KINGBASE | 金仓社区

数据库的假作真时真亦假
无为有处有还无

分享嘉宾：嘴毒、心狠、刺头刘——刘华阳

KING大咖面对面-北京站

新的方式来进行分享

我猜麦麦老师的表情应该是这样

01 数据库的哲学--假作真时真亦假

>>> 习惯成自然

>>> 跟随别人的思维

>>> 数据库重要吗

>>> 真的重要吗

数据库的哲学--假作真时真亦假

需求→数据库概念-→产品-→用户-→维护者→数据库产业

最初的数据库是**导航数据库**，主要采用**层次模型**（树形结构，支持一对多关系）和**网络模型**（更灵活，支持多种关系）

1970 年，IBM 的科学家 E.F. Codd 发表了一系列论文，提出了**关系模型 (Relational Model)** 的概念。

随着关系模型被广泛接受和商业化，**关系数据库管理系统 (RDBMS)** 开始兴起，并逐渐成为主流。

随着技术进步和互联网的爆发，出现了面向对象数据库、分布式数据库、数据仓库，以及近年来为了处理非结构化数据而诞生的 **NoSQL** 数据库、**云数据库**和**自治驾驶数据库**等。

数据库的哲学--假作真时真亦假

DBA 只是因为数据库操作复杂，知识凌乱，数据重要，公司

工作流程，紧急问题处理

一个整体需求中的一个环节而已

Wake Up

数据库的哲学--假作真时真亦假

假作真时真亦假--- 我的理解

当你认为这个公司离不开你的时候，你就开始了衰败

当你认为你的技术别人都不会，你就开始衰败

当你强调你的重要性的时候，你就开始衰败

没有一个公司可以容忍

你

重要的影响到公司运营

数据库的哲学--假作真时真亦假

当现实被伪装成假象时，真与假就失去了分界

这个世界本身就没有什么是你所相信的“真实”

数据库的本质不是性能、不是功能、不是分布式

而是

帮企业维持一份持续可用、可追溯、可信赖的数据状态

这是你的工作本质及向外提供的根本价值！！！

数据库的哲学--假作真时真亦假

民营企业自己用的话,优先免费开源的,实在不行才会用商用数据库或者云数据库

09:05

错误

优先是云数据库

云数据库也是个大坑

公众号 · 聪明的市场

技术死直男，思维一根筋，活该你赚不到钱

11月06日
已发表

技术死直男，思维一根筋，活该你赚不到钱 原创

167 6 11 5 1 0 ¥0.00

0

区别	项目传统部署 (CAPEX)	云服务 (OPEX)
会计处理	固定资产，分年折旧	当期费用，直接入账
税务处理	折旧周期长	当期可抵扣
现金流	一次性沉淀资金	持续性支出
财务风险	高（沉没成本）	低（可调整）

公众号 · 聪明的市场

02 SQL 能跑 ≠ 系统能用

>>>

数据库的哲学– 无为有处有还无

SQL 能跑 ≠ 系统能用

这句话我理解的

看到了“有”，误以为是本质

看不到 “无”，以为不存在

我们从一段有意思的故事开始！！

什么是有 你看的到吗?

数据库的哲学– 无为有处有还无

真正的力量来自不可见之处，可见之处往往只是表象、不具决定性

SQL 能跑

数据正确 SQL能跑

数据最终一致性 数据才能正确

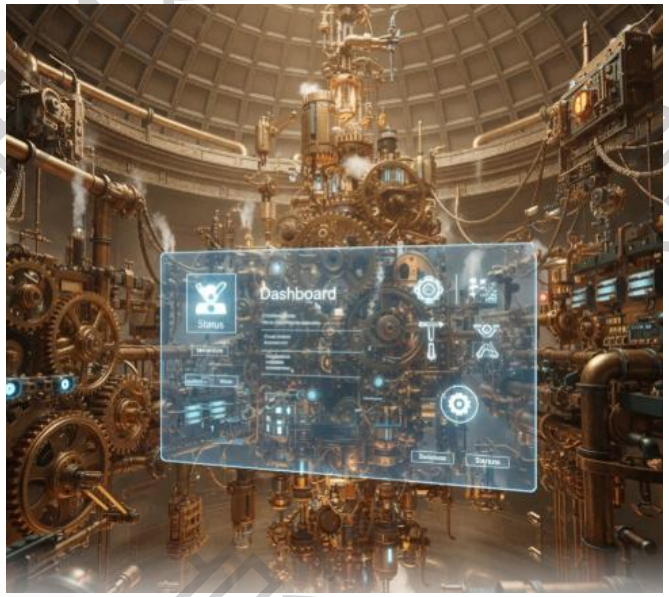
数据最终一致性 要靠日志来实现

日志的写入磁盘后 事务才能commit

日志写入效率提高 通过算法和硬件能力提升实现

更好的硬件 解决了数据库系统的运行效率

更好的算法 解决了数据库系统的运行效率



谁的成本更低，谁的实现更简单，谁实现的效率更高

谁解决了数据库的运行效率低成本简单实现的问题

数据库的哲学– 无为有处有还无

TPS 在数据库测试中，获得了非常高的成绩

但TPS里面都是极端的短事务

数据库可以挂接N个从节点

但从节点和主库延迟严重，无法进行业务正常使用

数据库记录的日志齐全

但是信息记录的比较杂乱，分析难度高

数据库报警指标齐全

但数据库告警阈值设置有误

HTAP 数据库能力非常强悍

但TP AP 数据之间同步延迟大 或 查询路由并不智能

数据库的哲学– 无为有处有还无

操作数据库≠ 真的懂数据库

如果仅仅是重复性的劳动，而不理解“无”（本质，底层，代价，需求，边界），并未掌握数据库本身，数据库是复杂的，每个角度，每个面都可以成为一个体系，一个理论，甚至一个小的学科，但掌握了一个角度，一个面并未真正掌握数据库本身。

我们看到，公司的数据库监控大屏，各种TPS,QPS的峰值波动，各种优化SQL的手段，添加索引后的酣畅淋漓。

我们看不到的是，数据库元数据定义的混乱，语义设计的混乱，补丁式的业务逻辑在表中的字段扩散，如同耗子洞一样的数据访问的链路。

表象繁华--- 背后虚无 看似有—本质无

数据库的哲学– 无为有处有还无

SELECT * FROM big_table WHERE col1 = 'value' ORDER BY col2 LIMIT 10;

最后我们以一个SQL作为结案陈词

看似简单，但我可以问出很多问题：

一条语句，建立索引是建立联合索引吗？ Limit 10 会影响执行语句的速度吗？ 在不同的数据量下？

数据库系统设计，应该是先过滤在排序，还是用排序的索引在进行过滤，那么不同的索引顺序，IO的代价

CPU的代价是怎么评估的，如果这个SQL在两种不同的存储结构上 B+TREE LSM –TREE, 如果 col1 是一个稀疏值

我又应该怎么建立索引，为什么每次limit 10的记录都是一样的，如果我的数据库引擎支持并行，那么我的数据

聚合方式又是什么，如果在PG里面这个数据经常重复，我怎么能让他的执行计划共享而不是在每个进程都产生

产生一个同样的执行计划，内存被浪费了，如果是MySQL prefer_ordering_index 是不是会出问题…….

最终，他只是一条SQL 能运行的SQL ！！

KINGBASE 金仓数据库 | 数据库平替用金仓

THANKS

成为世界卓越的数据库产品与服务提供商