系统工程

书上没有=。=

- 设计、实现、配置和操作一个系统，包括硬件、软件和人

目标

- 解释一个系统中的软件为什么受到系统工程众多因素的影响。
- 介绍系统总体特性的概念，如可靠性和保密性。
- 解释在系统设计过程中为什么要考虑系统的环境。
- 解释系统工程过程和系统获得过程

内容

- 系统总体特性
- 系统及其环境
- 系统建模
- 系统工程过程
- 系统采购

什么是系统?

- 一个系统是一组相互关联、能一起工作从而达到某个目标的相关组件的集合。
- 一个系统包括软件，机械、电气以及电子上的硬件，由人操作。
- 系统组件相互依赖、紧密关联。
- 系统组件的属性和行为是混合的。

系统工程的问题

- 大系统通常被设计用来解决难度很大的问题。
- 系统工程需要许多相互交叉的学科
  - 设计可以交替使用的组件的可能性几乎是无限的。
  - 工程学科之间的相互不信任和缺乏谅解。
- 系统必须被设计成在一个变化的环境中可以工作很多年。

软件和系统工程

- 系统中软件的比例在增长。软件驱动的通用的电子系统正在替代特殊目的的系统。
- 系统工程的问题与软件工程的问题相似
- 在系统工程中软件被认为是一个问题。许多大型的系统项目由于软件问题被延迟。

系统总体特性

- 系统作为一个整体的特性，而不是归结于系统各组件的特性。
- 总体特性是系统各组件之间相互关系的结果。
- 它们只有当组件集成到系统中才能被评估和测量。

总体特性的举例

- 系统的总重量
  - 可以从各个组件的特性中计算得到的总体特性的例子。
- 系统的可靠性 系统从开始使用到第一次出现故障的时间
  - 依赖于系统组件的可靠性以及组件之间的相互关联。
- 系统的可用性 系统从开始使用到第一次出现严重故障以至于无法继续使用的时间
  - 这是一个综合的特性，不光依赖于系统的硬件、软件，还依赖于系统操作人员和系统使用的环境。

可维护性：修复系统故障所需时间

总体特性的类型

- 功能特性
  - 当系统的所有部分一起工作以达到一些目标的时候表现出来。举例来说当自行车被装配起来之后就具有了运输工具的功能特性。
- 非功能特性
  - 如可靠性、性能、安全性和保密性。这些特性表现为在特定的操作环境中系统的表现行为。对以计算机为基础的系统来说，有时会要求极高，如果在某些特性达不到最低要求，系统可能就无法使用。

系统可靠性工程

- 由于组件是相互依赖的，组件失效会传播到整个系统。
- 经常由于组件间无法预料的相互关系引起系统失效。
- 无法预料所有可能的组件间相互关系。
- 软件可靠性的测量可能给出系统可靠性的错误的描述。

可靠性的影响因素

- *硬件可靠性*
  - 某个硬件组件失效的可能性有多大，修复该组件需要多长时间?
- *软件可靠性*
  - 一个软件组件产生不正确的输出的可能性有多大。软件失效与硬件失效有明显不同，因为软件不存在老化问题。
- *操作员可靠性*
  - 系统操作员出现操作失误的可能性有多大?

