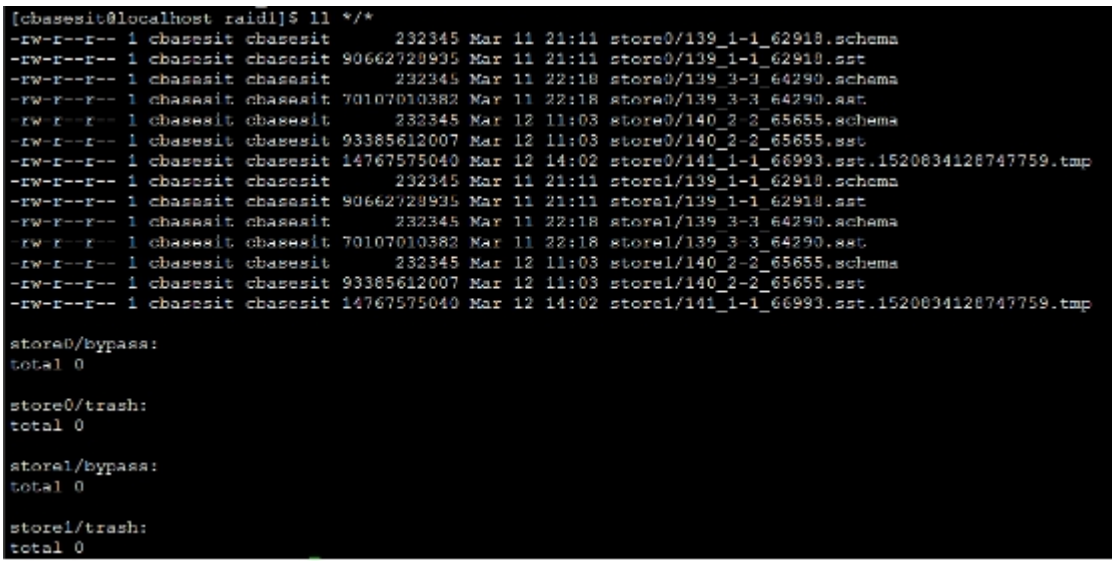
./app_name/sstable/下是CS管理维护的sstable文件及sstable的idx文件，sstable文件以数字命名，idx的文件命名格式为idx_version_disk。其中version是基准数据版本号，disk是数据目录编号。

因为CS的数据是多版本的，不可能把所有版本的数据一直保留，CS会把无用的旧版本数据移到Recycle目录，Recycle目录需要定期删除，否则会浪费磁盘空间。



2.2 增量数据与UpdateServer

所有的DML操作产生的数据会先写入到UpdateServer的内存中，再经过合并操作合并到CS的基准数据中。主数据目录下的ups_data目录是UpdateServer的数据目录，UPS每次内存冻结操作后会将内存中的数据dump到该目录下，该目录是一个模拟raid目录，每个raid下有store0和store1两个目录，UPS的每个冻结版本会在某个raid的store0和store1目录下各存储一份，通常storeN都会单独软连接到一个磁盘挂载目录。以raid1目录为例，目录结构如下所示：



每个冻结版本会在./store0/和./store1/下各存储一份，数据格式为：

majorVersion_minorVersion-minorVersion_replayPoint.sst[schema]

majorVersion是大版本号，minorVersion是小版本号，replayPoint是日志回放点，该回放点是UPS某个commitlog的文件名。

.sst是某个版本的内存dump到磁盘的文件，.schema是该版本对应的schema文件。上图中的*.tmp文件表示该版本的正在dump到磁盘，但尚未结束。

2.3 commitlog与UpdateServer

commitlog文件存储路径是：主数据目录/ups_commitlog/

与IBM的redo log类似，CBASE的所有DML/DDL操作都会写入commitlog，DML/DDL同步到备UPS，UPS重启，数据库备份恢复都会用到commitlog，因此commitlog非常重要。

