

# 前 言

体系与体系工程的理论和方法与系统学和系统工程一样，很多时候被冠以“无用之学”之名。这不是对这门学科或研究领域的否定，恰恰相反，这说明了体系工程相关理论方法研究的重要性。无用不是真的没有用处，而是指这些理论方法在指导具体实践时，不如现代科学一般立竿见影地解决具体问题，推动工程和技术的发展。老子《道德经》提出“故有之以利，无之以用”。体系工程这门无用之学的用处或价值主要体现在两个方面：思维价值和方法论价值。

思维(thinking)是人脑对客观事物的间接、概括的反映。思维的间接性是指思维可以到达感官所不能至的地方。概括性是指思维可以超脱事物的个别属性，认识到事物的本质和规律。本书的第 1~3 章是基础，通过讲述体系与体系工程概念、内涵、特征、表征、架构等方面的知识，使读者对“什么是体系、有哪些类型的体系、体系有什么特征、体系如何描述”等问题有深刻的认识，建立体系思维，这样在实践中遇到各类体系时，能够快速识别体系的类型、把握关系的体系特性、找到合适的体系描述手段，为体系设计分析和运用评估打下基础。这是本书的思维价值所在。

方法论(methodology)是指人们关于认识世界、改造世界的方法的理论，是一种以解决问题为目标的理论体系或系统，通常涉及对问题的阶段、任务、工具、方法的论述。思维帮助人们认识对象到底是什么，方法论则主要说明怎么办的问题。对体系而言，体系思维帮助人们从体系视角认识研究对象的体系属性，体系工程方法论则帮助人们按工程化方式来分析、规划、设计、运用、评估和治理体系。本书通过体系工程过程和体系“建用抗治”方法论来介绍体系工程生命周期模型、体系“建用抗治”方法论框架和机制原理，并通过第 5~7 章详细阐述体系需求分析、能力规划、架构设计的一系列方法技术，这是本书的方法论价值所在。

本书由张维明总体设计策划，刘俊先、罗爱民、陈涛共同完成。各个章节的内容及分工概括如下。

第 1 章为体系与体系工程概述。由张维明负责撰写。该章概述体系和体系工程的基本知识，对比体系与系统、体系工程与系统工程的异同，介绍体系工程的主要挑战。

第 2 章为体系的表征。由刘俊先负责撰写。该章介绍体系的复杂性、动态性、涌现性和智能性特征，分析对这些特征进行建模表征的挑战。

第3章为体系架构建模。由罗爱民负责撰写。该章主要概述架构的相关概念,介绍架构框架的发展及统一架构框架,重点研究分析体系架构的概念内涵并提出一种七视图的体系架构框架。

第4章为体系工程过程。由张维明、刘俊先负责撰写。该章在介绍集成体系工程过程模型并对比分析已有几种集合体系的体系工程过程模型后,面向集合体系建设运用需要,提出一种基于能力、架构中心的体系工程过程模型,并剖析其中的机制原理。

第5章为体系能力需求分析。由陈涛撰写。该章面向体系需求分析需要,聚焦作战概念、杀伤链和杀伤网的关键环节,提出体系能力需求分析的流程及产出模型。

第6章为体系规划。由刘俊先负责撰写。该章对比分析体系规划与设计、计划等相关概念的区别联系,定义体系规划模式,提出一种基于能力的体系规划过程,明确体系规划过程各环节输出的模型,介绍体系与系统协商、体系能力权衡等方面的问题及相关模型。

第7章为体系架构设计。由罗爱民、陈涛撰写。该章在明确体系设计和体系架构设计概念与范畴基础上,重点针对如何应用智能技术解决架构设计的问题,首次提出以人为主、基于计算和智能设计三种体系架构设计范式,针对体系架构智能设计介绍衍生式设计和人机共生设计方法。

第8章为体系“建用抗治”方法论。由张维明、刘俊先负责撰写。该章首先提出一种体系“建用抗治”方法论框架,结合使命工程和体系工程阐述了体系建设与运用协同的方法,概述开放性与互操作性、敏捷适应性、鲁棒性与韧性、可演进性与能控性等基本原则。