可靠性的关联

- 硬件失效会产生虚假的信号，使得信号超出软件所预期的范围。
- 软件错误会使警报激活，从而引起操作员紧张，导致操作错误。
- 系统安装的环境会影响它的可靠性。

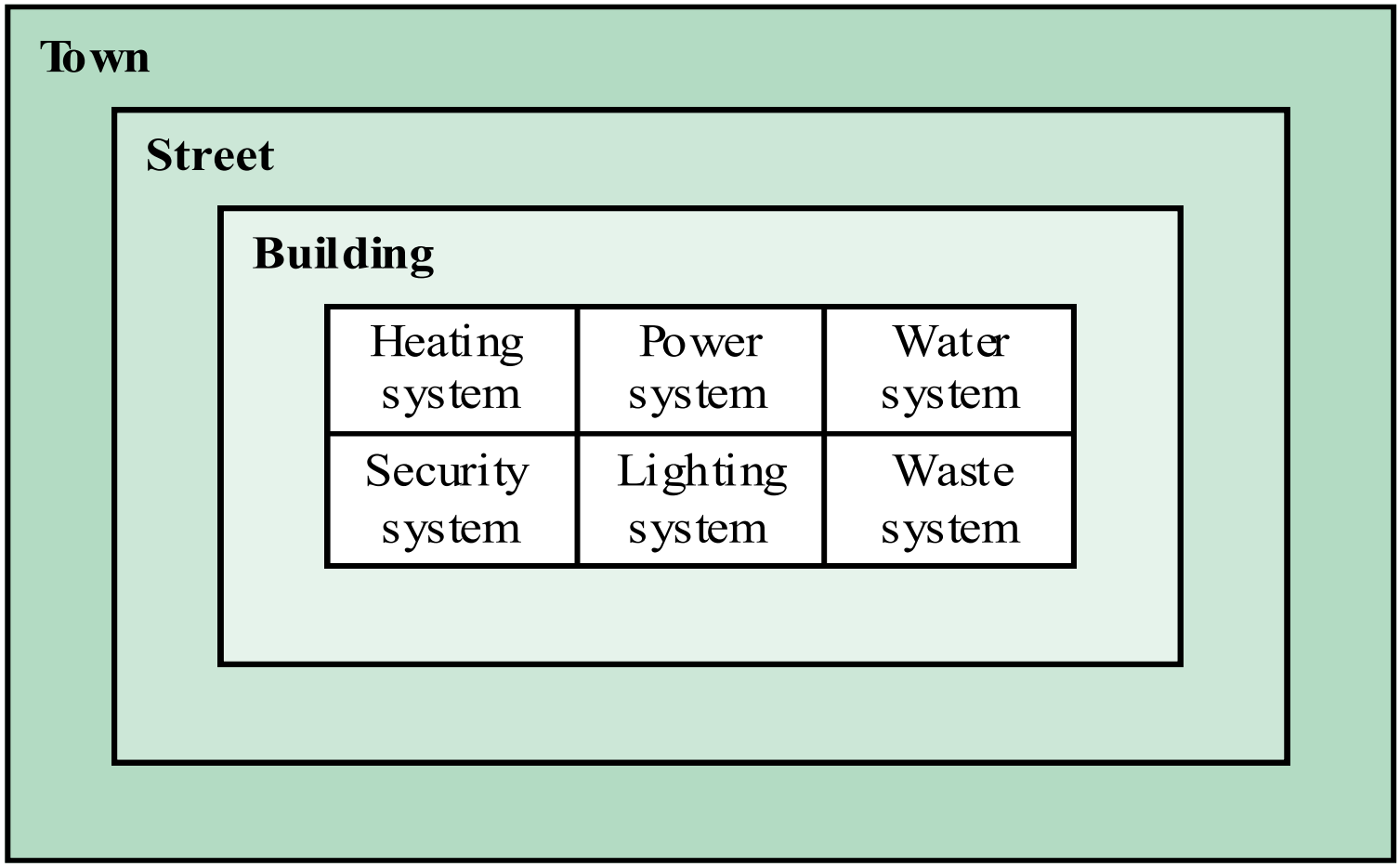
非常特性

- 像性能和可靠性这样的特性能够被度量。
- 然而，有些特性是系统不应展现出来的特性
  - 安全性－系统不应在不安全的方式下工作。
  - 保密性－系统不应允许未经授权的使用。
- 度量或评估这些特性是非常困难的。

系统及其环境

- 系统不是孤立的，是在一定的环境中存在的。
- 系统的功能可能改变其环境。
- 环境影响系统的功能。例如系统从环境供电。
- 组织的环境和物理环境一样重要。包括由政治、经济、社会和环境等因素决定的政策和流程。

系统的层级


Town
Street
Building
Heating system
Power system
Water system
Security system
Lighting system
Waste system

