

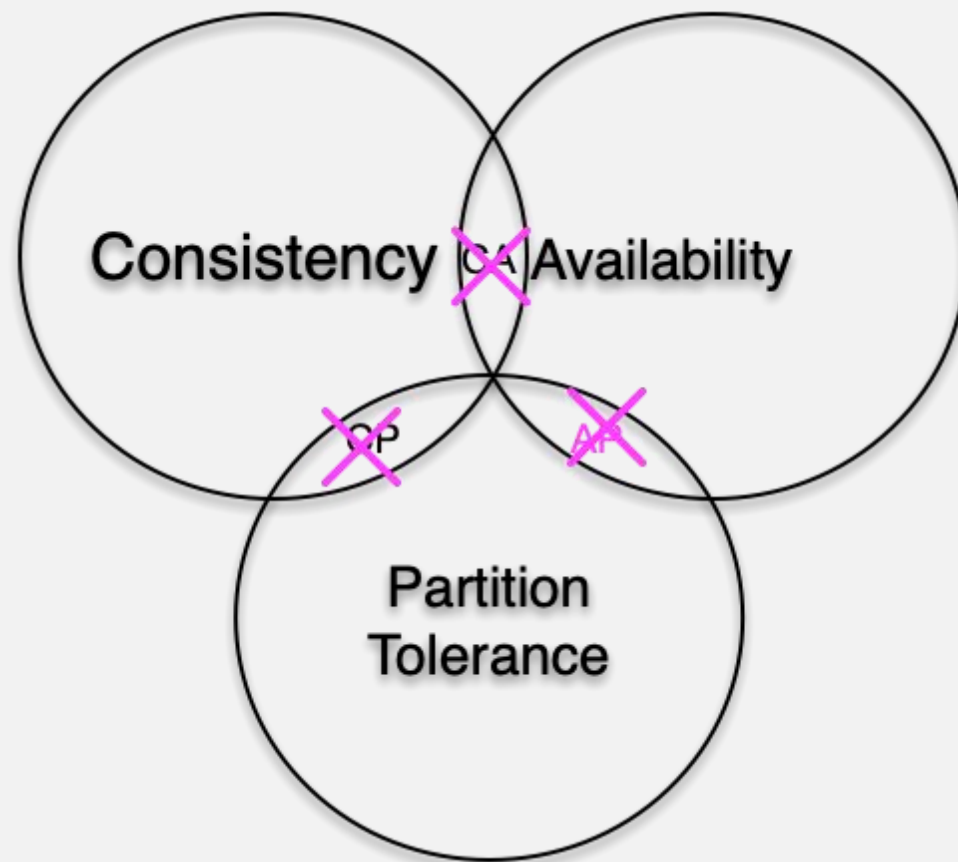
课时31

如何设计足够可靠的分布式缓存体系以满足大中型移动互联网系统的需要

1. 传统 CAP 的突破
2. BASE 理论
3. 一致性问题及应对
4. 分布式系统多区数据一致性案例

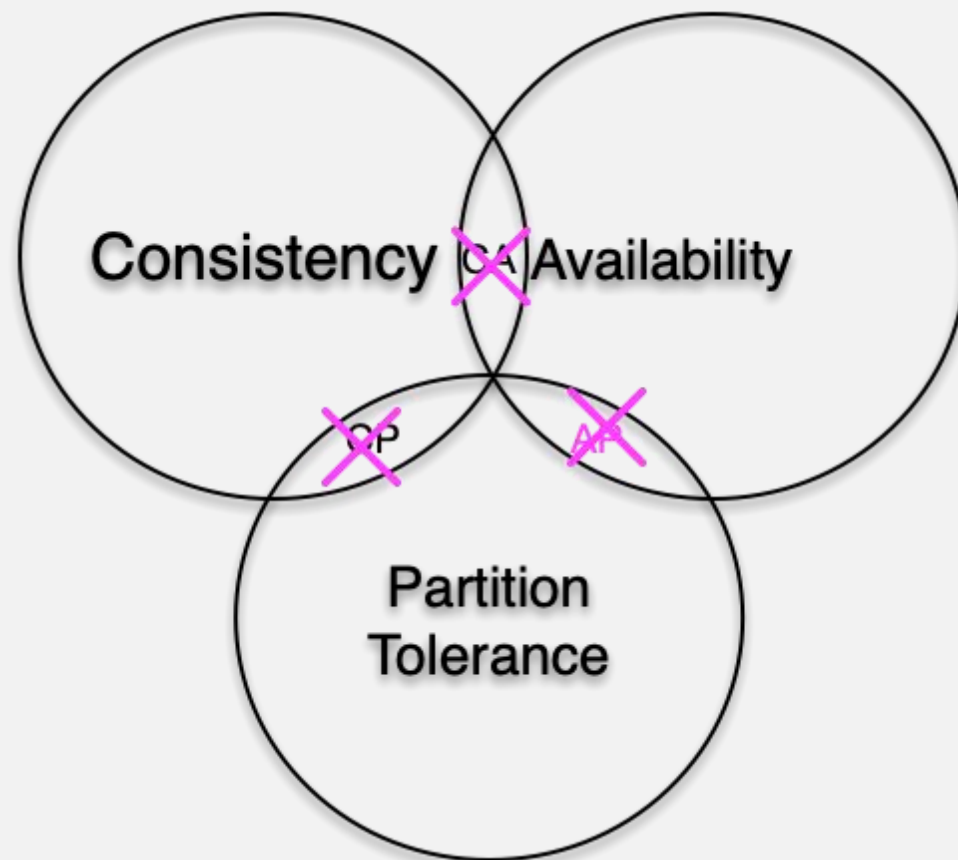
传统CAP的突破

- 当前互联网分布式系统发展趋势
 - 私有云、公有云融合
 - 全国、全球多区部署
- CAP理论需要进行修正及突破
 - 每种要素存在多种中间状态
 - 三要素之间存在相互关联
 - 不能简单应用成三选二



传统CAP的突破

- 2012年, Eric Brewer 再次发布: 《CAP 理论十二年回顾: "规则"变了》
- 对CAP理论, 特别是CAP使用中的一些误区, 进一步进行修正、拓展
 - 三选二有误导性
 - CAP与其他理论的比较
 - 大部分时光, 分区不发生, 可同时提供CA
 - 分区发生时如何应对, or 如何应对潜在的分区问题