ups_commitlog/目录下另有两个重要的文件，分别是log_replay_point，和commit_point文件。

log_replay_point标识该UPS下次重启的日志回放点，因此内容是某个commitlog的文件名，与UPS主数据目录ups_data下**majorVersion_minorVersion-minorVersion_replayPoint.sst[schema]**

replayPoint相对应。

log_replay_point更新时机：最新sst dump成功，则sst中replayPoint会被更新到log_replay_point中。

commit_point标识该UPS当前最新日志提交点，因此内容是数据库内部日志提交号。只有当CBASE的UPS节点需要异常恢复时，才需要关注日志提交号。

3 系统结果码

CBASE 结果码可以分为用户结果码和系统结果码。用户结果码主要用于定位和解决 CBASE Java 客户端用户或者 MySQL 命令行客户端返回的错误；系统结果码主要用于定位和解决 CBASE 的系统内部错误。

注意：在使用 CBASE 时，如果返回系统结果码或者您无法解决用户结果码的问题时，请联系 “ CBASE团队 ” 。

CBASE 用户结果码如表3-1 所示。

表 3-1 用户结果码

结果码	含义	参数
-12	请求超时。	const int OB_RESPONSE_TIME_OUT
-13	分配内存空间失败。	const int OB_ALLOCATE_MEMORY_FAILED
-14	UpdateServer 内存溢出。	const int OB_MEM_OVERFLOW
-38	读写集群不是主集群。	const int OB_NOT_MASTER
-42	用户不存在。	const int OB_USER_NOT_EXIST
-43	连接 CBase 的密码错误。	const int OB_PASSWORD_WRONG
-46	没有操作权限。	const int OB_NO_PERMISSION
-49	UpdateServer 处理超时。	const int OB_PROCESS_TIMEOUT
-56	建立 TCP 连接失败。	const int OB_CONN_ERROR
-64	MergeServer 将 RPC 请求拆分成多个 SESSION 从 ChunkServer 上读取数据时，SESSION 被杀死。	const int OB_SESSION_KILLED
-119	冻结时事务被回滚。	const int OB_TRANS_ROLLBACKED
-4007	主集群在每日合并过程中不允许进行 DDL 操作。备集群不允许进行 DD 操作。	const int OB_OP_NOT_ALLOW
-5001	SQL 语法错误	const int OB_ERR_PARSE_SQL
-5009	列不存在。	const int OB_ERR_COLUMN_UNKNOWN
-5019	表格不存在。	const int OB_ERR_TABLE_UNKNOWN
-5035	用户不存在。	const int OB_ERR_USER_NOT_EXIST
-5036	没有操作权限。	const int OB_ERR_NO_PRIVILEGE
-5038	连接 CBase 的密码错误。	const int OB_ERR_WRONG_PASSWORD
-5040	更新 RowKey 列错误。	const int OB_ERR_UPDATE_ROWKEY_COLUMN
-5041	更新 JOIN 列错误。	const int OB_ERR_UPDATE_JOIN_COLUMN
-5048	执行 REPLACE 操作的行锁冲突。	const int OB_ERR_EXCLUSIVE_LOCK_CONFLICT

结果码	含义	参数
-5049	执行 INSERT/UPDATE/DELETE 操作的行锁冲突。	const int OB_ERR_SHARED_LOCK_CONFLICT
-5060	事务已开始，开启新事务失败。	const int OB_ERR_TRANS_ALREADY_STARTED
-5062	不在事务中。	const int OB_ERR_NOT_IN_TRANS