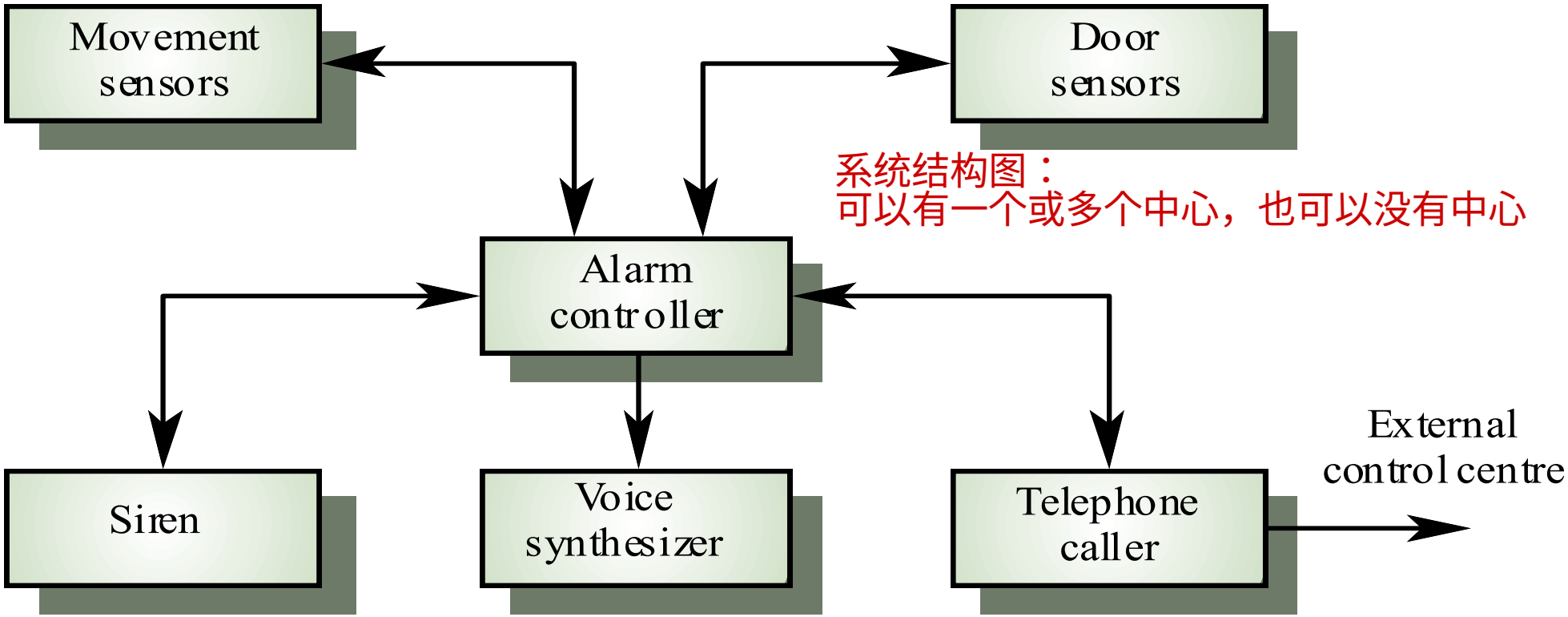
人和组织的因素

- 过程变更
  - 系统需要对环境中的工作过程作相应的变更吗?
- 工作变化
  - 系统是否使用户的技能失效或者引起用户工作方式的改变?
- 组织的变化
  - 系统是否改变了机构中的政治权利结构?

