

- 实现多程序同时执行的原因
  - 资源利用率
  - 公平性
  - 便利性
- 线程的优势
  - 发挥多处理器的强大能力
  - 建模的简单性
    - 将复杂并且异步的工作流进一步分解为一组简单并且同步的工作流
    - 每个工作流在一个单独的线程中运行
    - 在特定的同步位置进行交互
  - 异步事件的简化处理
    - 非阻塞IO复杂度远远高于同步IO
  - 响应更灵敏的用户界面
- 线程的风险
  - 安全性问题
  - 活跃性问题
    - 安全性问题的含义是永远不发生糟糕的事情
    - 活跃性的目标是某件正确的事情最终会发生
    - 无意中造成的无限循环,从而使得之后的代码无法执行
  - 性能问题
    - 服务时间过长
    - 响应不灵敏
    - 吞吐率过低
    - 资源消耗过高
    - 可伸缩性较低