

我能不能找一个大牛帮我调优

- 先把程序写出来，之后再让专家在生产环境中帮我调优
 - 这个想法是错误的.....
- 性能调优（目前情况下性能优化至最优）
 - 根据当前CPU能力、可用内存、I/O子系统等资源情况来设置相应参数
 - 通过索引、物理结构、SQL的优化，具体提高某一个查询的性能

如果有个专家能通过一些参数、技巧提高了你的系统一个数量级的性能，不能说这个专家牛逼，大概只能说明你的程序太烂了。



性能拙劣的罪魁祸首是错误的设计

- 提高整体性能
 - 技巧决定系统性能的下限
 - 设计决定系统性能的上限
- 比如，新闻的门户网站
 - 动态页面vs静态页面
 - 静态页面+内容管理系统



性能优化要考虑整体

- 性能指标都是有成本的、安全和优化中寻找平衡
- 性能指标以吞吐量为核心（每秒处理多少事务）
 - 而尽量不用一个事务几秒能处理完成
- 性能指标要考虑整体性
 - 优化手段本身就有很大的风险，只不过你没意识到罢了
 - 任何一个技术可以解决一个问题，但必然存在另一个问题的风险
 - 对于带来的风险，控制在可接受的范围才是有成果
 - 性能优化技术，使得性能变好，维持和变差是等概率的事件



使用优化工具

MySQL常用的工具

MySQLadmin

MySQLshow

SHOW [SESSION | GLOBAL] variables

SHOW [SESSION | GLOBAL] STATUS

Information_schema

SHOW ENGIN INNODB STATUS

SHOW PROCESSLIST

Explain

Show index

Show log

MySQL不常用，但是好用的工具

zabbix

监控主机、系统、数据库

Pt-query-digest

分析慢日志

MySQLslap

分析慢日志

sysbench

压力测试工具

MySQL profiling

统计数据库整体状态工具

Performance Schema

性能状态统计的数据

Workbench

管理、备份、监控、分析、
优化工具



整体层面的性能优化考虑

- 问题一：cpu负载高，io负载低
 - 内存不够
 - 磁盘性能差（磁盘问题、raid设计不好、raid降级）
 - **SQL的问题**
 - **并发锁机制的问题**
 - **事务设计问题，大量小数据IO**
 - **大量的全表扫描**



整体层面的性能优化考虑

- 问题二：IO负载高，CPU负载低
 - 大量小的IO执行写操作
 - Autocommit, 产生大量小IO
 - 大量大的IO执行写操作
 - SQL的问题
 - IO/PS磁盘限定一个每秒最大IO次数



整体层面的性能优化考虑

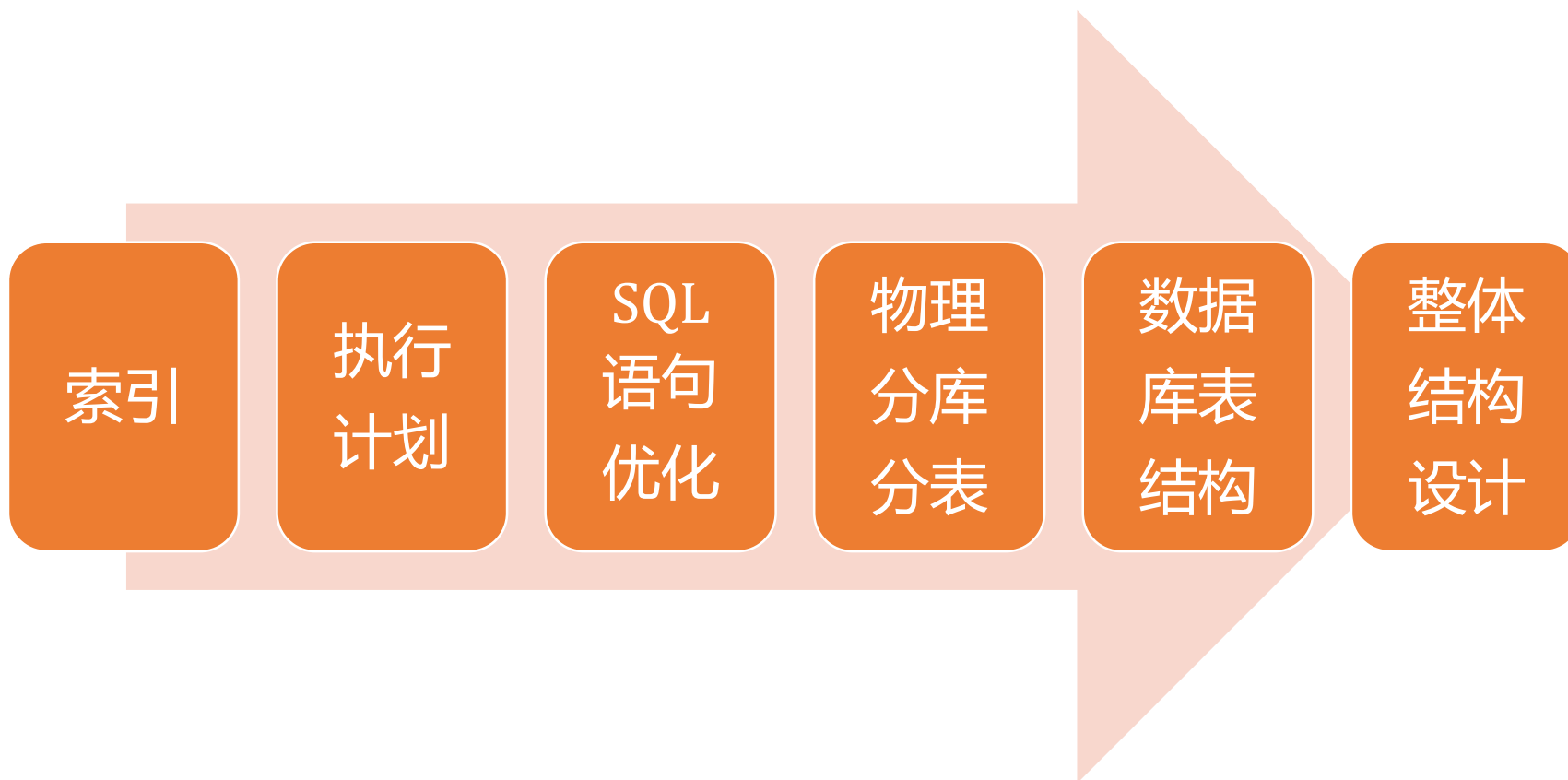
- 问题三：IO和CPU负载都高
 - 硬件不够用了
 - **SQL存在问题**

性能问题，90%的问题来源都是程序员的问题

开发环境到生产环境是一场灾难



SQL优化的方向



思考题

- 你对你常用的关系数据库系统中，去寻找一些针对优化的工具，去尝试使用一些性能的分析 and 监控工具（查看数据库官方Reference，首先使用官方的命令和工具）



End

下一章节，再见

