**对称多处理**(Symmetric Multiprocessing, SMP)不仅指计算机硬件体系结构，也指采用该体系结构的操作系统行为。使用对称多处理的操作系统可调度进程或线程到所有的处理器上运行。对称多处理体系结构与单处理器体系结构相比，具有更多的优势:

* **性能:** 若计算机要完成的工作可组织为让部分工作并行完成，则有多个处理器的系统与只有一个同类型处理器的系统相比，性能更佳。对多进程设计而言，单处理器体系结构只能一次执行一个进程，此时所有其他进程都在等待处理器。对多处理系统而言，多个进程可分别在不同的处理器上同时运行。
* **可用性:** 在对称多处理计算机中，由于所有处理器都可以执行相同的功能，因而单个处理器的失效并不会导致机器停止。相反，系统可以继续运行，只是性能有所降低。
* **增量增长:** 用户可通过添加额外处理器来增强系统的功能。
* **可扩展性:** 生产商可根据系统配置的处理器数量，提供一系列不同价格和性能特征的产品。

多线程和对称多处理是两个独立的概念。即使在单处理器系统中，多线程对结构化的应用程序和内核进程也是很有用的。由于多处理器可以并行运行多个进程，因而对称多处理计算机对非线程化的进程也是有用的。二者互补

**面向对象技术**(object-oriented design)用于给小内核增加模块化扩展。在操作系统一级，基于对象的结构可使程序员定制操作系统，而不会破坏系统的完整性。

对称多处理操作系统设计考虑因素

在SMP系统中，内核可在任何一个处理器上执行，最典型的情况是每个处理器分别从可用进程池或线程池获取任务并自行调度。内核可由多个进程或多个线程构造而成，允许各部分并行执行。由于共享资源及内核各部分同时运行引起的并发事件(如设备访问)，SMP方式使操作系统变得更复杂。

SMP操作系统管理 处理器资源 和其他计算机资源，以使用户能以与多道单处理器系统相同的方式看待SMP系统。用户可能会使用多进程或多线程的方式来构建应用，而不关心计算机使用的是单处理器还是多处理器。因此，多处理器操作系统不仅要提供多道系统的所有功能，而且必须提供适应多处理器需要的额外功能。关键设计问题为：

* **并发进程或线程:** 内核程序应可以重进不同的处理器，以使多个处理器能同时执行同一段内核代码。当多个处理器执行内核的相同或不同部分时，位避免数据损坏和无效操作，需要妥善管理内核表和数据结构。
* **调度:** 任何一个处理器都可以执行调度，这既增加了执行调度策略的复杂度，也增加了保证调度相关数据结构不被损坏的复杂度。如果使用的是内核级多线程方式，就存在将同一进程的多个线程同时调度在多个处理器上的可能性。
* **同步:** 因为可能会存在多个活跃进程访问共享地址空间或共享I/O资源的情况，因此必须认真考虑如何提供有效的同步机制这一问题。同步用来实现互斥及事件排序。在多处理器操作系统中，锁是一种通用的同步机制。
* **内存管理:** 多处理器上的内存管理不仅要处理单处理器上内存管理涉及的所有问题。另外，操作系统还要充分利用硬件提供的并行性来实现最优性能。不同处理器上的分页机制必须进行调整，以实现多处理器共享页或内存段的数据一致性，执行页面置换。物理页必须保证在重新使用前不能访问到它以前的内容。
* **可靠性和容错性:** 出现处理器故障时，操作系统应能妥善地降低故障的影响。调度器和操作系统的其他部分必须能识别出发生故障的处理器，并重新组织管理表。

对于当前的多核系统，可以从三个层次开发其潜在的并行能力。首先是每个核内部硬件并行，即指令级并行，其次是处理器层次上的潜在并行能力，即在每个处理器上多道程序或多线程程序的执行能力；最后是在多核中一个应用程序以并发多进程或多线程形式执行的潜在并行能力。