从最初计划撰写复杂体系规划理论与方法的著作并拟定撰写提纲,到提出体系“建用抗治”方法论框架,再到形成目前版本的书稿,历经7年有余,最终稿的很多内容已经偏离了原先的计划。原因是多方面的,部分原因是研究深入认识变化所带来的变更,还有就是国内外体系领域研究实践的发展引发的一些变更,更多的还是体系工程领域研究本身的困难性、挑战性所带来的不得已的变通。本书中的很多想法思路是初次探索成文,未必如其他科学技术领域般严谨,因此出版本书的目的更多的是抛砖引玉、共享交流,为推动体系工程领域研究添薪助力。

作 者

2025 年于长沙

# 目 录

## 前言

|                      |    |
|----------------------|----|
| 第 1 章 体系与体系工程概述      | 1  |
| 1.1 体系的概念与分类         | 1  |
| 1.1.1 体系的定义          | 1  |
| 1.1.2 体系的主要特征        | 2  |
| 1.1.3 体系的分类          | 4  |
| 1.2 系统和体系的比较         | 6  |
| 1.2.1 系统和体系特征的比较     | 6  |
| 1.2.2 系统和体系开发运行环境的比较 | 6  |
| 1.3 几种典型体系           | 11 |
| 1.3.1 生产制造体系         | 11 |
| 1.3.2 城市应急响应体系       | 12 |
| 1.3.3 边缘体系           | 13 |
| 1.3.4 车联网体系          | 16 |
| 1.3.5 网络信息体系         | 18 |
| 1.4 体系工程             | 20 |
| 1.4.1 体系工程的出现        | 20 |
| 1.4.2 体系工程的定义        | 21 |
| 1.4.3 体系工程领域的研究      | 22 |
| 1.4.4 系统工程与体系工程的比较   | 24 |
| 1.5 体系工程面临的挑战        | 25 |
| 1.5.1 建模与仿真          | 25 |
| 1.5.2 涌现的理解          | 26 |
| 1.5.3 体系度量和指标        | 26 |
| 1.5.4 架构模式           | 26 |
| 1.5.5 韧性、适应性和灵活性工程化  | 27 |
| 1.6 本章小结             | 27 |
| 参考文献                 | 27 |
| 第 2 章 体系的表征          | 31 |
| 2.1 体系表征概述           | 31 |

---

|       |                  |    |
|-------|------------------|----|
| 2.1.1 | 关于表征             | 31 |
| 2.1.2 | 体系表征的目的、内容及方法    | 31 |
| 2.2   | 体系的复杂性           | 32 |
| 2.2.1 | 复杂性概念及分类         | 32 |
| 2.2.2 | 复杂性的因与果          | 35 |
| 2.2.3 | 复杂性的应对           | 36 |
| 2.3   | 体系的动态性           | 38 |
| 2.3.1 | 体系动态性的本质与范围      | 38 |
| 2.3.2 | 体系主要特征与动态性的关系    | 39 |
| 2.3.3 | 体系动态性的建模挑战       | 41 |
| 2.4   | 体系的涌现性           | 41 |
| 2.4.1 | 涌现性的概念           | 41 |
| 2.4.2 | 体系涌现性的分类         | 42 |
| 2.4.3 | 体系涌现行为的建模分析方法    | 43 |
| 2.5   | 体系的智能性           | 45 |
| 2.5.1 | 体系智能性的定义         | 45 |
| 2.5.2 | 体系内部特征维度的智能性     | 47 |
| 2.5.3 | 体系智能性的表现         | 52 |
| 2.6   | 本章小结             | 53 |
|       | 参考文献             | 53 |
| 第3章   | 体系架构建模           | 54 |
| 3.1   | 体系架构的概念内涵        | 54 |
| 3.1.1 | 架构相关概念           | 54 |
| 3.1.2 | 体系架构的概念          | 55 |
| 3.1.3 | 体系架构的内涵          | 57 |
| 3.2   | 架构描述方法与架构框架      | 59 |
| 3.2.1 | 架构描述概念模型         | 59 |
| 3.2.2 | 多视图的架构描述方法       | 61 |
| 3.2.3 | 架构框架及军事领域架构框架发展  | 62 |
| 3.2.4 | 美军、英军和北约架构框架发展分析 | 67 |
| 3.3   | 统一架构框架           | 70 |
| 3.3.1 | UPDM             | 70 |
| 3.3.2 | UAF1.0           | 71 |
| 3.3.3 | UAF1.2           | 76 |
| 3.4   | 体系架构框架           | 79 |