系统体系结构建模

- 体系结构模型是子系统组成系统的一个抽象的视图。
- 包括子系统间的主要的信息流。
- 通常用方块图表达
- 在模型中可以区分功能组件的不同类型

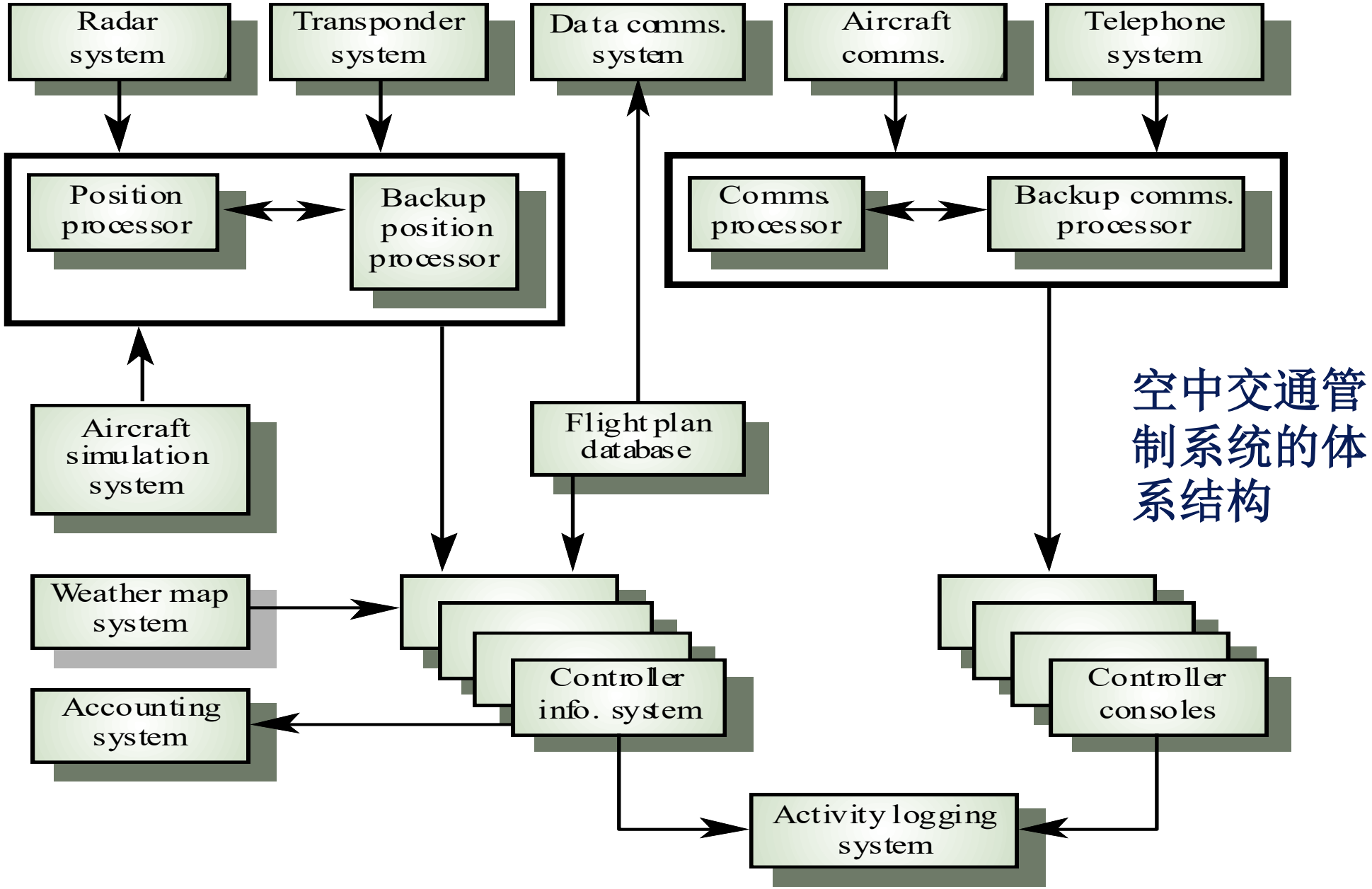
入侵者警报系统



系统结构图：
可以有一个或多个中心，也可以没有中心

警报系统中的组件类型

- 传感器
  - 运动传感器, 门传感器
- 执行机构
  - 警笛
- 通信
  - 电话呼叫器
- 调度
  - 警报控制器
- 接口
  - 语音合成器



空中交通管制系统的体系结构

系统功能组件

- 传感器组件
- 执行机构组件
- 计算组件
- 通信组件
- 调度组件
- 接口组件

系统组件

- 传感器组件
  - 从系统环境中收集信息，例如空中交通管制系统中的雷达。
- 执行机构组件
  - 引起系统环境的某些变化,例如工艺控制系统中管道中物料流速的增减。
- 计算组件
  - 针对一个输入执行计算产生输出。例如，计算机系统中的浮点运算器。