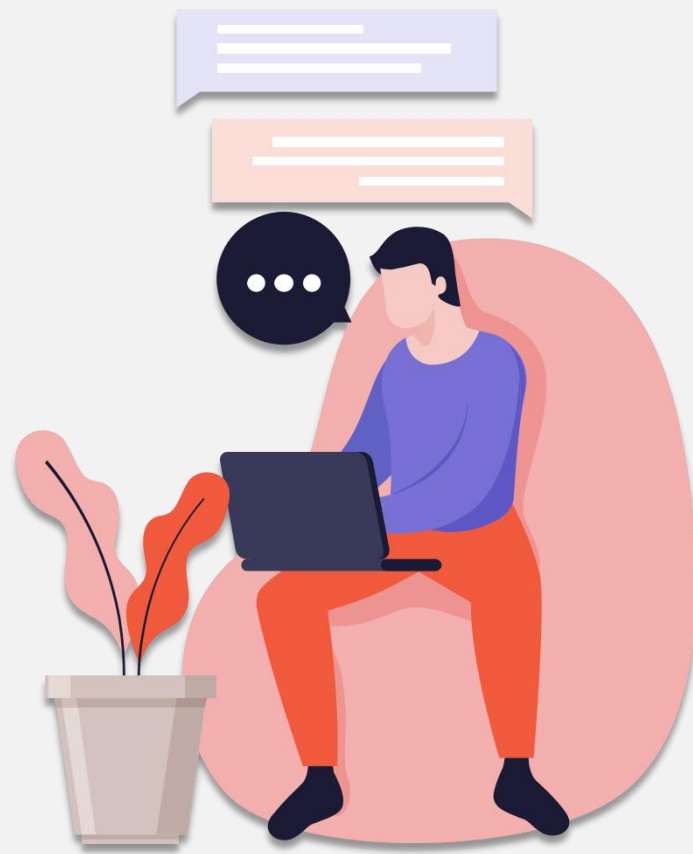
传统CAP的突破

- 分区发生时，对于分区的应对
 - 探测感知分区的发生
 - 分区模式下的处理策略
 - 启动恢复，并补偿分区模式期间的错误



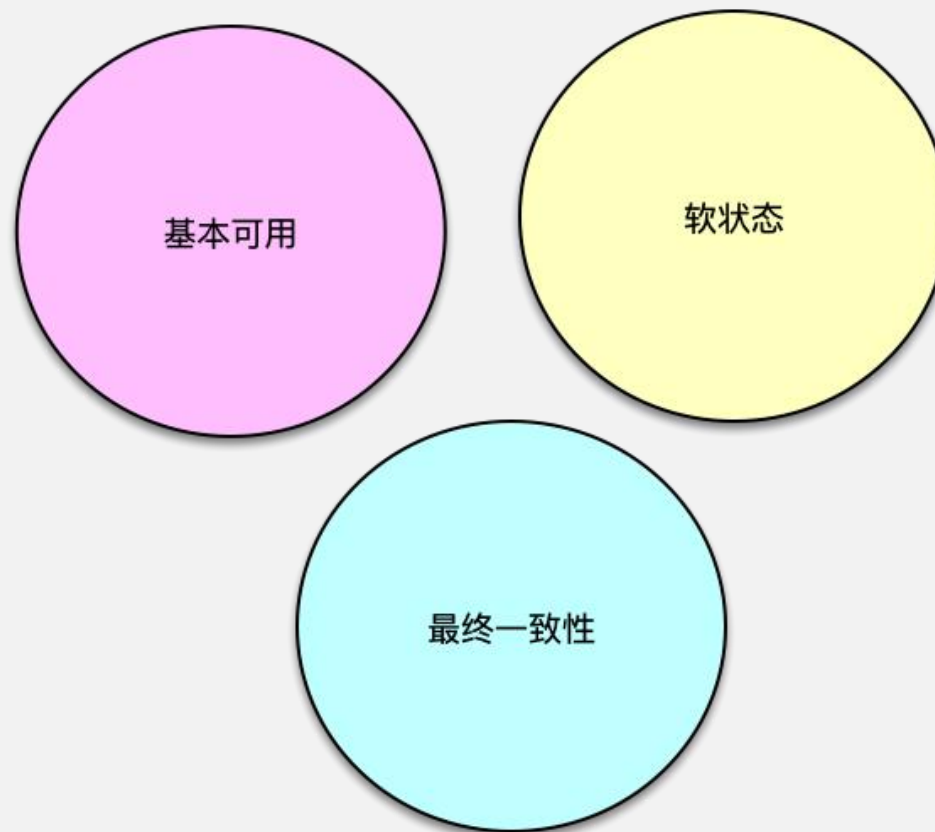
BASE 理论

- BASE理论最初由Brewer 及他的同事提出
- 当前大规模互联网系统分布式实践已验证，并同时CAP定理进行了延伸
- BASE理论及实践是对一致性和可用性权衡的结果
- 基本思想：分布式系统功能要适当权衡，尽力保持系统稳定性，确保主体功能可用和最终一致