CBASE 系统结果码说明如表 3-2 所示。

表 3-2 系统结果码

结果码	含义	参数
-1	Object 类型错误。	const int OB_OBJ_TYPE_ERROR
-2	传入的参数错误。	const int OB_INVALID_ARGUMENT
-3	数组越界。	const int OB_ARRAY_OUT_OF_RANGE
-4	服务器监听错误。	const int OB_SERVER_LISTEN_ERROR
-5	已经初始化。	const int OB_INIT_TWICE
-6	没有初始化。	const int OB_NOT_INIT
-7	不支持的功能。	const int OB_NOT_SUPPORTED
-8	迭代到最后一行。	const int OB_ITER_END
-9	IO 错误。	const int OB_IO_ERROR
-10	版本错误。	const int OB_ERROR_FUNC_VERSION
-11	包未发送。	const int OB_PACKET_NOT_SENT
-15	系统错误。	const int OB_ERR_SYS
-16	未知错误。	const int OB_ERR_UNEXPECTED
-17	项已存在。	const int OB_ENTRY_EXIST
-18	项不存在。	const int OB_ENTRY_NOT_EXIST
-19	大小越界。	const int OB_SIZE_OVERFLOW
-20	引用计数不为零。	const int OB_REF_NUM_NOT_ZERO
-21	值发生冲突。	const int OB_CONFLICT_VALUE
-22	项未设置。	const int OB_ITEM_NOT_SETTED
-23	需要重试。	const int OB_EAGAIN
-24	缓冲不足。	const int OB_BUF_NOT_ENOUGH
-25	同步 Slave 过程失败。	const int OB_PARTIAL_FAILED
-26	读取内容为空。	const int OB_READ_NOTHING
-27	文件不存在。	const int OB_FILE_NOT_EXIST
-28	日志不连续。	const int OB_DISCONTINUOUS_LOG
-29	Schema 错误。	const int OB_SCHEMA_ERROR
-30	不提供该数据服务。	const int OB_DATA_NOT_SERVE

结果码	含义	参数
-31	未知的 Object。	const int OB_UNKNOWN_OBJ
-32	没有监控数据。	const int OB_NO_MONITOR_DATA
-33	序列化失败。	const int OB_SERIALIZE_ERROR
-34	反序列化失败。	const int OB_DESERIALIZE_ERROR
-35	AIO 超时。	const int OB_AIO_TIMEOUT
-36	需要重试。	const int OB_NEED_RETRY
-37	SSTable 过多。	const int OB_TOO_MANY_SSTABLE
-39	Token 失效。	const int OB_TOKEN_EXPIRED
-40	加密失败。	const int OB_ENCRYPT_FAILED
-41	解密失败。	const int OB_DECRYPT_FAILED
-44	秘钥错误。	const int OB_SKEY_VERSION_WRONG
-45	不是 Token 类型。	const int OB_NOT_A_TOKEN
-47	条件检查失败。	const int OB_COND_CHECK_FAIL
-48	未注册。	const int OB_NOT_REGISTERED
-50	不是这个 Object。	const int OB_NOT_THE_OBJECT
-51	重复注册。	const int OB_ALREADY_REGISTERED
-52	最后一个日志损坏。	const int OB_LAST_LOG_RUINNED
-53	没有 ChunkServer 被选择。	const int OB_NO_CS_SELECTED
-54	没有 Tablets 被创建。	const int OB_NO_TABLETS_CREATED
-55	无效。	const int OB_INVALID_ERROR
-57	十进制数溢出。	const int OB_DECIMAL_OVERFLOW_WARN
-58	十进制数不合法。	const int OB_DECIMAL_ILLEGAL_ERROR
-59	除零非法。	const int OB_OBJ_DIVIDE_BY_ZERO
-60	除法错误。	const int OB_OBJ_DIVIDE_ERROR
-61	不是十进制数。	const int OB_NOT_A_DECIMAL
-62	十进制数精度不一致。	const int OB_DECIMAL_PRECISION_NOT_EQUAL
-63	获取空取值范围。	const int OB_EMPTY_RANGE

