**Chapter 12. Other Quality Attributes[我觉得这章不重要诶……]**

了解：其它软件质量属性如可变性、可移植性、开发可分布性、伸缩性、可部署性、移动性、可监控性、生命财产安全性。其它类别的质量属性如架构质量属性、商业属性、系统质量属性。ISO/IEC FCD 25010 产品质量标准。

1. Variability(可变性):

它是指在预先计划中，系统及其artifact，为了在彼此不同的一组变体的生产的能力。

1. Portability（可移植性）：

也是一种特殊的可修改形式。指的是软件可以在一个平台轻松地更改为在另一个不同平台上运行。

1. Development Distributability（开发可分布性）:

设计软件以支持分布式软件开发的质量。

1. Scalability（可伸缩性）:水平可伸缩性（向外扩展）是指向逻辑单元添加更多资源，例如向群集添加另一台服务器。 垂直可伸缩性（向上扩展）是指向物理单元添加更多资源，例如向计算机添加更多内存。
2. Deployability（可部署性）：可执行文件如何到达主机平台以及如何调用它。
3. Mobility（移动性）：处理平台的移动和可供性问题（例如，尺寸，显示器类型，输入设备类型，可用性和带宽量以及电池寿命）。
4. Monitorability(可监控性)：处理操作人员在执行系统时监控系统的能力。
5. Safety: 软件安全是指软件能够避免进入造成或导致损害、伤害或生命损失的状态，并在确实进入坏状态时恢复和限制损害。与安全性有关的体系结构关注点几乎与可用性(即防止、检测和从故障中恢复)相同。
6. Conceptual Integrity(概念完整性):指架构设计的一致性。它有助于架构的可理解性。概念完整性要求在架构中以相同的方式完成相同的事情。
7. Marketability:一些系统是由它们的架构进行营销的，这些架构有时具有它们自己的含义，独立于它们给系统带来的其他质量属性(例如面向服务的或基于云的)。
8. Quality in Use：各利益相关方使用该系统的品质。 例如:

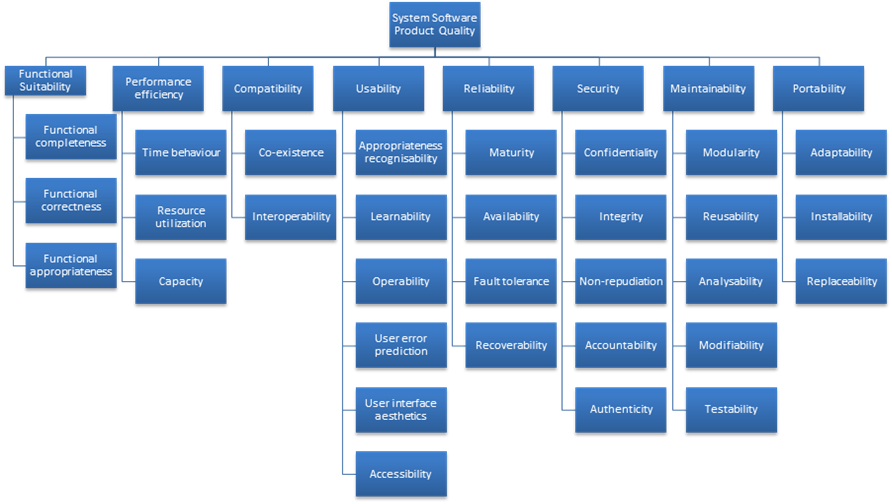
Effectiveness：衡量系统是否正确的标准

Efficiency：开发系统所需的工作量和时间

Freedom from risk：产品或系统影响经济状况，人类生命，健康或环境的程度

软件架构可对系统的质量属性产生重大影响。

Standard Lists of Quality Attributes:



List advantages: 可以提供有用的list，以帮助收集者确保不会忽视任何重要需求。可以作为创建自己的list的基础，list包含所关注的质量属性。

List disadvantages: 没有所谓完成的list。list经常产生比理解更多的争议。

list通常声称是分类法。 但什么是拒绝服务攻击？他们迫使架构师注意list中的每个质量属性。

Summary：

属性的分类可能会提供一些帮助，但它们的缺点往往超过它们的优点。

您可能需要为“新”质量属性设计或分析系统。 虽然这可能具有挑战性，但它是可行的。

理解：如何处理未知的质量属性。

假设必须处理没有紧凑知识体系的质量属性，例如： 绿色计算。

你是做什么？

三步骤：

1. 为质量属性建模
2. 为质量属性组装一套策略
3. 构建设计清单