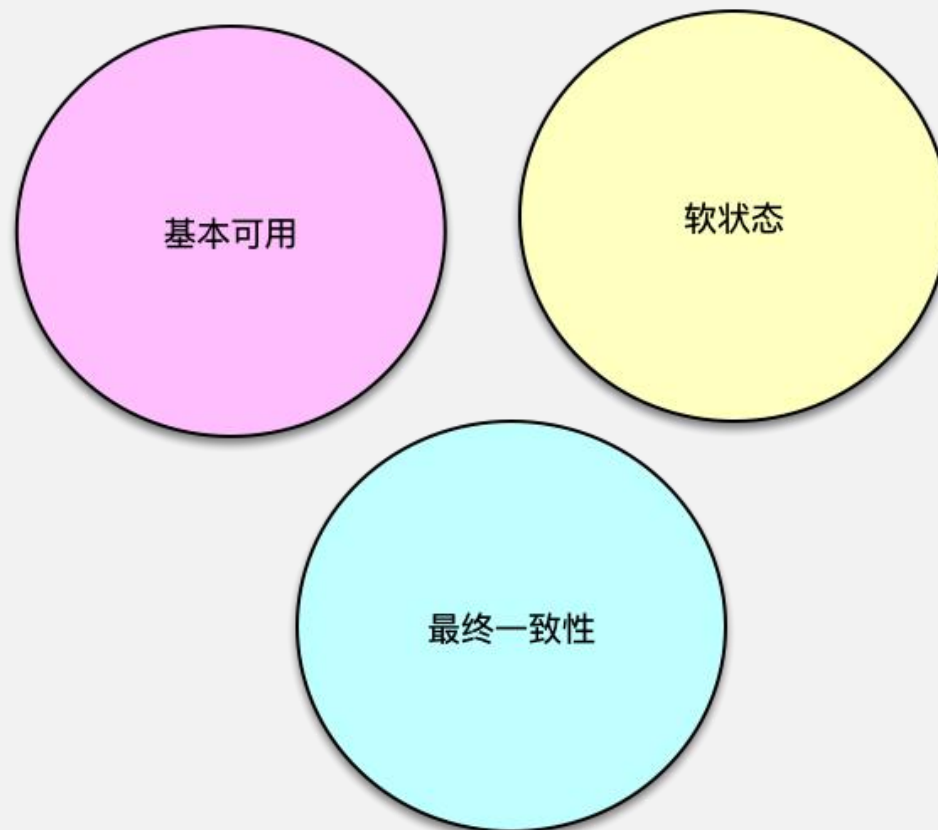
BASE 理论

- BASE 三要素
 - 基本可用 (Basically Available)
 - 软状态 (Soft State)
 - 最终一致性 (Eventual Consistency)



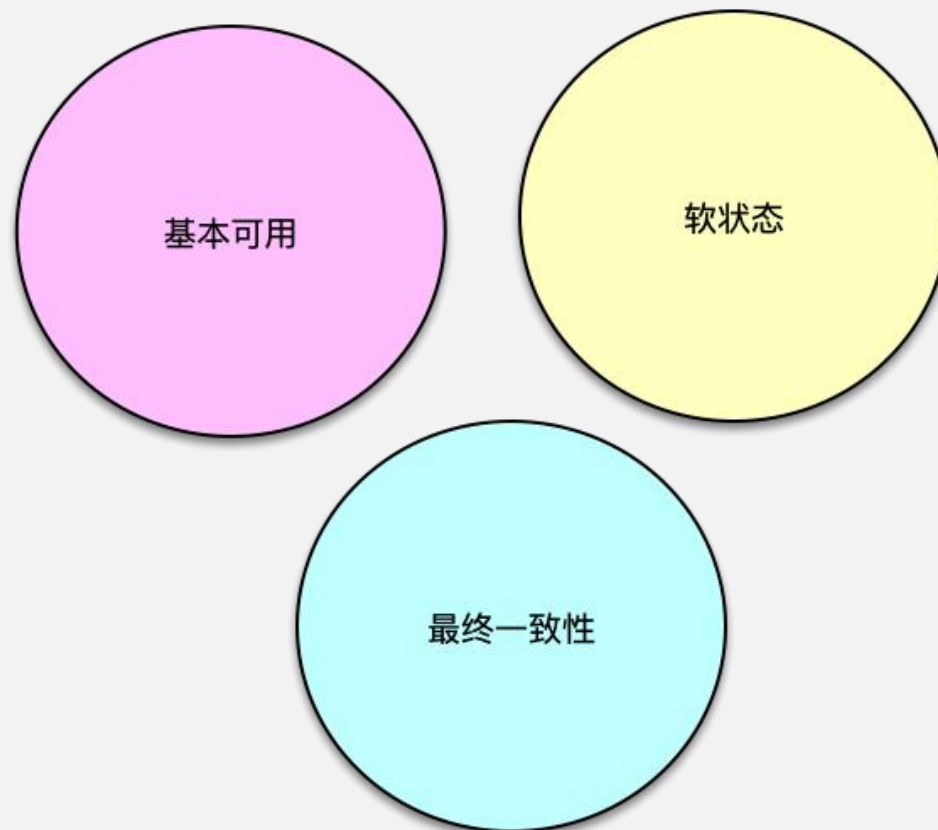
BASE 理论分析

- 基本可用
 - 系统故障时，允许损失部分可用性
 - 损失范围：部分SLA（响应时间、处理性能），部分核心功能
 - 目标：系统整体稳定，核心功能基本可用
- 软状态
 - 允许系统存在中间状态
 - 故障发生时，各分区同步延时或暂停，但仍在分别对外提供服务