结果码	含义	参数
-65	日志不同步。	const int OB_LOG_NOT_SYNC
-66	目录不存在。	const int OB_DIR_NOT_EXIST
-67	结束流式接口调用。	const int OB_NET_SESSION_END
-68	日志无效。	const int OB_INVALID_LOG
-69	日志填充。	const int OB_FOR_PADDING
-70	数据无效。	const int OB_INVALID_DATA
-71	已经完成。	const int OB_ALREADY_DONE
-72	操作被取消。	const int OB_CANCELED
-73	数据源变更。	const int OB_LOG_SRC_CHANGED
-74	日志不对齐。	数据格式错误。
-75	日志记录丢失。	const int OB_LOG_MISSING
-76	需要等待。	const int OB_NEED_WAIT
-77	不支持的操作。	const int OB_NOT_IMPLEMENT
-78	被零除。	const int OB_DIVISION_BY_ZERO
-79	值超出范围。	const int OB_VALUE_OUT_OF_RANGE
-80	超出内存限制。	const int OB_EXCEED_MEM_LIMIT
-81	未知的结果。	const int OB_RESULT_UNKNOWN
-82	数据格式错误。	const int OB_ERR_DATA_FORMAT
-83	可能循环。	const int OB_MAYBE_LOOP
-84	没有结果。	const int OB_NO_RESULT
-85	队列溢出。	const int OB_QUEUE_OVERFLOW
-101	死锁。	const int OB_DEAD_LOCK
-102	日志不完整。	const int OB_PARTIAL_LOG
-103	效验和错误。	const int OB_CHECKSUM_ERROR
-104	初始化失败。	const int OB_INIT_FAIL
-110	读出全零日志。	const int OB_READ_ZERO_LOG
-111	切换日志错误。	const int OB_SWITCH_LOG_NOT_MATCH

结果码	含义	参数
-112	写日志未开始。	const int OB_LOG_NOT_START
-113	致命错误状态。	const int OB_IN_FATAL_STATE
-114	停止状态。	const int OB_IN_STOP_STATE
-115	主 UpdateServer 已存在。	const int OB_UPS_MASTER_EXISTS
-116	日志未清除。	const int OB_LOG_NOT_CLEAR
-117	文件已经存在。	const int OB_FILE_ALREADY_EXIST
-118	未知的包。	const int OB_UNKNOWN_PACKET
-120	日志过大。	const int OB_LOG_TOO_LARGE
-121	同步发包失败。	const int OB_RPC_SEND_ERROR
-122	异步发包失败。	const int OB_RPC_POST_ERROR
-123	Libeasy 错误。	const int OB_LIBEASY_ERROR
-124	连接错误。	const int OB_CONNECT_ERROR
-125	不处于释放状态。	const int OB_NOT_FREE
-126	初始化 SQL 语境失败。	const int OB_INIT_SQL_CONTEXT_ERROR
-127	跳过无效的 Row。	const int OB_SKIP_INVALID_ROW
-131	系统配置表错误。	const int OB_SYS_CONFIG_TABLE_ERROR
-132	读取配置失败。	const int OB_READ_CONFIG_ERROR
-140	事务不匹配。	const int OB_TRANS_NOT_MATCH
-141	只读事务。	const int OB_TRANS_IS_READONLY
-142	Row 被修改。	const int OB_ROW_MODIFIED
-143	版本不匹配。	const int OB_VERSION_NOT_MATCH
-144	无效的地址。	const int OB_BAD_ADDRESS
-145	重复的 COLUMN。	const int OB_DUPLICATE_COLUMN
-146	进队列失败	const int OB_ENQUEUE_FAILED
-147	配置无效。	const int OB_INVALID_CONFIG
-1001	ChunkServer cache 未命中。	const int OB_CS_CACHE_NOT_HIT
-1002	ChunkServer 超时。	const int OB_CS_TIMEOUT

