

MySQL高扩展架构设计

吴炳锡

知数堂联合创始人

吴炳锡

- 知数堂联合创始人及MySQL课程讲师
- 企业级MySQL架构设计服务顾问
- 曾任职飞信（新媒传信）数据库架构师，某彩票公司数据库DBA，10年+MySQL从业经验
- 专注MySQL优化，高可用，多IDC多活架构

大纲

- MySQL在高并发结构中的挑战
- 成功经验的总结
- Cache选择
- FAQ

挑战

数据量大

强一致性

可用

服务范围广

高并发

灵活

多IDC架构

Share nothing

优点

MySQL本身的特点

- 无执行计划缓存
- Query单核运算
- 在MySQL 5.7以前对于连接数据敏感（建议控制在300个以下）
- 基于存储引擎的解决方案(InnoDB, TokuDB, MyRocks, Spider)
- 不支持事务嵌套，不支持hash join
- 分布式事务支持比较弱

挑战

- 成功案列
 - 微信财富通
 - 物流
 - P2P信贷
 - 游戏行业
 - 互联网行业

The screenshot shows a search interface with a dropdown menu set to '全国' (National) and a search bar containing 'mysql'. A blue '搜索' (Search) button is on the right. Below the search bar, it indicates '共找到 10000+ 职位' (Found 10000+ positions). The results list four job listings:

- mysql DBA**
20-28万 | 北京,上海,深圳 | 大专
11分钟前
- MySQL DBA**
20-36万 | 上海,北京 | 本科及以上
24小时前
- MYSQL DBA高级工程师**
30-50万 | 深圳 | 大专及以上 |
22小时前
- 资深DBA工程师 (Mysql方向)**
18-30万 | 上海 | 大专及以上 |
23小时前

大纲

- MySQL在高并发结构中的挑战
- 成功经验的总结
- 灵活的拆分结构
- FAQ

成功经验总结

- 容量规划
- 分库分表
- 分布式事务
- DB调用
- 可用性
- 多IDC架构

成功经验总结

- 容量规划
 - 把DB做为一个存储资源（注意资源分配对齐）
 - 参考：
 - PCI-E 存储基础上
 - 单实例 innodb_buffer_pool：60-80G
 - 单实例大小1T左右，单库大小200G左右
 - 提倡单机多实例

成功经验总结

- 优点： 可控，方便迁移，内部做成DB资源管理平台易下手
- 反之： 单机单实例，存储在4T以上，备份管理非常难受

TokuDB

<https://github.com/XeLabs/tokudb>

成功经验总结

- 分库分表
 - 项目开始, ALL IN一个DB <200G
 - >200G 后: 先功能拆分, 按功能分配到不同的DB
 - 按功能拆后在单库大于200G后再考虑, 水平拆分
 - Range
 - Mod
 - 单实例达到1T左右, 考虑要分Set
 - 1-2000万 ->Set1
 - 2000万-4000万->Set2