---

|                                |            |
|--------------------------------|------------|
| 3.4.1 不同类型体系的架构框架选择 .....      | 79         |
| 3.4.2 体系架构框架视图组成与定义 .....      | 80         |
| 3.4.3 体系架构框架视图模型组成 .....       | 81         |
| 3.5 本章小结 .....                 | 84         |
| 参考文献 .....                     | 85         |
| <b>第4章 体系工程过程 .....</b>        | <b>86</b>  |
| 4.1 集成体系的体系工程过程 .....          | 86         |
| 4.2 集合体系的体系工程过程 .....          | 88         |
| 4.2.1 类 Vee 型体系工程过程模型的不足 ..... | 88         |
| 4.2.2 适应性演进体系工程过程模型 .....      | 89         |
| 4.2.3 美国 DoD 的体系工程过程模型 .....   | 90         |
| 4.2.4 WAVE 体系工程过程模型 .....      | 91         |
| 4.3 体系工程过程的产出物 .....           | 93         |
| 4.3.1 美国国防部体系工程过程的产出物 .....    | 93         |
| 4.3.2 体系能力相关产出物 .....          | 94         |
| 4.3.3 体系技术产出物 .....            | 95         |
| 4.3.4 系统相关产出物 .....            | 96         |
| 4.3.5 体系管理产出物 .....            | 96         |
| 4.3.6 体系规划产出物 .....            | 97         |
| 4.4 基于能力、架构中心的体系工程过程 .....     | 98         |
| 4.4.1 能力的概念 .....              | 98         |
| 4.4.2 基于能力、架构中心的体系工程环模型 .....  | 102        |
| 4.4.3 体系工程过程环模型的多视角分析 .....    | 106        |
| 4.5 本章小结 .....                 | 109        |
| 参考文献 .....                     | 109        |
| <b>第5章 体系能力需求分析 .....</b>      | <b>111</b> |
| 5.1 体系能力需求 .....               | 111        |
| 5.1.1 需求的概念及分类 .....           | 111        |
| 5.1.2 认识体系能力需求 .....           | 114        |
| 5.1.3 体系能力需求分析中的难点问题 .....     | 117        |
| 5.2 作战概念及其分析 .....             | 120        |
| 5.2.1 认识作战概念 .....             | 120        |
| 5.2.2 美军的马赛克战 .....            | 124        |
| 5.2.3 马赛克战的本质——决策中心战 .....     | 126        |
| 5.2.4 马赛克战的作用机理 .....          | 128        |

|                               |            |
|-------------------------------|------------|
| 5.3 端到端流程分析：杀伤链与杀伤网 .....     | 132        |
| 5.3.1 杀伤链概念的提出及发展 .....       | 133        |
| 5.3.2 基于杀伤链模型的作战问题研究 .....    | 136        |
| 5.3.3 杀伤链的发展——杀伤网 .....       | 140        |
| 5.3.4 杀伤网的构建方式 .....          | 143        |
| 5.4 体系能力需求分析过程 .....          | 146        |
| 5.4.1 提出作战概念 .....            | 147        |
| 5.4.2 设计作战概念 .....            | 149        |
| 5.4.3 研究作战场景 .....            | 155        |
| 5.4.4 分析能力需求 .....            | 161        |
| 5.5 本章小结 .....                | 166        |
| 参考文献 .....                    | 166        |
| <b>第6章 体系规划</b> .....         | <b>168</b> |
| 6.1 体系规划概述 .....              | 168        |
| 6.1.1 从“设计”到“规划+设计” .....     | 168        |
| 6.1.2 体系规划的概念与特点 .....        | 171        |
| 6.1.3 体系规划模式：面向威胁还是基于能力 ..... | 173        |
| 6.2 基于能力的规划 .....             | 176        |
| 6.2.1 相关研究进展 .....            | 176        |
| 6.2.2 基于兵棋推演的能力规划 .....       | 176        |
| 6.2.3 美国空军 ISR 能力规划与分析 .....  | 178        |
| 6.2.4 IDA 基于能力的规划 .....       | 182        |
| 6.3 基于能力的体系规划过程 .....         | 182        |
| 6.3.1 基于能力的体系规划流程 .....       | 182        |
| 6.3.2 确定能力目标 .....            | 183        |
| 6.3.3 明确能力基线 .....            | 184        |
| 6.3.4 分析能力差距 .....            | 187        |
| 6.3.5 形成能力差距解决方案 .....        | 188        |
| 6.3.6 确定能力发展演进路径 .....        | 190        |
| 6.4 体系规划的方法模型 .....           | 193        |
| 6.4.1 体系能力权衡分析 .....          | 193        |
| 6.4.2 体系与系统协商 .....           | 196        |
| 6.5 本章小结 .....                | 203        |
| 参考文献 .....                    | 203        |