结果码	含义	参数
-1008	Tablet 不存在。	const int OB_CS_TABLET_NOT_EXIST
-1010	需要重试。	const int OB_CS_EAGAIN
-1011	扫描下一列。	const int OB_GET_NEXT_COLUMN
-1012	扫描下一行。	const int OB_GET_NEXT_ROW
-1013	反序列化失败。	const int OB_DESERIALIZE_ERROR
-1014	RowKey 无效。	const int OB_INVALID_ROW_KEY
-1015	不支持的查询模式。	const int OB_SEARCH_MODE_NOT_IMPLEMENT
-1016	错误的 Block 索引。	const int OB_INVALID_BLOCK_INDEX
-1017	错误的 Block 数据。	const int OB_INVALID_BLOCK_DATA
-1018	未找到。	const int OB_SEARCH_NOT_FOUND
-1020	查询的 Key 或者取值范围不在当前的 Tablet 中。	const int OB_BEYOND_THE_RANGE
-1021	完成移动块缓存。	const int OB_CS_COMPLETE_TRAVERSAL
-1022	Row 的结尾。	const int OB_END_OF_ROW
-1024	ChunkServer 合并失败。	const int OB_CS_MERGE_ERROR
-1025	ChunkServer 的 Schema 不兼容。	const int OB_CS_SCHEMA_INCOMPATIBLE
-1026	ChunkServer 服务未启动。	const int OB_CS_SERVICE_NOT_STARTED
-1027	Lease 过期。	const int OB_CS_LEASE_EXPIRED
-1028	合并超时。	const int OB_CS_MERGE_HAS_TIMEOUT
-1029	表已经被删除。	const int OB_CS_TABLE_HAS_DELETED
-1030	ChunkServer 合并取消。	const int OB_CS_MERGE_CANCELED
-1031	压缩依赖库出错。	const int OB_CS_COMPRESS_LIB_ERROR
-1032	超出磁盘空间。	const int OB_CS_OUTOF_DISK_SPACE
-1033	ChunkServer 移植已经存在的 Tablet。	const int OB_CS_MIGRATE_IN_EXIST
-1037	错误的 SSTable 数据。	const int OB_WRONG_SSTABLE_DATA
-1039	COLUMN GROUP 未找到。	const int OB_COLUMN_GROUP_NOT_FOUND
-1040	没有导入的 SSTable。	const int OB_NO_IMPORT_SSTABLE
-1041	导入 SSTable 不存在。	const int OB_IMPORT_SSTABLE_NOT_EXIST

结果码	含义	参数
-2001	UpdateServer 事务正在进行中。	const int OB_UPS_TRANS_RUNNING
-2002	MemTable 已经被冻结。	const int OB_FREEZE_MEMTABLE_TWICE
-2003	MemTable 已经被删除。	const int OB_DROP_MEMTABLE_TWICE
-2004	MemTable 启动的版本无效。	const int OB_INVALID_START_VERSION
-2005	UpdateServer 不存在。	const int OB_UPS_NOT_EXIST
-2006	UpdateServer 获取 memtable 或者 sstable 失败。	const int OB_UPS_ACQUIRE_TABLE_FAIL
-2007	主版本无效。	const int OB_UPS_INVALID_MAJOR_VERSION
-2008	UpdateServer 表未冻结。	const int OB_UPS_TABLE_NOT_FROZEN
-2009	UpdateServer 切主超时。	const int OB_UPS_CHANGE_MASTER_TIMEOUT
-2010	强制终止时间。	const int OB_FORCE_TIME_OUT
-3001	时间戳错误。	const int OB_ERROR_TIME_STAMP
-3002	交叉错误。	const int OB_ERROR_INTRESECT
-3003	超出取值范围。	const int OB_ERROR_OUT_OF_RANGE
-3004	RootServer 状态初始化。	const int OB_RS_STATUS_INIT
-3005	不提供旁路服务。	const int OB_IMPORT_NOT_IN_SERVER
-3006	发现取值范围。	const int OB_FIND_OUT_OF_RANGE
-3007	转换错误。	const int OB_CONVERT_ERROR
-3008	遍历 MergeServer 列表介绍。	const int OB_MS_ITER_END
-4001	内部状态错误。	const int OB_INNER_STAT_ERROR
-4002	Schema 版本太旧。	const int OB_OLD_SCHEMA_VERSION
-4003	输入参数错误。	const int OB_INPUT_PARAM_ERROR
-4004	找不到空项。	const int OB_NO_EMPTY_ENTRY
-4005	释放 Schema 失败。	const int OB_RELEASE_SCHEMA_ERROR
-4006	项目数量统计错误。	const int OB_ITEM_COUNT_ERROR
-4008	ChunkServer 缓存失败。	const int OB_CHUNK_SERVER_ERROR
-4009	没有新 Schema。	const int OB_NO_NEW_SCHEMA
-4010	子扫描请求过多。	const int OB_MS_SUB_REQ_TOO_MANY