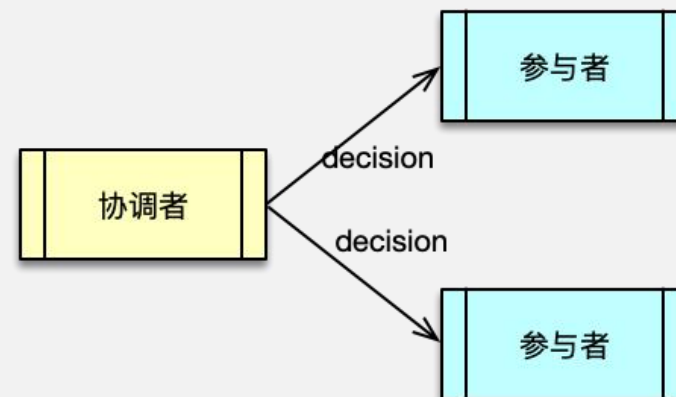
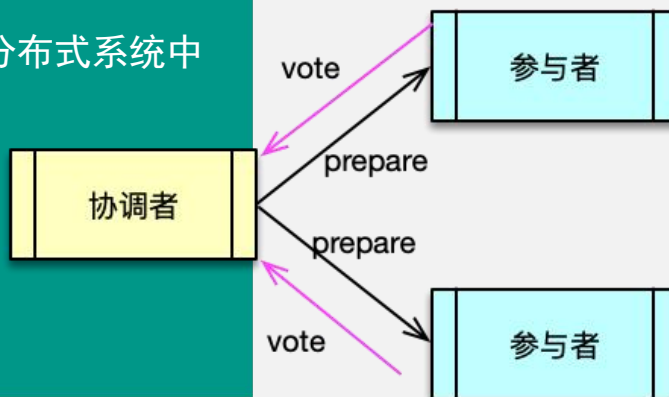
BASE 理论分析

- 最终一致性
 - 不需要实时保持强一致性，系统故障时，可以容忍数据的不一致
 - 故障恢复后，数据进行同步，最终达到一致的状态
- BASE 理论面向大中型互联网分布式系统
 - 用户体验第一，可用性优先
 - 故障时，牺牲部分可用性、一致性，来保持系统核心功能的可用性
 - 系统故障恢复后，确保最终达到一致



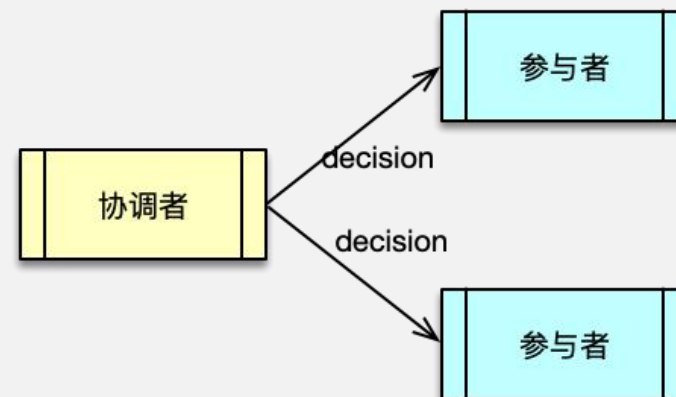
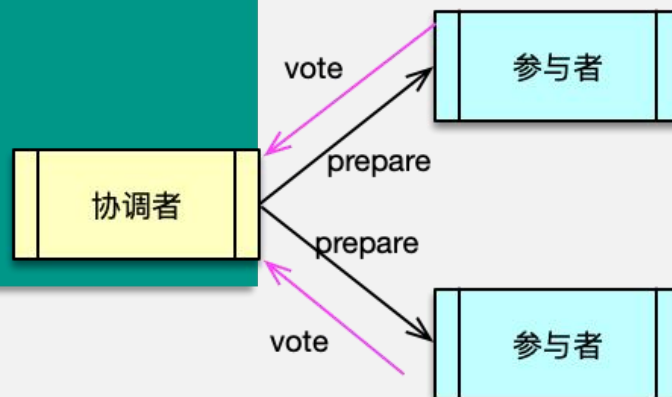
一致性问题及应对