|                            |     |
|----------------------------|-----|
| <b>第 7 章 体系架构设计</b>        | 205 |
| 7.1 体系设计概述                 | 205 |
| 7.1.1 体系设计的概念              | 205 |
| 7.1.2 体系设计的基本原则            | 209 |
| 7.1.3 体系设计与规划的关系           | 213 |
| 7.2 体系架构设计的过程与内容           | 214 |
| 7.2.1 体系架构设计过程             | 214 |
| 7.2.2 体系组成设计               | 214 |
| 7.2.3 组成结构设计               | 220 |
| 7.2.4 规则机制设计和质量特征设计        | 222 |
| 7.2.5 体系架构设计的范式            | 224 |
| 7.3 体系架构衍生式设计              | 229 |
| 7.3.1 衍生式设计概述              | 229 |
| 7.3.2 几种典型的衍生式设计框架         | 230 |
| 7.3.3 体系架构衍生式设计框架探索        | 237 |
| 7.4 体系的人机共生设计              | 239 |
| 7.4.1 背景                   | 239 |
| 7.4.2 人机共生设计流程             | 241 |
| 7.4.3 关键技术挑战               | 244 |
| 7.4.4 项目目标                 | 247 |
| 7.5 本章小结                   | 248 |
| 参考文献                       | 248 |
| <b>第 8 章 体系“建用抗治”方法论</b>   | 249 |
| 8.1 体系“建用抗治”方法论框架          | 249 |
| 8.1.1 体系“建用抗治”方法论的提出       | 249 |
| 8.1.2 自顶向下的体系规划设计          | 251 |
| 8.1.3 自底向上的体系组织运用          | 252 |
| 8.2 使命工程                   | 253 |
| 8.2.1 使命工程概述               | 253 |
| 8.2.2 使命工程过程               | 254 |
| 8.2.3 使命参考架构               | 258 |
| 8.3 体系建设与运用协同              | 259 |
| 8.3.1 体系工程和使命工程的协同运用模式     | 259 |
| 8.3.2 参考架构解构与能力资源封装——任务能力包 | 260 |
| 8.3.3 基于任务能力包的体系敏捷运用       | 263 |

---

|                                |     |
|--------------------------------|-----|
| 8.3.4 基于任务能力包的体系建设运用协同框架 ..... | 267 |
| 8.4 体系“建用抗治”的基本原则 .....        | 268 |
| 8.4.1 体系建设原则：开放性和互操作性 .....    | 268 |
| 8.4.2 体系运用原则：敏捷适应性 .....       | 272 |
| 8.4.3 体系对抗原则：鲁棒性和韧性 .....      | 276 |
| 8.4.4 体系治理原则：可演进性和能控性 .....    | 279 |
| 8.5 本章小结 .....                 | 281 |
| 参考文献 .....                     | 281 |

# 第 1 章 体系与体系工程概述

体系(system of systems, SoS)是怎么发展而来的? 如何构建体系? 要回答这些问题, 首先要了解体系、体系工程的基本概念, 并明晰它们与系统、系统工程的区别。

## 1.1 体系的概念与分类

### 1.1.1 体系的定义

IEEE STD 42010 将系统定义为一个被组织起来实现特定行为以完成一项任务的构件组合<sup>[1]</sup>。而体系则通常被认为是由系统构成的复杂系统。构成体系的系统称为成分系统或体系成员系统, 本书统一称为体系成员或成员系统。体系成员组合在一起构成了更大的系统, 执行单个体系成员独自完成不了的任务, 这种现象是一种涌现行为。

目前对体系并没有公认的权威定义, 以下列出了不同研究者提出的体系定义。

(1) Jamshidi<sup>[2]</sup>将体系定义为有限个体系成员的集成, 这些体系成员各自独立运行, 在一定时间内连接在一起来完成特定的高层目标。

(2) de Laurentis 认为体系是一个由异构系统组成的跨域网络集合, 这些异构系统可能表现出运行和管理的独立性、地理分布及涌现和演进行为, 在对系统及其相互作用单独建模时, 这些行为会不明显。

