[openGauss的系统架构](https://www.modb.pro/db/30002)

openGauss是单机系统,业务数据存储在单个物理节点上,数据访问任务被推送到服务节点上执行,通过服务器的高并发,实现对数据处理的快速响应,同时通过日志复制把数据复制到备机,提供数据的高可靠和读扩展。

OLTP存储技术以高并发读写为主,数据实时性要求非常高,数据以行的形式组织，非常适合于面向外存设计的行存储引擎

OLAP主要面向大数据量分析场景，对数据的存储效率非常高，列存储引擎可以提供很高的压缩比，同时面向列的计算,cpu指令高速缓存和数据高速缓存的命中率比较高按需读取列数据,大大减少不必要的磁盘读取,非常适合数据分析场景

openGauss整个系统是可插拔、自组装的,并支持多个存储引擎来满足不同场景的业务诉求,目前支持的有行存储引擎，列存储引擎、内存引擎。

按照存储介质和并发控制机制,分为磁盘引擎和内存引擎

磁盘引擎主要是面向通用的，大容量的业务场景，内存引擎是面向容量可控、追求极致性能的业务场景

根据不同的业务场景对数据的访问和使用,openGauss进一步提供了astore(追加写优化格式)，cstore(列存储格式)和可扩展的数据元组和数据页面组织方式， 内存引擎中,openGauss当前提供基于Masstree结构组织的mstore(内存优化格式)数据组织格式。

数据存储引擎要解决的问题

1. 存储的数据必须满足ACID
2. 支持高并发读写、高性能；
3. 充分发挥硬件的性能,解决数据的高效存储和检索能力；

**存储引擎的整体架构**

从整个数据库服务的组成架构上来看,存储引擎向上对接sql引擎,为sql引擎提供标准的数据格式（元组），向下对接存储介质,通过存储介质提供的特定端口，对存储介质的数据完成读写操作。

OpenGauss存储引擎的特点

1. 统一的日志系统

在openGauss的存储引擎中,磁盘引擎和内存引擎共用一套日志系统,保证数据库故障恢复场景下,各引擎内和引擎之间的数据持久性和一致性，

2》多种并发控制和事务系统

在OpenGauss的存储引擎中,有两种并发控制和事务系统:适合高并发、高冲突、追求确定性结果的悲观并发控制,适合低冲突、短平快、低时延的乐观并发控制