- 分布式系统中，数据存储多个副本，分布在不同的物理的机器
- 保持分布式系统的一致性，核心是解决分布式系统中的数据一致性
- 保持数据一致性的方案
 - 分布式事务
 - 主从复制
 - 业务层消息总线



一致性问题及应对

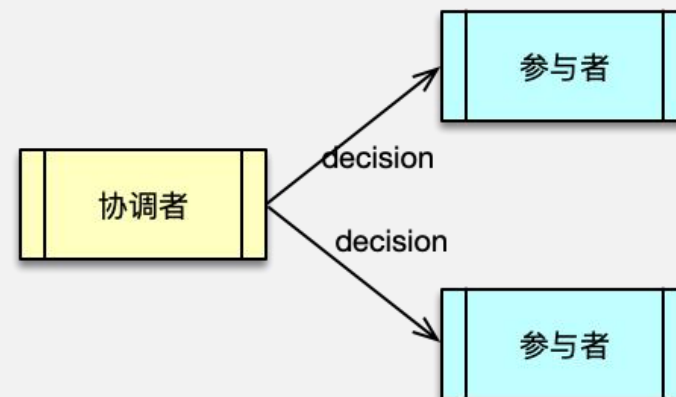
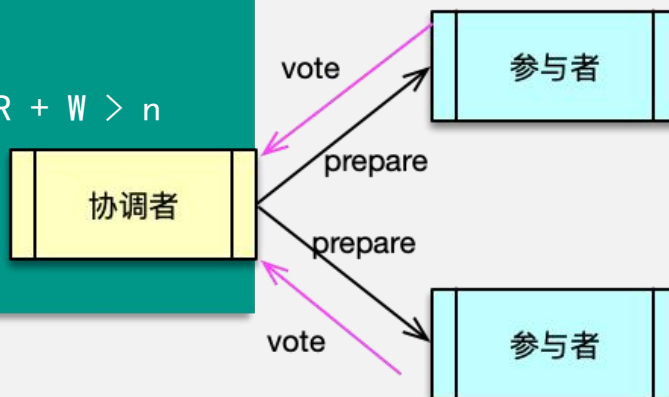
- 分布式事务，各节点均能正常执行事务内一系列操作才会提交，否则就进行回滚，强一致
- 分布式事务方案
 - 2PC、3PC
 - Paxos
 - Others (Zab, Raft...)



一致性问题及应对

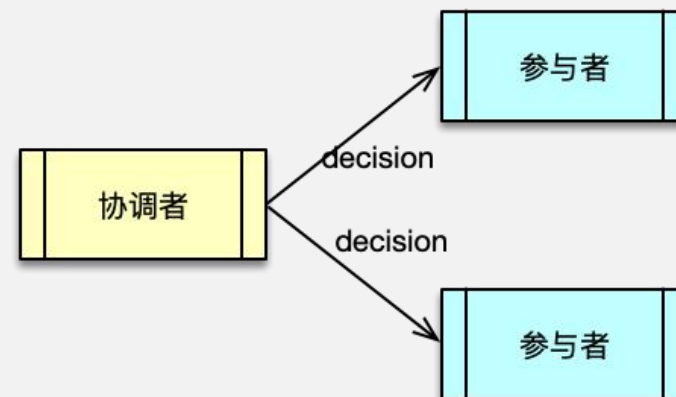
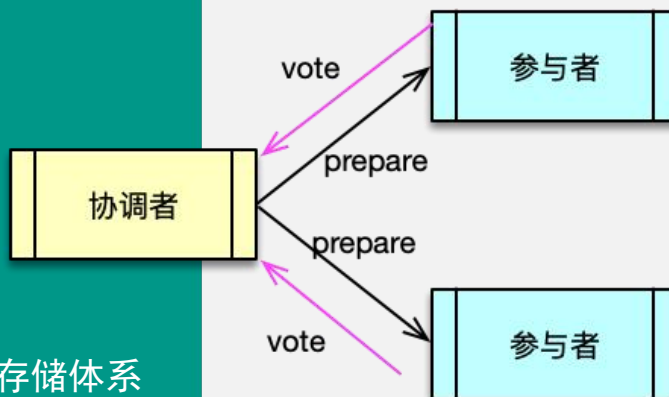
- 分布式事务之版本仲裁