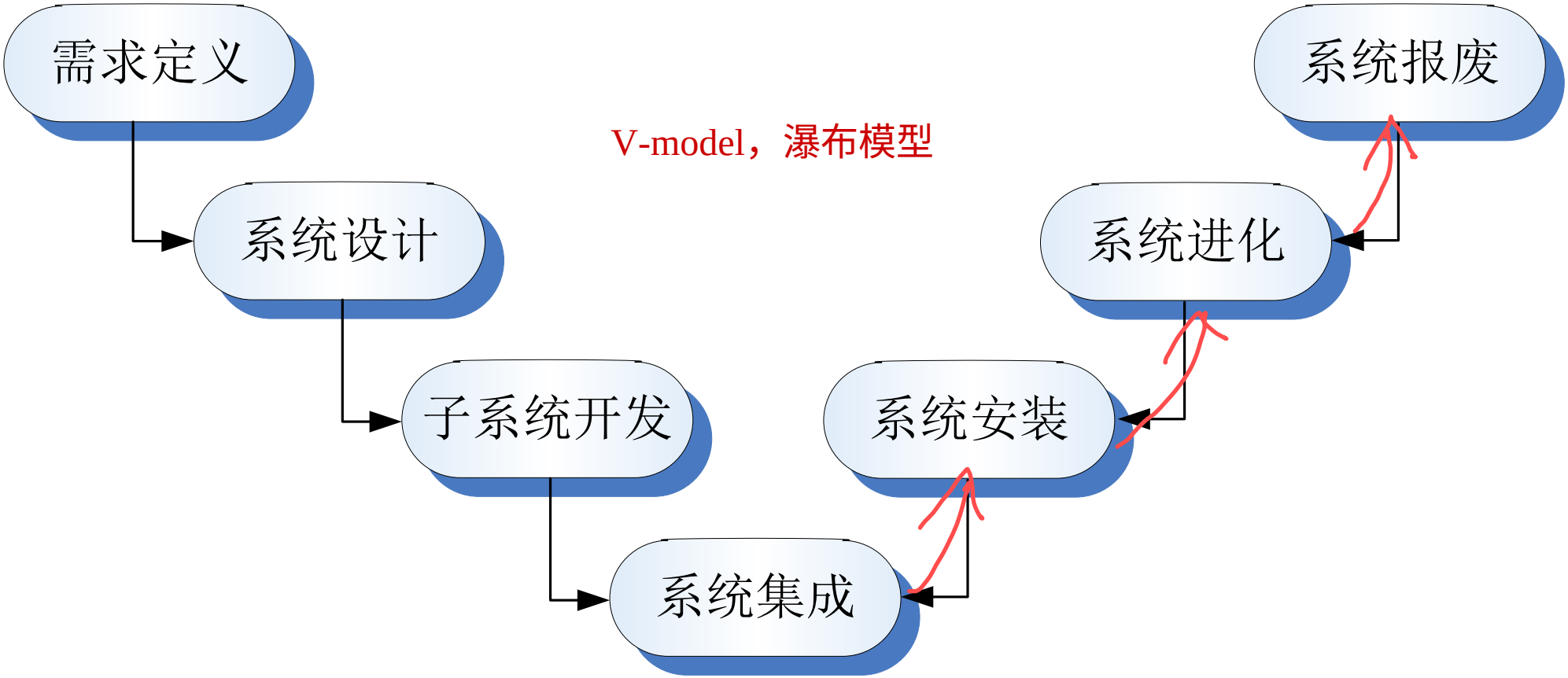
系统组件

- 通信组件
  - 允许系统组件与其它组件之间通信。例如连接分布式计算机的网络。
- 调度组件
  - 协调其它系统组件间的相互作用。例如实时系统中的调度程序。
- 接口组件
  - 便于其它系统组件的操作，例如操作员接口。
- 所有组件现在通常都是软件控制的。

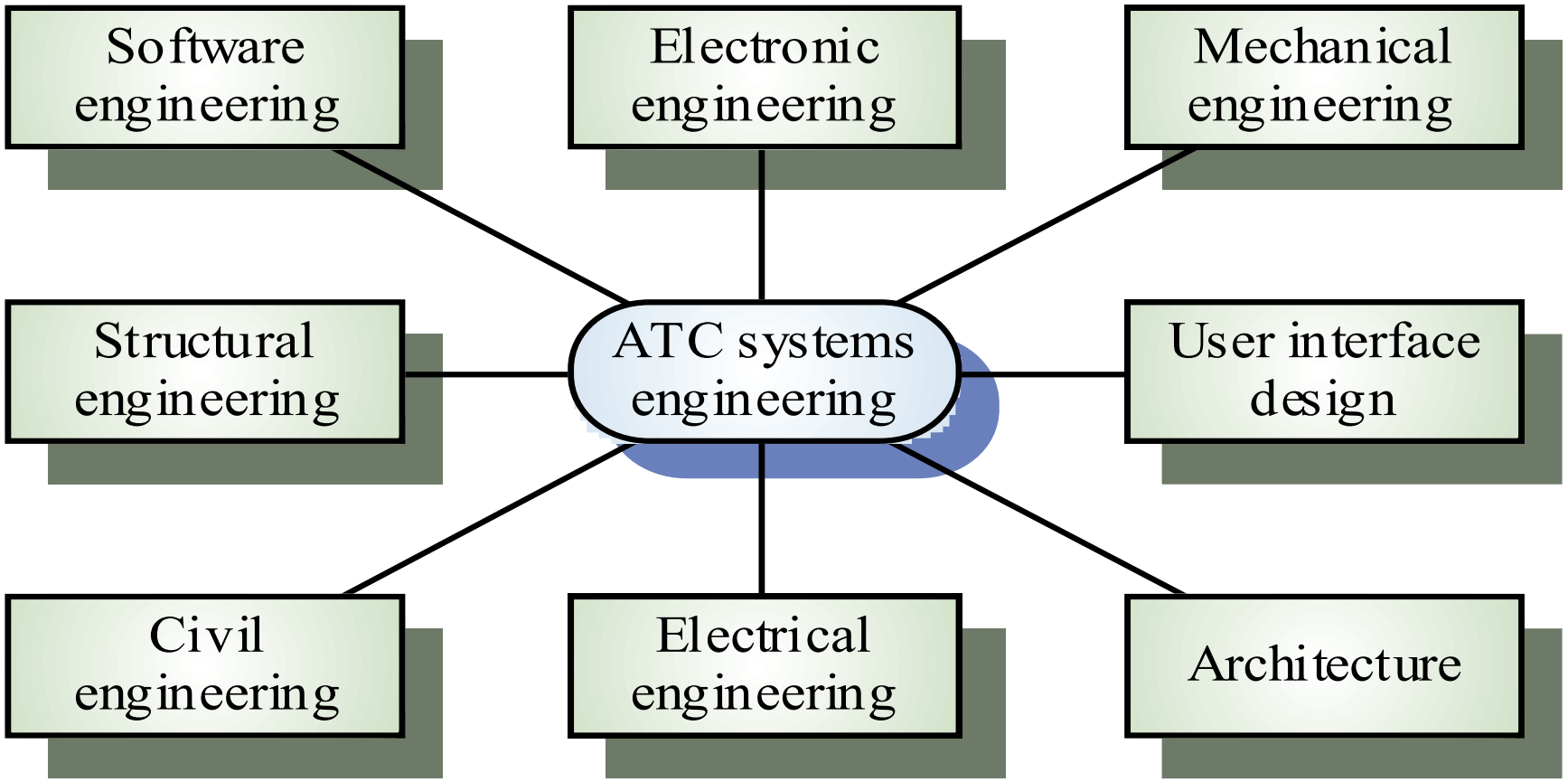
系统工程过程

- 由于需要平行开发系统的不同部分，通常遵循一个瀑布模型
  - 由于硬件的改变代价很大，所以各阶段尽量不反复。软件需要补偿硬件的问题。
- 需要不同学科的工程师一起工作
  - 容易产生误解。不同的学科使用不同的用语，需要很多沟通。

