

# 腾讯云HTAP数据库TiDB

产品简介

产品文档



腾讯云

## 【版权声明】

©2013-2017 腾讯云版权所有

本文档著作权归腾讯云单独所有，未经腾讯云事先书面许可，任何主体不得以任何形式复制、修改、抄袭、传播全部或部分本文档内容。

## 【商标声明】



及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。

## 【服务声明】

本文档意在向客户介绍腾讯云全部或部分产品、服务的当时的整体概况，部分产品、服务的内容可能有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或模式的承诺或保证。

## 文档目录

文档声明.....	2
产品简介.....	4
产品概述 .....	4
产品优势 .....	5
系统架构 .....	6

产品简介

## 产品概述

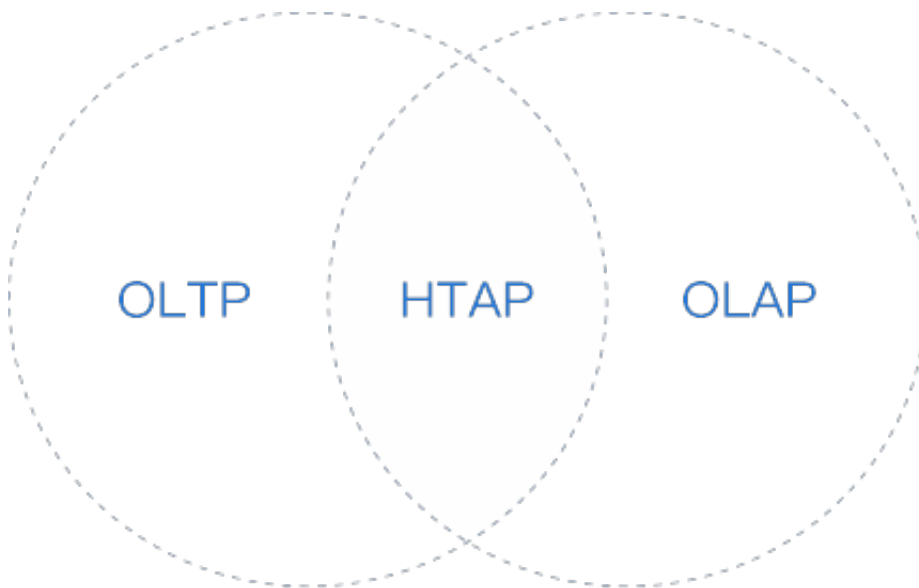
HTAP数据库 ( HTAP Database for

TiDB ) 是腾讯云基于国内最新的NewSQL开源数据库TiDB打造的一款同时支持 联机事务处理 ( OLTP ) 和 联机分析处理 ( OLAP ) 两种业务类型的分布式数据库产品。该产品打破了传统业务架构的界限，使得两种业务在同一份数据上进行，免去数据从运营数据库复制，落地，转换，加载到数据仓库的复杂流程，极大的降低了数据存储成本，缩短了数据分析处理的延迟时间，为实时的商业分析以及企业决策提供了保障。

该数据库的设计受到 Google Spanner/F1 论文的启发, 实现了自动的水平伸缩, 强一致性的分布式事务和基于 Raft 算法的多副本复制等重要的 NewSQL 特性; 并结合了 RDBMS 和 NoSQL 的优点, 使得部署简单, 弹性扩容和异步表结构变更不影响在线业务; 为用户提供了真正的异地多活, 自动故障恢复服务; 同时兼容 MySQL 协议, 使用户的迁移使用成本降到极低。

HTAP (Hybrid Transactional/Analytical Processing)

混合事务分析处理，是美国知名咨询公司Gartner提出的一个名词。



## 产品优势

### 可无限扩展的MySQL

Share Nothing 分布式架构，支持分布式事务和强一致性的水平弹性扩展。无论多大的数据量，简单增加节点即可解决。开发人员无需考虑分布式事务和一致性问题，极大简化代码设计和维护的复杂度，是传统分布式数据库中间件（Cobar, TDDL, MyCAT, KingShard）理想的替代方案。

### 兼容MySQL协议

支持MySQL社区所有周边工具（驱动，管理工具，应用框架）如 myloader/mydumper/MySQL JDBC Driver/Navicat/Workbench/WordPress/主流ORM框架等。