(3) 美国国防部在《体系的系统工程指南》<sup>[3]</sup>里将体系定义为一组独立且有用的系统构成的能提供独特能力的更大系统。

(4) 美国国土安全部认为体系由一组独立运行系统联系在一起构成, 支持应急响应人员有效地完成日常业务, 应对计划事件或重大事件<sup>[4]</sup>。

(5) 老道明大学的国家体系工程中心(National Centers for System of Systems Engineering, NCSOSE)认为体系是一类元系统(meta-system), 其具备多元功能, 这些多元功能由各类子系统协同供给<sup>[5]</sup>。

(6) Kotov<sup>[6]</sup>认为体系是大规模并发分布式系统, 其组成部分本身就是复杂系统。

(7) Kuras 和 White<sup>[7]</sup>认为体系是为实现单独运行的单个系统通常无法实现的目的而有意集成的一组系统。体系中的系统通常是单独开发的, 以实现各自的特

定目的，它们可以在与体系相关的相同环境中独立运行……并且可以在地理上分布。通常，它们的边界清晰而稳定，系统通过定义良好的接口绑定在一起。如果任何系统发生重大变化，体系通常会继续运行，但其总体能力可能会发生变化。

(8) 诺思罗普·格鲁曼公司认为体系是一个由独立自主的系统所组成的系统，这些独立自主系统作为一个整体满足特定的需求<sup>[8]</sup>。

(9) Sheard<sup>[9]</sup>认为体系是一个复杂的、有目的的整体，它由复杂的独立组成部分组成，这些组成部分的高度互操作性使它们能够组成不同的配置甚至不同的体系，其特点是上下文的复杂性会显著影响其行为并使其难以理解，具有模糊和/或不断变化的边界，并展现出涌现特性。

(10) 美国空军认为体系是由一组系统配置得到的，成员系统在使用过程中可以加入或退出；每个成员系统都有权利提供有用服务，并且管理各自的服务；它们一起展现出协同、超越的能力<sup>[10]</sup>。

(11) White<sup>[11]</sup>认为体系是一组系统，其功能是实现独立运行的单个系统通常无法实现的目的……每个系统都可以独立运行，主要是为了实现其各自的目的而进行管理。体系可以在地理上分布，并且可以表现出演进发展或涌现行为。

(12) Wikipedia<sup>[12]</sup>指出体系是一个面向任务或专用系统的集合，这些系统将其资源和能力集中在一起，以获得一个新的、更复杂的巨系统，它提供了比简单的成员系统总和更多的功能和性能。

### 1.1.2 体系的主要特征

Maier<sup>[13]</sup>认为体系有五个主要特征：

(1) 业务独立性。每一个体系成员独立运行来完成其自身的使命。

(2) 管理独立性。每个体系成员被独立管理，自行决定按被组合时不曾预期的方式进行演进。

(3) 地理分布。体系成员在地理上是分离的，成员间仅交换信息，不交换物质和能量。

(4) 演进开发。体系作为一个整体随时间而演化，以响应体系成员和业务环境的改变。

(5) 涌现行为。新的行为涌现自体系成员的交互。因为体系成员是独立演化的，因此，涌现的行为可能是短暂存在的。

Boardman 和 Sauser<sup>[14]</sup>提出了另一组区分体系与传统单一系统属性。具体如下所示<sup>[15]</sup>。

(1) 自主性(autonomy)：每个成员系统必须能够自由地实现其目标。

(2) 从属性(belonging)：每个成员系统必须能适应成为体系的一部分，并且必须相信通过变更、提供服务及与其他系统合作所带来的价值……从属性的意义在

于参与,从属性与独立性此消彼长。

(3) 连通性(connectivity):与单一系统不同,体系需要将原本封装的系统分解开,允许外界访问通常不会展现在其表面或系统边界的一些内部连通性。由于遗留组件的存在及成员系统被允许自行决定其相互连接的方式,因此,体系必须具备这一特征。

(4) 多样性(diversity):受设计限制影响,成员系统的功能相当有限,而体系出于必要应具有极其多样化的系统能力。这种多样性可以应用于各种连接系统及整体。

(5) 涌现性(emergence):各个组成部分通过相互作用而产生新的行为,使整体能力大于部分之和。自主性和多样性之间的区别在于:多样性确保了系统具有多种能力,而自主性则确保利用多样性来完成系统目标。