成功经验总结

- 分库分表

先功能

AuthDB

UserDB

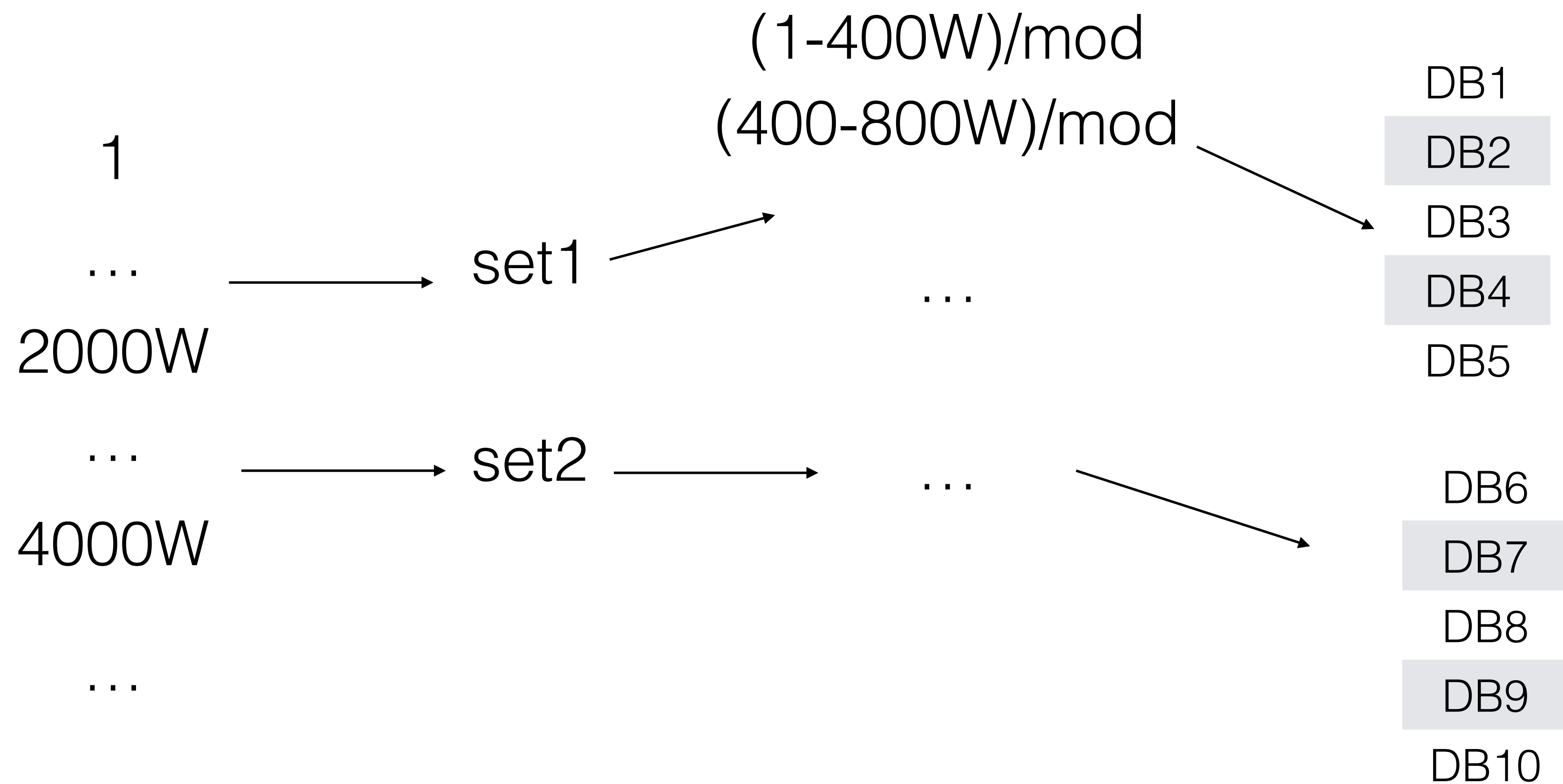
MsgDB

LogDB



成功经验总结

- 分库分表



成功经验总结

- 分布式事务（复杂事务）
 - 利用消息队列解藕
 - 引入状态机的概念
 - 保证针对对象自己的事务没完成，不能开始新的事务



成功经验总结

- DB调用
 - 虚拟功能 DB
 - 单DB只对自己的服务开放权限，拒绝其它服务直接访问其它功能DB
 - 服务之间走服务调用
- 流式调用
 - 把并发控制在一定范围之内，引入过载保护
- 长服务链调用

成功经验总结

- 可用性
 - 高可用
 - MHA
 - 自主实现
 - 服务架构上的支持
 - DB在故障切换中程序会不会异常
 - DB切换后故障，有没有备选方案（2-3层的兜底）



