# CAP原则

在进行分布式软件系统设计时，有一个很重要的原则，需要设计师在设计过程中进行取舍。

CAP原则：

在一个分布式系统中，C（数据一致性），A（可用性），P（分区容错性）三者不可兼得。

举个例子：

设计一个分布式的数据存储系统，为了提高系统的可用性（防止网络，硬件等异常导致的数据不可访问），我们一般会将同一份数据存放在多个节点上，以提高某一个节点故障时的数据可用性。

但此时将面临以下问题，

1：如果我们将多个节点安排在一个分区内部，可以提高这些节点之间的数据一致性，但无法做到分布到多个分区提高容错能力。

2：如果我们将数据节点分布到多个异地分区，这样可以提高分区容错性，但在多个分区之间进行数据同步，不能做到数据的强一致性。

3：如果我们分布到多个分区，并提供数据的强一致性，那么在分区之间进行数据同步这个期间，我们没有办法做到让外部对数据系统进行访问。

因此，对任何一个分布式系统而言，在进行架构设计时，需要考虑以上三个点，并根据业务特征进行取舍。

比如说：保证数据的高可用，牺牲一定的一致性。或者确保数据的强一致性，但牺牲一定的可用性。