与 Maier 一样,Boardman 和 Sauser 也强调成员系统和涌现行为的独立性。但是,Abbott<sup>[16]</sup>提出的不同定义并未明确指出这些属性。Abbott 将体系视为一种由成员系统组成的环境,而非成员系统形成的层次结构。Abbott 指出体系存在三个关键特征。

(1) 顶层开放性:体系并非一种层次结构,且不存在“顶级”系统。新的成员系统和应用程序可能会不断增加。

(2) 底层开放性:体系无固定底层。体系只有更低层,没有最低层。

(3) 持续缓慢演化。体系永无止境,它会随着自身环境的变化而不断演化。

尽管 Abbott 并未明确指出体系的成员系统具有自主性,但是根据其定义,整个体系固有地缺乏集中管理机构,各个组成部分之间无层次之分。Abbott 认为,分级控制结构可能不利于进行有益协作。

Cocks<sup>[17]</sup>认为,根据其性质,体系具有地理分布、演化和涌现等典型特征,但是它们并非系统成为体系的关键属性。Cocks 提出了一个更简单的定义:“体系是包含一个或多个系统的系统工程过程的产物,其中不同成员系统在重要的集成和生命周期发展环节独立于整体系统的管理控制或影响。”该定义明确强调了组件的独立性,Cocks 明确指出,“重要的是系统的管理控制——而非其本身的任何内在特征”。

体系的讨论范围在某些方面确实可以重合,即使很少有研究人员使用相同的术语来表示相关概念。普遍认为,成员独立性是一个重要因素;这在很大程度上造成了体系工程的特殊复杂性。在这方面,不同研究人员的关注点不同,他们关注的重点包括成员系统的自主性、集中控制或层次结构的缺乏及成员系统超出体系影响的持续独立性(由此形成与系统目标无关的演化压力)。对体系的大多数描述或表征也提及涌现行为这一特征,并且部分学者认识到,体系通常会持续演化。

COMPASS(comprehensive modelling for advanced SoS)研究项目<sup>[18]</sup>的参与者

Nielsen 等<sup>[19]</sup>基于对体系特征文献的广泛研究,确定了体系的八个关键特性。具体如下所示。

- (1) 自主性(autonomy): 每个成员均可独立决策。
- (2) 独立性(independence): 各个成员可持续在体系之外独立运行。
- (3) 分布性(distribution): 各个成员是分散的,因此需要进行连接。
- (4) 演化性(evolution): 体系在其整个生命周期中不断发展。
- (5) 涌现行为(emergent behaviour): 整个体系提供的功能高于任何成员单独提供的功能。
- (6) 动态行为(dynamic behaviour): 可以添加新的成员或调整现有成员的相互关系。
- (7) 相互依赖性(interdependence): 体系成员之间相互依赖,并以此实现体系目标。
- (8) 互操作性(interoperability): 体系由多个成员组成,实现不同系统的连接需要对各种接口、协议和标准进行集成与调整。

### 1.1.3 体系的分类

根据单一的广泛使用的分类系统,可将体系分为四种类型。其中集成体系、协作体系、虚拟体系三种类型由 Maier<sup>[20]</sup>提出,后来 Dahmann 和 Baldwin<sup>[21]</sup>新增了一种类型,即集合体系。体系的四大类型如下所示。

(1) 集成体系(directed SoS)。为实现特定目标而建立和管理的一种体系类型。体系成员具备独立运行的能力,但是其实际运行服从体系集中管理。美国陆军曾计划开发的未来作战系统(future combat system, FCS)可认为是一个集成体系。

(2) 集合体系(acknowledged SoS)。成员仍然保持独立的目标、生命周期和资源,但是体系也建立了统一的目标、管理权限和自有资源。体系实施集中管理,但是体系成员在松散从属关系下运行(保留各自独立的所有者关系)。通过协调体系和成员之间的需求而做出变更。弹道导弹防御体系就是一个集合体系,因为这个体系中的卫星、雷达、指挥中心、拦截力量等可能属于不同的军兵种,且部分成员也担负弹道导弹防御之外的其他任务。

(3) 协作体系(collaborative SoS)。体系中没有集中管理,但是体系成员就完成统一使命自愿达成一致,在体系统一政策规定下运行。一些志愿者组织就是协作体系。