系统工程过程



学科间的关联


Software engineering
Electronic engineering
Mechanical engineering
Structural engineering
ATC systems engineering
User interface design
Civil engineering
Electrical engineering
Architecture

系统需求定义

- 在该阶段的需求定义的三种类型
  - 抽象的功能需求： 系统功能用一种抽象的方式定义
  - 系统特性： 定义系统的非功能性需求
  - 系统不应有的特征： 描述系统不可接受的行为
- 还要定义系统总体的机构的目标

不需要用系统功能的形式描述，说明为什么要对特别的环境建立该系统。

系统目标

- 功能目标
  - 大楼火灾和入侵者报警系统提供内部和外部的火灾和非法入侵报警。
- 机构的目标
  - 确保大楼中的正常工作秩序不被一些严重事件中断，这些严重事件包括火灾和非法入侵。

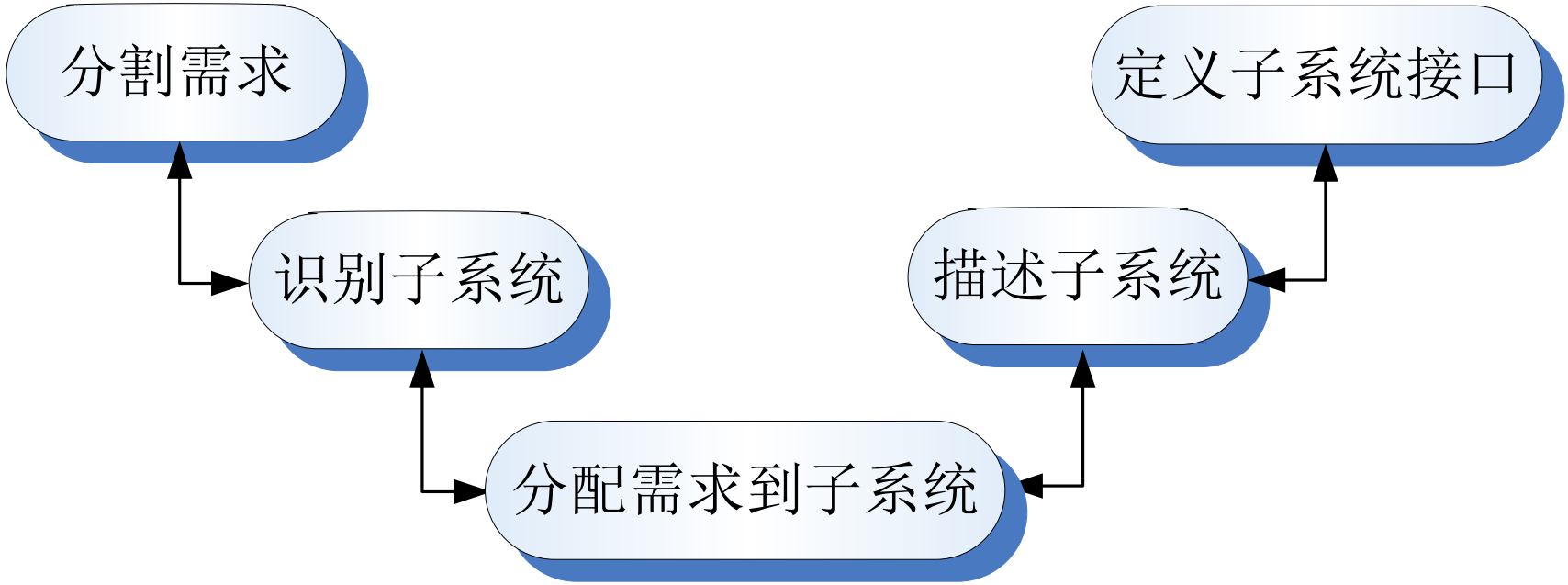
系统需求的问题

- 系统在描述的同时也在发生着变化
- 必须预见到整个系统生命周期中的硬件和通信开发
- 没有系统组件结构全貌的情况下难于定义非功能需求

系统设计过程

- 分割需求
  - 将需求归结到相关的集合
- 识别子系统
  - 识别能够分别满足一组需求的一系列子系统
- 为子系统分配需求
  - 当集成COTS（现货商品/外部购买子系统,Commercial-Off-The-Shelf）时会带来特定的问题
- 描述子系统功能
- 定义子系统接口
  - 子系统并行开发的关键