成功经验总结

- 可用性
 - 自动化的安全阈值控制
 - 高可用切换过程中产生的DB不可用处理
 - 多写的机制中数据一致性是不是方便校验
 - 后期数据补偿方案



成功经验总结

- DB调用
 - 长服务链调用
 - 超时的控制
 - 中间失败环境处理
 - 放弃线程，引入callback机制

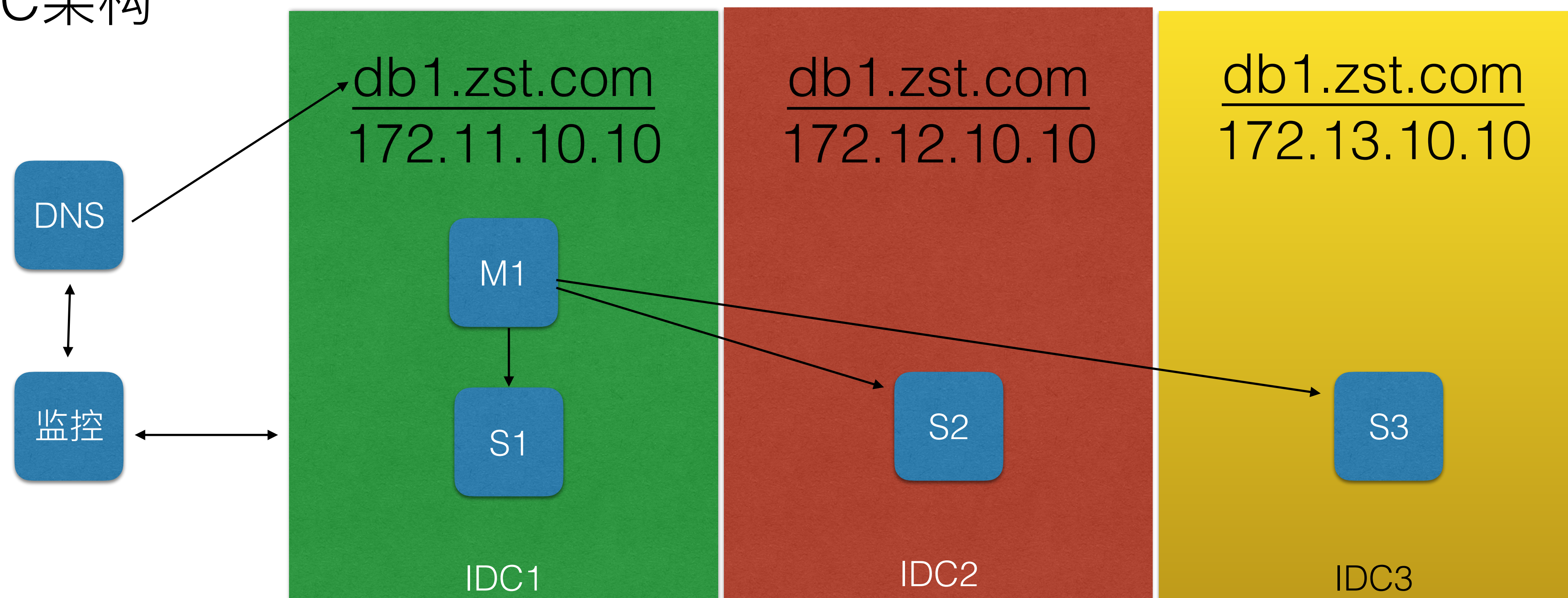
成功经验总结

- 多IDC结构
 - 单点写入，其它IDC就近读取
 - passport
 - 多点写入，总线汇总
 - 叫号器
 - linkedin databus
- 对外提供访问： DNS

SmartDNS

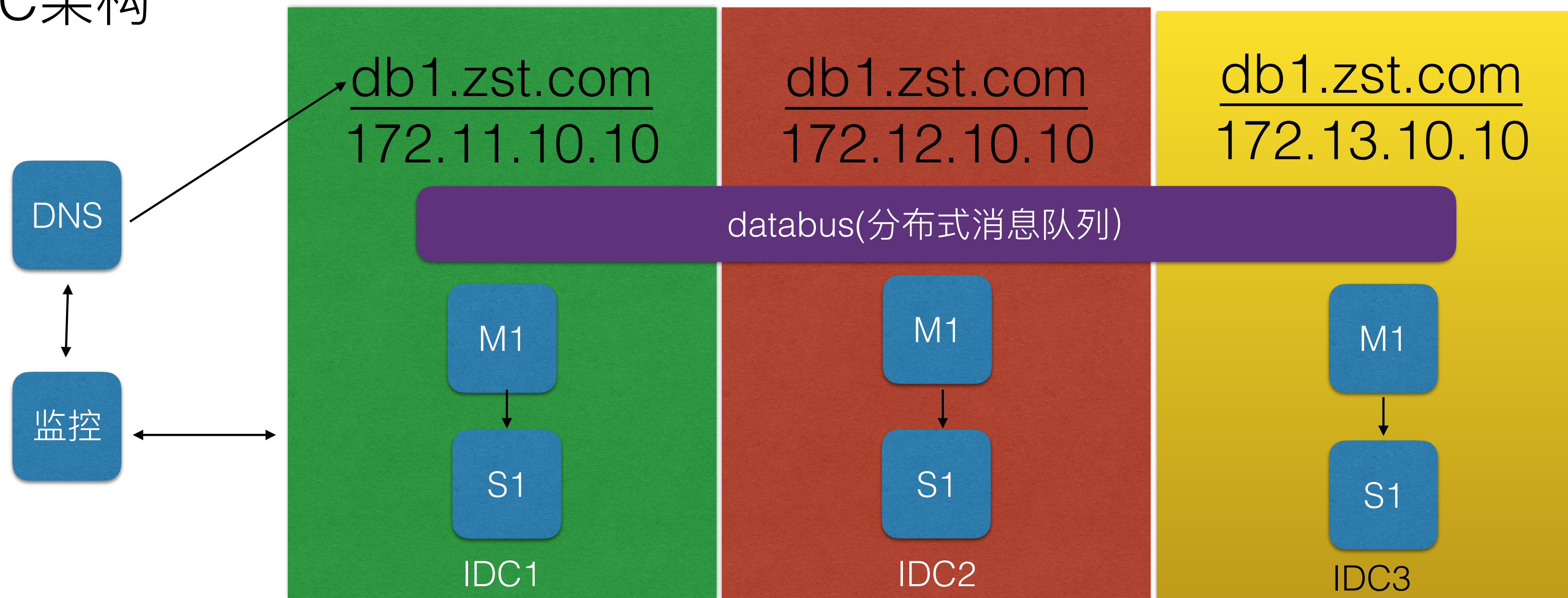
成功经验总结

- 多IDC架构



成功经验总结

- 多IDC架构



大纲

- MySQL在高并发结构中的挑战
- 成功经验的总结
- Cache选择
- FAQ

Cache选择

- 项目开始可以不用考虑Cache
- 只缓存调用最多的（处理冒尖的方案）
- 推荐：
 - Redis-cluster
 - MongoDB Cluster
- 易失性Cache
 - Memcache
 - Redis
- 非易失性Cache
 - Redis
 - MongoDB
 - MySQL NDB Cluster
- 自主研发
 - 语言选择
 - 先进性？

谢谢，希望有所帮助

- 指导思想
- 动态分库
- 基于Consul的域名接入
- 代理