

CAP 理论

- 起源
- 定义
- 应用场景

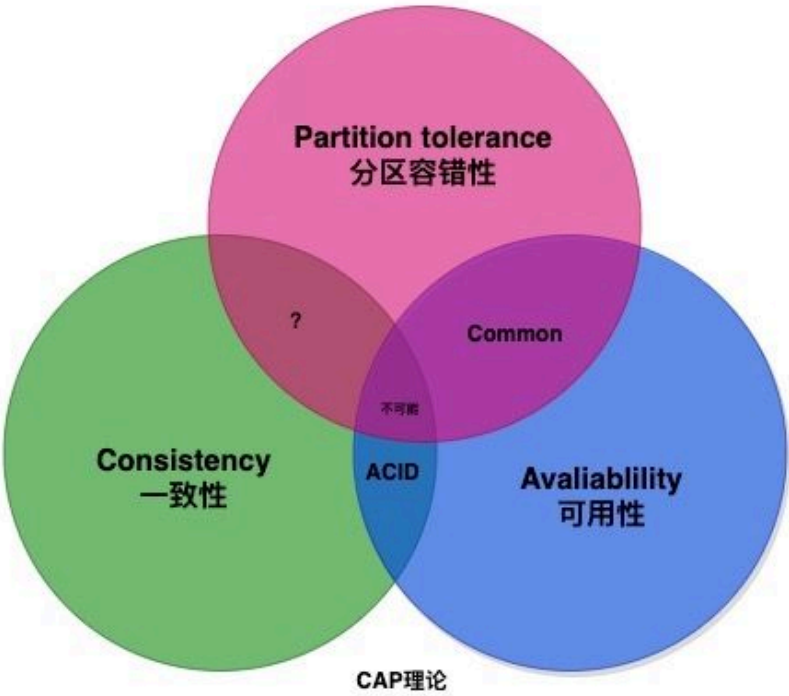
只要提到分布式系统，CAP 原理就是没法绕开的。因为 CAP 理论是分布式系统的基石。既然如此，那么什么是 CAP 原理，它又是如何产生的呢？

起源

CAP 原理最早出现在 2000 年，是加州大学伯克利分校（University of California, Berkeley）的 Eric Brewer 教授在 ACM 组织的 Principles of Distributed Computing (PODC) 研讨会上提出猜想。在 2002 年，麻省理工学院（MIT）赛斯·吉尔伯特和南希·林奇发表了布鲁尔猜想的证明，使之成为一个定理。

定义

在理论计算科学中，CAP 定理（CAP theorem），又称为布鲁尔定理（Brewer's theorem），它指出对于一个分布式计算系统来说，不可能同时满足以下三点：一致性（Consistency）（等同于所有节点访问同一份最新的数据副本）可用性（Avaliability）（每次请求都能获取到非错的响应——但是不保证获取的数据为最新数据）分区容错性（Partition tolerance）（以实际效果而言，分区相当于对通信的时限要求。系统如果不能在时限内达成数据一致性，就意味着发生了分区的情况，必须就当前操作在 C 和 A 之间作出选择）



根据定理, 分布式系统只能满足三项中的两项而不可能满足全部三项。理解 CAP 理论的最简单方式是想象两个节点分处分区两侧。允许至少一个节点更新状态会导致数据不一致，即丧失了 C 性质。如果为了保证数据一致性，将分区一侧的节点设置为不可用，那么又丧失了 A 性质。除非两个节点可以互相通信，才能既保证 C 又保证 A，这又导致丧失了 P 性质。也就是说，如图所示的三者交叉的位置，是不可能实现的。

应用场景

既然 CAP 三者同时满足的条件是不可能的，所以在系统设计的时候就需要作出取舍，比如弱化某些特性来支持其他两者。

弱化一致性对结果不敏感的应用，可以允许在新版本上线后过一段时间才能更新成功，不保证强一致性，保持最终状态的一致性。典型的如，静态网站。

弱化可用性对结果一致性很敏感的应用，如银行取款机，当系统发生故障时停止服务。如清结算，转账等。目前的 paxos、raft 等算法都是为保证强一致而设计的。这些系统会在内部异常时拒绝或者阻塞客户端请求。

弱化分区容错性现实中，网络分区出现概率较小，但难以避免。实践中，网络通信通过双通道等机制增强可靠性，达到高稳定的网络通信。