系统设计过程



系统设计的问题

- 将需求分割到硬件、软件和人的组件中可能需要商谈
- 困难的设计问题经常被假设能够用软件解决
- 硬件平台可能对软件需求来说是不适当的，所以软件必须补偿这一点

子系统开发

- 并行开发硬件、软件和通信部分
- 可能需要一些COTS系统采购
- 实现团队间缺乏交流
- 系统变更的对应缓慢，意味着由于返工需要会引起开发计划延迟

系统整合

- 将硬件、软件和人整合为一个完整系统的过程
- 为了递增地定位错误所以子系统一个一个整合
- 子系统间的接口问题通常在这个阶段被发现
- 系统中子系统的交付时间不一致的问题

系统安装

- 环境假设可能不正确
- 用户对引入新系统有抵触
- 新系统需要与一个现有系统共存
- 可能有物理上的安装问题 (如接线问题)
- 操作员培训

系统操作

- 可能带来无法预料的需求
- 系统设计者可能没有预料到用户的使用方式
- 跟其它系统交互时可能出现问题
  - 物理上不兼容
  - 数据转换问题
  - 界面不一致引起操作员错误增加

系统进化

- 大系统有很长的生命周期。它们必须进化以满足变化的需求。
- 进化是昂贵的
  - 必须从技术和业务的角度来对变更进行分析
  - 子系统相互作用所以可能引起无法预料的问题
  - 最初设计决策的理由经常没有记录
  - 变更后系统结构被破坏
- 我们把需要维护的已存在系统叫做遗留系统

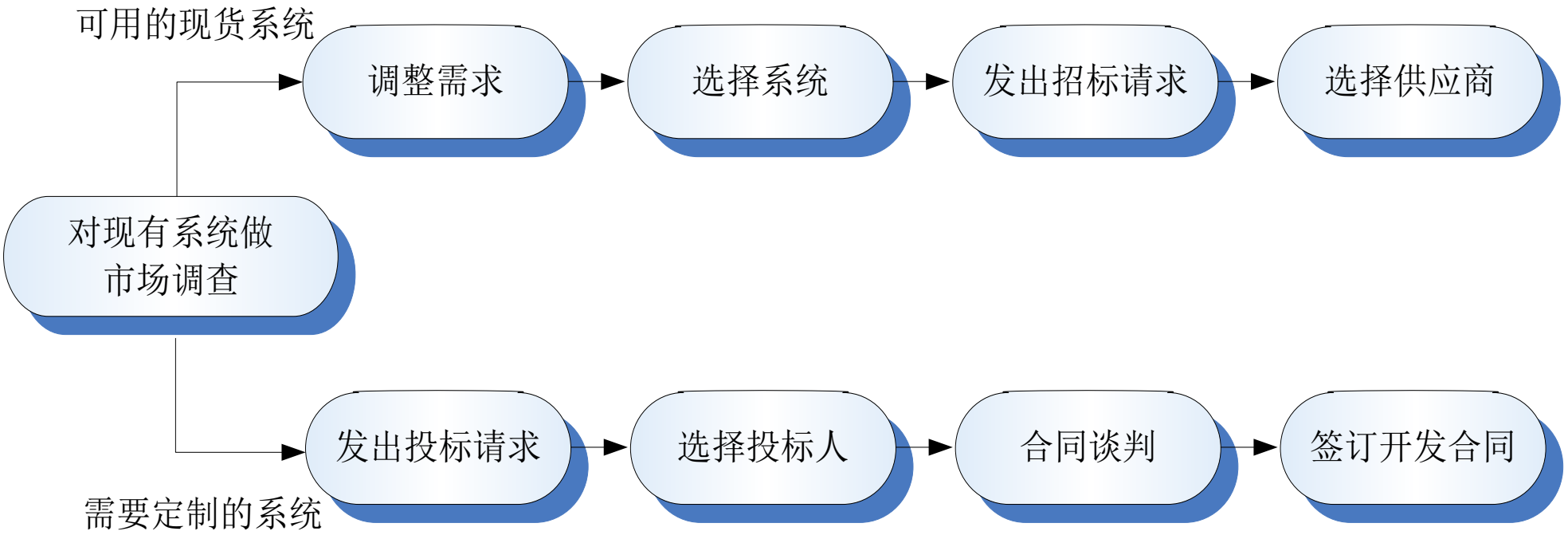
系统退役

- 系统有效生存期结束后退出服务
- 可能需要材料移除(如 危险化学品)、引起环境污染
  - 必须在系统设计阶段规划好
- 数据可能需要通过重构和转换应用到其它系统

系统采购

- 一个机构采购一个系统以满足一些需要
- 在采购前需要做系统描述和结构设计
  - 系统开发合同需要一个规格描述
  - 规格描述便于购买到COTS现货系统。 一般情况下比从零做起开发系统要低廉。