结果码	含义	参数
-5000	启动 SQL 失败。	const int OB_ERR_SQL_START
-5001	语法解析初始化失败。	const int OB_ERR_PARSER_INIT
-5002	解析 SQL 失败。	const int OB_ERR_RESOLVE_SQL
-5003	产生物理执行计划错误。	const int OB_ERR_GEN_PLAN
-5004	未知的系统功能错误。	const int OB_ERR_UNKNOWN_SYS_FUNC
-5005	解析 SQL 语法时分配内存错误。	const int OB_ERR_PARSER_MALLOC_FAILED
-5006	解析 SQL 语法错误。	const int OB_ERR_PARSER_SYNTAX
-5007	COLUMN 大小错误。	const int OB_ERR_COLUMN_SIZE
-5008	重复列。	const int OB_ERR_COLUMN_DUPLICATE
-5010	未知的操作错误。	const int OB_ERR_OPERATOR_UNKNOWN
-5011	“*”使用错误。	const int OB_ERR_STAR_DUPLICATE
-5012	ID 不合法。	const int OB_ERR_ILLEGAL_ID
-5013	位置错误。	const int OB_ERR_WRONG_POS
-5014	值不合法。	const int OB_ERR_ILLEGAL_VALUE
-5015	列描述错误。	const int OB_ERR_COLUMN_AMBIGOUS
-5016	逻辑计划失败。	const int OB_ERR_LOGICAL_PLAN_FAILD
-5017	Schema 未设置。	const int OB_ERR_SCHEMA_UNSET
-5018	名称不合法。	const int OB_ERR_ILLEGAL_NAME
-5020	表格重复。	const int OB_ERR_TABLE_DUPLICATE
-5021	字符串被截断。	const int OB_ERR_NAME_TRUNCATE
-5022	表达式错误。	const int OB_ERR_EXPR_UNKNOWN
-5023	类型不合法。	const int OB_ERR_ILLEGAL_TYPE
-5024	主键已经存在。	const int OB_ERR_PRIMARY_KEY_DUPLICATE
-5025	已经存在。	const int OB_ERR_ALREADY_EXISTS
-5026	create_time 列已经存在。	const int OB_ERR_CREATETIME_DUPLICATE
-5027	modify_time 列已经存在。	const int OB_ERR_MODIFYTIME_DUPLICATE
-5028	索引非法。	const int OB_ERR_ILLEGAL_INDEX