### 数据量越大读写越快

通过简单增加节点，提供无上限，线性扩展的高并发、高吞吐的集群处理能力。在大数据量下（千万级别），随机写入和复杂查询性能远优于MySQL。

### OLAP性能

可并行执行大数据量的复杂查询，包括聚合和Join等。

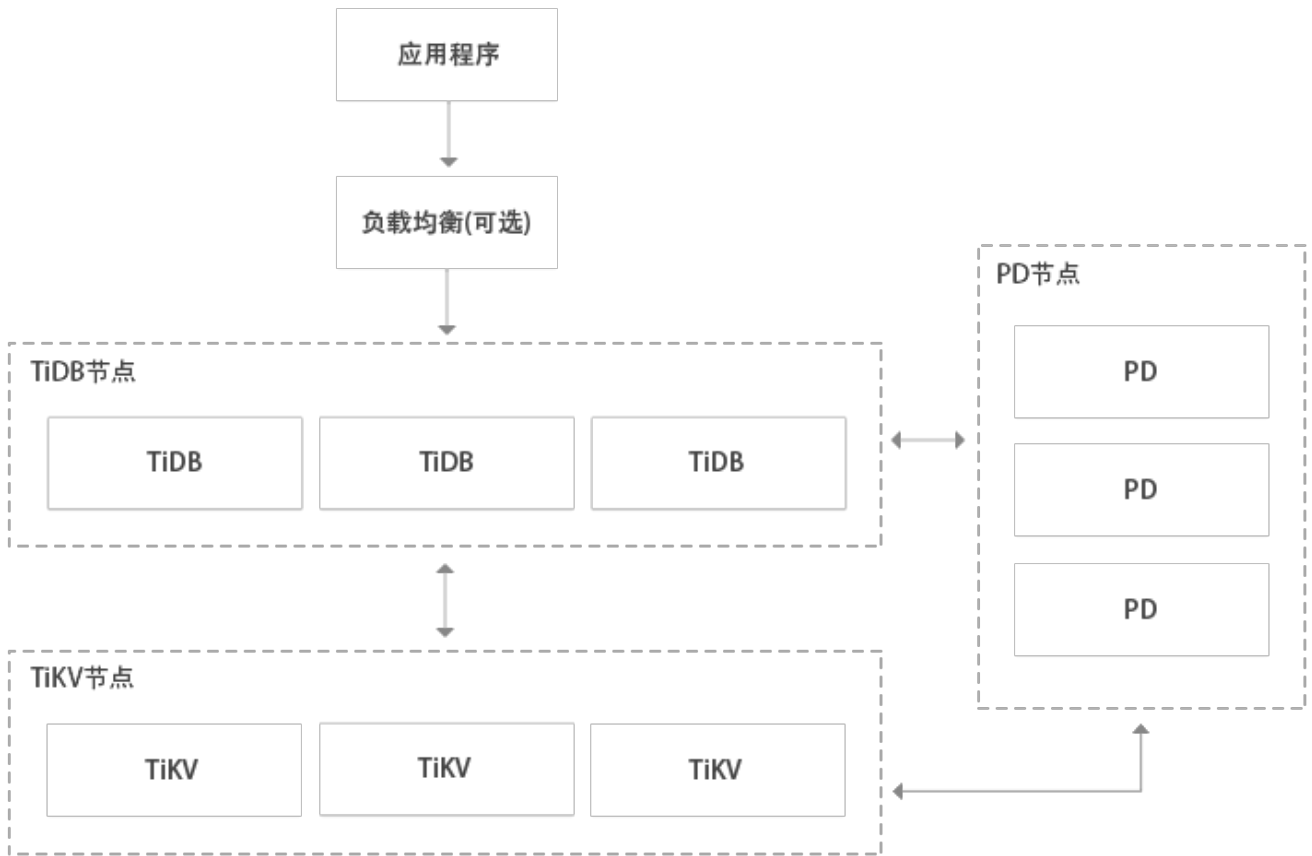
### 跨数据中心多活

基于Raft分布式选举算法，自动完成多副本写入，数据分片的拆分、聚合、重分布，100%跨可用区数据强一致，保证在任意可用区宕机时，数据不丢失且自动切换。

### 云数据库DBaaS

底层通过Kubernetes、Docker、多租户等技术，实现数据层面资源最有效的隔离、分配和调度。提供故障自动恢复，一键扩容，滚动升级等功能，使运维更简便，成本更低。

## 系统架构



HTAP数据库TiDB 集群主要分为三个组件：

### TiDB节点

TiDB 负责接收 SQL 请求，处理 SQL 相关的逻辑，并通过 PD 找到存储计算所需数据的 TiKV 地址，与 TiKV 交互获取数据，最终返回结果。

TiDB 是无状态的，其本身并不存储数据，只负责计算，可以无限水平扩展，可以通过负载均衡组件（如LVS、HAProxy 或 F5）对外提供统一的接入地址。

### PD 节点

Placement Driver (简称 PD) 是整个集群的管理模块，主要负责以下三个工作：

1. 存储集群的元信息（某个 Key 存储在哪个 TiKV 节点）；
2. 对 TiKV 集群进行调度和负载均衡（如数据的迁移、Raft group leader 的迁移等）；
3. 分配全局唯一且递增的事务 ID。

## TiKV 节点

TiKV 负责存储数据，从外部看 TiKV 是一个分布式的提供事务的 Key-Value 存储引擎。

存储数据的基本单位是 Region，每个 Region 负责存储一个 Key Range（从 StartKey 到 EndKey 的左闭右开区间）的数据，每个 TiKV 节点会负责多个 Region。TiKV 使用 Raft 协议做复制，保持数据的一致性和容灾。副本（副本数量可配置，默认保存三副本）以 Region 为单位进行管理，不同节点上的多个 Region 构成一个 Raft Group，互为副本。数据在多个 TiKV 之间的负载均衡由 PD 调度，这里也是以 Region 为单位进行调度。