系统采购过程


可用的现货系统
调整需求
选择系统
发出招标请求
选择供应商
对现有系统做
市场调查
发出投标请求
选择投标人
合同谈判
签订开发合同
需要定制的系统

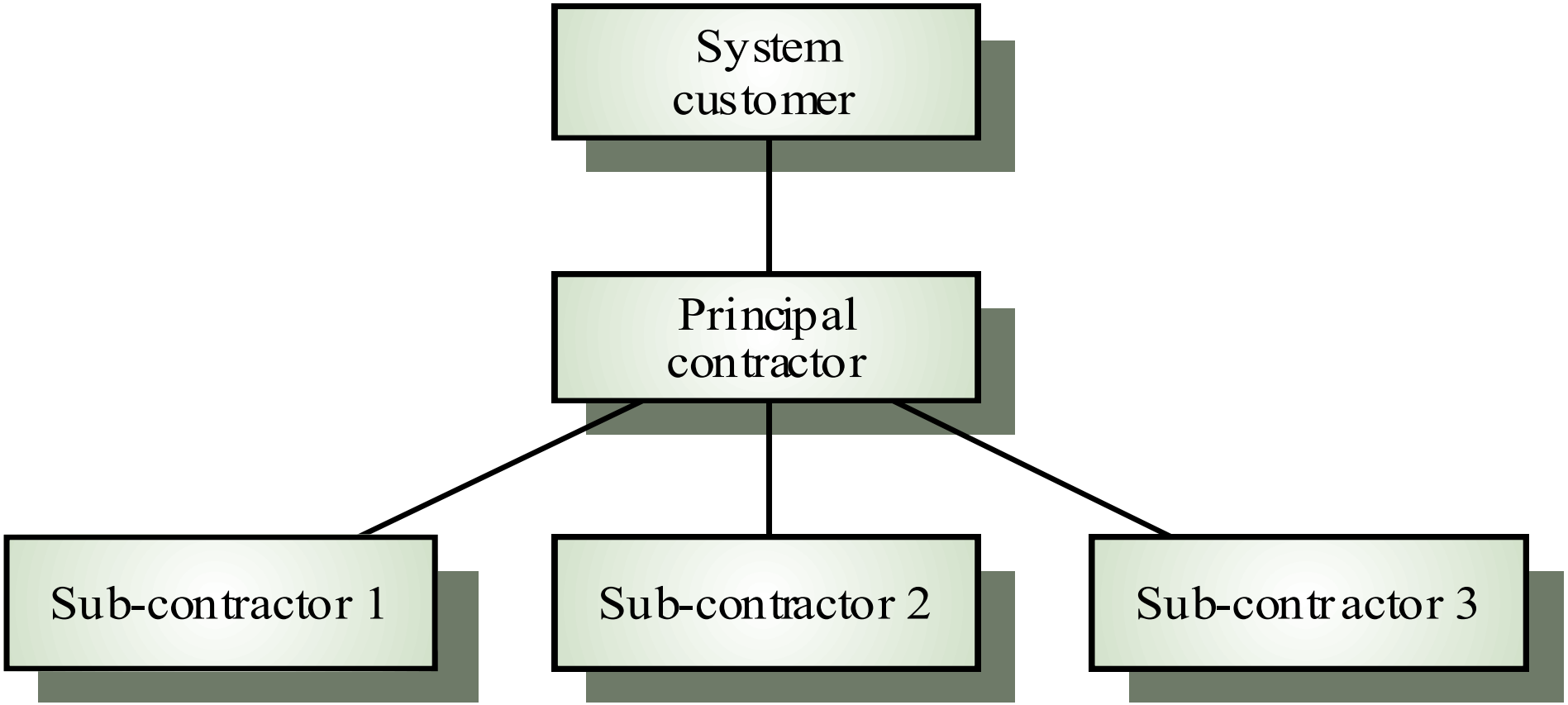
采购要点

- 需求可能不得不修改，以满足现货组件的能力
- 需求规格描述是系统开发合同的组成部分
- 系统承包商选择出来之后，通常需要一个合同谈判过程。

承包商和子承包商

- 大的硬件/软件系统的采购通常以主承包商为主。
- 转包合同发布给其它供应商以提供系统的部件
- 客户只面对主承包商，不直接涉及分包商

承包商/子承包商模型


System customer
Principal contractor
Sub-contractor 1
Sub-contractor 2
Sub-contractor 3

要点

- 系统工程涉及多个学科
- 总体特性是系统作为一个整体的特性，而不是单个组件的特性
- 系统体系结构模型描述子系统及其之间的关系。它们通常用方块图描述。

要点

- 系统组件类型有传感器、执行机构、计算组件、调度组件、通信组件和接口组件
- 系统工程过程通常是瀑布模型，包括描述、设计、开发和集成。
- 系统采购需要决定采购哪个系统、向谁采购

结论

- 系统工程是困难的！对复杂系统的开发永远不存在一个简单的答案。
- 软件工程师虽然没有所有的答案但是应该具有系统的视角
- 不同的学科需要互相认识到对方的强处、在系统工程过程中积极地配合