结果码	含义	参数
-5029	Schema 无效。	const int OB_ERR_INVALID_SCHEMA
-5030	插入空 RowKey。	const int OB_ERR_INSERT_NULL_ROWKEY
-5031	COLUMN 未找到。	const int OB_ERR_COLUMN_NOT_FOUND
-5032	删除空 RowKey。	const int OB_ERR_DELETE_NULL_ROWKEY
-5033	插入 Join 列错误。	const int OB_ERR_INSERT_INNER_JOIN_COLUMN
-5034	用户为空。	const int OB_ERR_USER_EMPTY
-5037	没有进入权限。	const int OB_ERR_NO_AVAILABLE_PRIVILEGE_ENTRY
-5039	用户被锁。	const int OB_ERR_USER_IS_LOCKED
-5042	无效的列名。	const int OB_ERR_INVALID_COLUMN_NUM
-5043	未知的 PREPARE 语句。	const int OB_ERR_PREPARE_STMT_UNKNOWN
-5044	未知变量。	const int OB_ERR_VARIABLE_UNKNOWN
-5045	初始化 SESSION 失败。	const int OB_ERR_SESSION_INIT
-5046	旧权限版本。	const int OB_ERR_OLDER_PRIVILEGE_VERSION
-5047	缺少 RowKey 列。	const int OB_ERR_LACK_OF_ROWKEY_COL
-5050	用户已存在。	const int OB_ERR_USER_EXIST
-5051	密码为空。	const int OB_ERR_PASSWORD_EMPTY
-5052	授予 CREATE TABLE 权限错误。	const int OB_ERR_GRANT_PRIVILEGES_TO_CREATE_TABLE
-5053	错误的动态参数。	const int OB_ERR_WRONG_DYNAMIC_PARAM
-5054	参数大小错误。	const int OB_ERR_PARAM_SIZE
-5055	未知的功能错误。	const int OB_ERR_FUNCTION_UNKNOWN
-5056	建立 molidfy_time 列错误。	const int OB_ERR_CREAT_MODIFY_TIME_COLUMN
-5057	修改 Primary Key 错误。	const int OB_ERR_MODIFY_PRIMARY_KEY
-5058	重复的参数。	const int OB_ERR_PARAM_DUPLICATE
-5059	SESSION 过多。	const int OB_ERR_TOO_MANY_SESSIONS
-5061	PS 次数太多。	const int OB_ERR_TOO_MANY_PS

结果码	含义	参数
-5063	未知的 HINT 错误。	const int OB_ERR_HINT_UNKNOWN
-5999	SQL 结束错误。	const int OB_ERR_SQL_END

4 术语

B

表

一个表由若干列（在 schema 中定义）和任意行组成，有些表的每一行都存储了 schema 中定义的每一列，这样的表是稠密的，如基准数据的表一般是稠密的；另外一些表行只存储了部分的列，例如 UpdateServer中的表的修改增量，这样的表是稀疏的。

C

Commit Log

操作日志。UPS 每次更新操作会产生一条 commit log，并追加到已有的commit log 之后，并通过同步 commit log 实现主备复制。完整的commit log 序列代表了 CBASE 的更新历史。

CS

ChunkServer

基准数据服务器。存储 CBASE 数据库中的基准数据，提供数据读取服务、执行定期合并以及数据分发。

D

DIO

Direct Input-Output，直接输入输出。

H

行

一个行由若干列组成，有些时候其中的部分列构成主键（row key）并且整个表按主键顺序存储。有些表（如 select 指令的结果）可以不包含主键。

J

基准数据

CBASE 存储的某个时间点以前的数据快照，作为静态数据以SSTable 的格式存放在 ChunkServer 上。

L

LMS

Listener MergeServer。只负责从集群的内部表中查询主备集群的流量分布信息和所有的其他 MergeServer 的地址列表。

列

一个列由列 ID(column_id)及其值(column_value)组成。

M

MS

MergeServer， 合并服务器。主要提供协议解析、SQL 解析、请求转发、结果合并和多表操作等功能。

MVCC

Multi-Version Concurrency Control，多版本并发控制。

慢查询

超过指定时间的 SQL 语句查询。

O

OB

OceanBase，淘宝核心系统研发部的海量存储系统。

OLAP

On-Line Analytical Processing，联机分析处理。

OLTP

On-Line Transaction Processing，联机事务处理。

R

RPC

Remote Process Call，远程方法调用。

RS

RootServer，主控服务器。主要进行集群管理、数据分布和副本管理，并提供 Listener 服务。

S

schema

表的列的类型、值范围等以及该表与其他表的 join 等关系称为表的schema。

U

UPS

UpdateServer，更新服务器，是集群中唯一能够接受写入的模块，存储每日更新的增量数据。

UPS Master

UpdateServer Master，主 UpdateServer。主要处理实际读写请求。UPS Slave

UpdateServer Slave，备 UpdateServer。主要用于实时备份。

V

VIP

Virtual IP，虚拟 IP 地址。RootServer 主机由 VIP 决定。

Z

增量数据

CBASE 存储的某个时间点以后的更新数据，存放在 UpdateServer上，在内存中以 btree 和 hash table 的形式组织。