- 每个数据副本附带版本信息
- 一致性仲裁: $R > n/2$ && $W > n/2$, $R + W > n$
- 仲裁者: 协调者, 业务client
- 仲裁后操作: 版本选择 or 合并



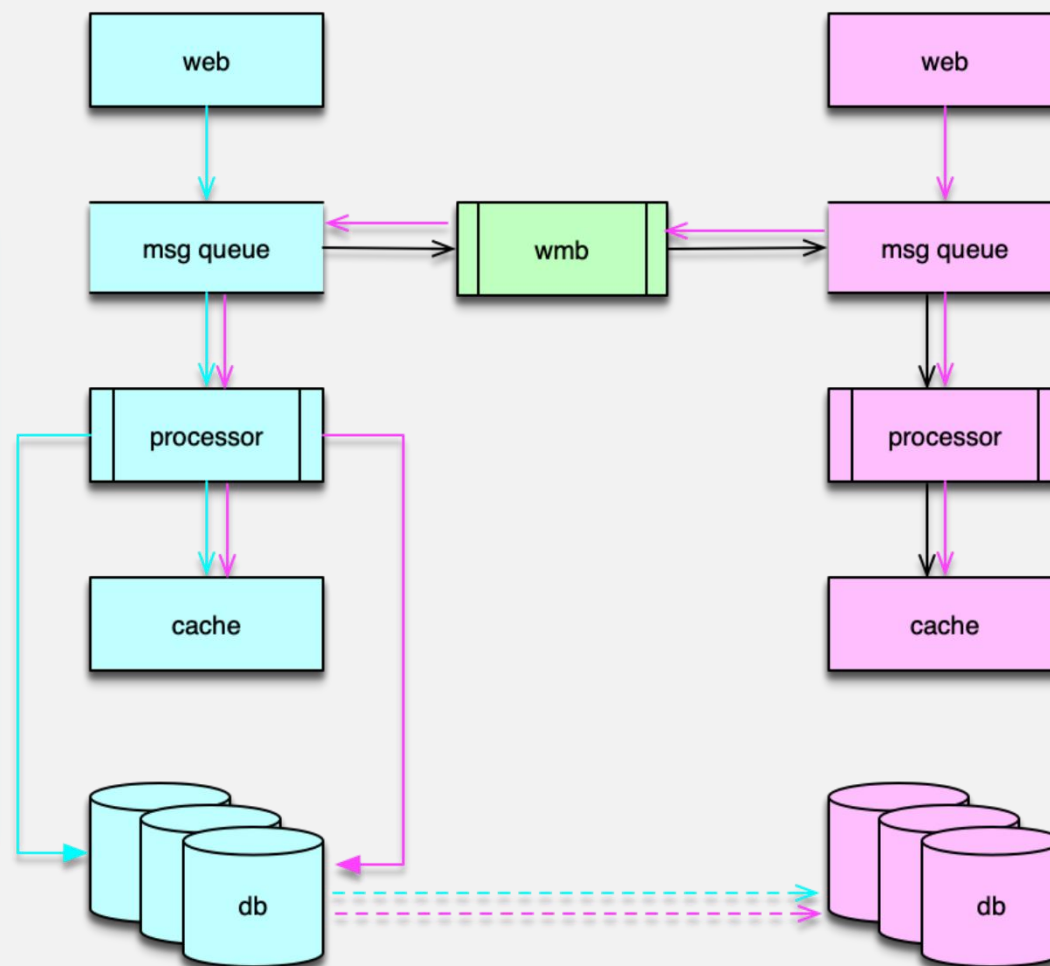
一致性问题及应对

- 主从复制
 - Mysql, Hbase
 - Redis
- 业务层消息总线
 - 异地多活
 - 各区域相互分发: push, pull
 - 处理本区及异地消息, 更新缓存/存储体系



一致性问题及应对

- 分布式系统多区数据一致性案例
 - 通过消息队列进行同步
 - 消息写本地队列，分区异地读取
 - 分区故障发生，读取失败，消息仍保留在各区本地消息队列中
 - 分区故障过程中，各区只处理本地事件
 - 分区故障排除后，继续读取异地消息队列，完成数据同步，实现数据一致性



Next: 课时32 《一个典型的分布式缓存系统是什么样的?》