# CAP

分布式系统中不可能同时保证一致性、可用性和分区容错性，这就是CAP定理。

* 一致性（consistency）：所有节点在同一时刻都能看到同样的数据，对于调用者而言数据具有**强一致性**
* 可用性（availability）：所有请求都可以在一定时间内收到确定是否成功的响应，不会一直等待
* 分区容忍性（Partition tolerance）：在网络发生分区的情况下，仍然能够对外正常提供服务（强一致性和可用性仍有保证）

虽然不能使得某个分布式场景同时满足这三个指标，但可以使之满足其中两个，即我们可以基于CAP理论，根据不同场景的不同业务要求进行算法上的适当权衡，这被称为CAP权衡方案：

* CA：最典型的就是2PC（两阶段提交），当网络发生分区时，在一定时间内不可能达到节点状态一致，即不具备P分区容错
* CP：最典型的就是多数派选举协议，如Paxos、raft，当网络分区发生时，为了达成共识，请求只能被动等待分区恢复，即不具备A可用性
* AP：最典型的就是Gossip冲突解决协议，当网络分区发生时，被分割的节点如果仍然能够对外提供服务却不能相互通信，这将导致节点状态不一致，即不具备C可用性。

对于分布式系统而言，网络分区时无可避免的，比如消息丢包，节点宕机，现实情况中我们面对的是一个不可靠的通信网络和有一定概率宕机的设备，在分布式系统设计时我们不得不考虑P，而只能在一致性和可用性之间进行权衡，

对于分布式系统工程来说，CAP理论更合适的描述是：在满足分区容错的前提下，没有算法能够同时满足一致性和可用性。

我们需要注意的是，在分布式系统中，一致性是有不同程度的，

* 强一致性：而CAP中的C就是强一致性，他要求多个节点能像单个节点一样操作并具备原子性，在时序上一致
* 序列一致性：不要求保证时序一致，但要求保证顺序一致，也就说如果某操作A先于操作B，那么操作B后client都可以看到操作A的结果
* 最终一致性：放宽对时序的要求，只要保证节点之间的数据是最终一致的

通常情况下，我们对于不是那么重要的数据存取操作，往往会降低一致性来增加可用性，只要能满足最终一致性即可，但对于敏感的元数据而言，则采用强一致性而忽略可用性。

# BASE

Base理论的核心思想是：即使无法做到强一致性，但每个应用都可以根据自身业务特点，采用适当的方式来使系统达到最终一致性。Base理论是对CAP理论中的一致性和可用性权衡的结果，来源于对大规模互联网系统分布式实践总结。

* 基本可用（basic availability）：指分布式系统出现不可预知的故障时，允许损失部分可用性
* 软状态（soft state）：允许系统中数据存在中间状态，并认为该中间状态的存在不会影响系统的整体可用性，即允许系统在不同节点的数据副本之间数据同步过程存在时延
* 最终一致（eventual consistency）：最终一致性强调所有数据副本，在经过一段时间同步后，最终能够达到一个一致的状态。

Base理论和传统事务的ACID特性是相反的，它完全不同于ACID的强一致性模型，而是通过牺牲强一致性来获取可用性，并允许数据在一段时间内是不一致的，但最终达到一致状态。

# FLP

FLP不可能性是分布式领域中一个非常著名的定理，他给出了一个结论：在异步通信场景，即使只有一个节点失败，没有任何算法能保证非失败节点能够达到一致性。

这意味着，在假设网络可靠、节点只会因为崩溃而失效的最小化异步模型系统中，仍然不存在一个可以解决一致性问题的确定性算法。

FLP定理基于下面几点假设：

* 异步通信：异步通信与同步通信的最大区别是没有时钟，不能时间同步、不能使用超时、不能探测失败、消息可任意延迟、消息可乱序。
* 通信健壮：只要节点非失败，消息虽然会被无限延迟，但最终会被送达， 且消息只会被送达一次
* Fail-stop模型：节点失败如同宕机，不在处理任何消息，相对拜占庭模型，不会产生错误消息
* 失败节点数：最多只有一个进程失败或单点故障。