(4) 虚拟体系(virtual SoS)。体系中没有集中管理或一致统一的使命,体系成员在各自(可能共享的)政策约定下运行。供应链体系就是一类虚拟体系,这类体系中的很多物流公司并没有被集中管理,它们的使命也不完全一致,但是能够在大的商业规范中运行。

图 1.1 给出了四类体系的逻辑示意图。

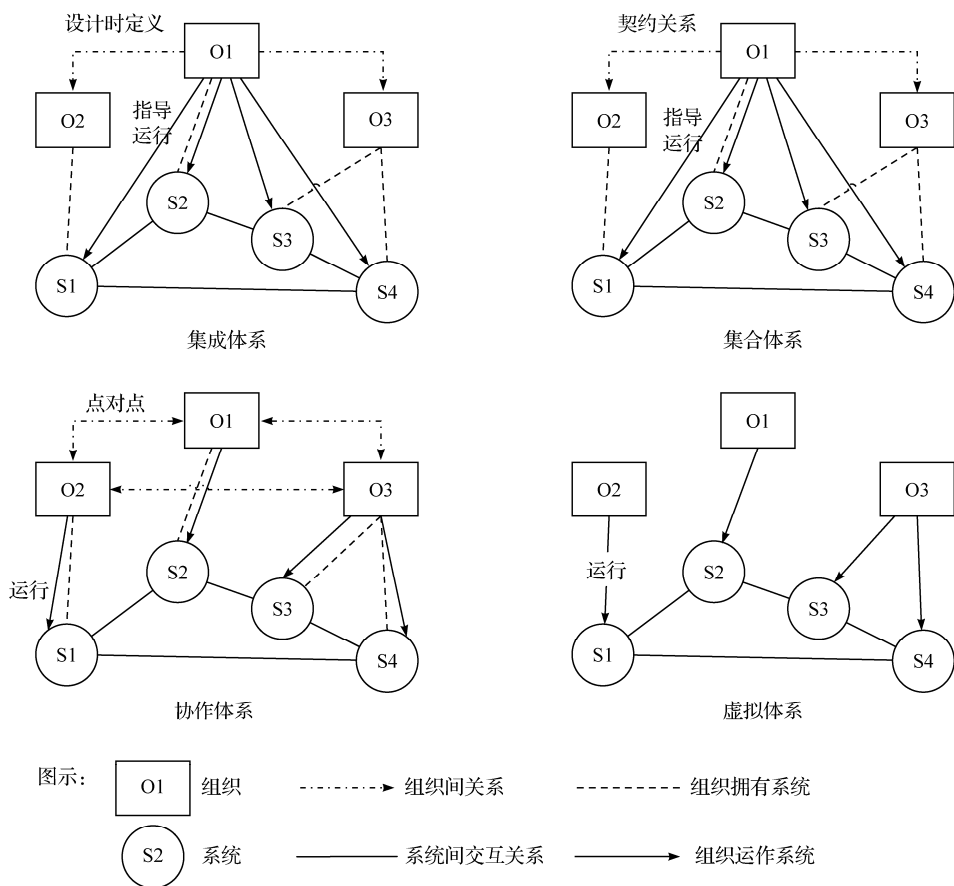


图 1.1 四类体系的逻辑示意图

在军事领域，有时也按以下三类对体系进行区分。

(1) 使命体系(mission SoS)。一起工作并提供更广泛能力的系统集合。如执行海上特定使命任务的海上编队。

(2) 平台体系(platform SoS)。装备了达成平台目标所需的独立系统(如传感器、武器、通信)的军事平台(如舰船、飞行器、卫星、地面车辆等)。如一艘航空母舰。

(3) 信息系统体系(IT-based SoS)。支持平台内、跨平台或跨系统方式运行的软硬件，为作战提供支持，以满足使命或平台目标要求。如航母编队指挥控制系统等。

使命体系、平台体系和信息系统体系可以是依次包含的关系。一个使命体系可以包含多个平台体系，而一个平台体系可以包含多个信息系统体系。

## 1.2 系统和体系的比较

### 1.2.1 系统和体系特征的比较

体系是一类特殊的系统，体系各方面的特征相对于一般系统而言有较大不同。

Thompson<sup>[22]</sup>从范围、规格、控制、演进、测试、故障、技术、涌现和开发等方面对系统和体系作了对比，结果如表 1.1 所示。

表 1.1 系统和体系特征的比较

| 特征 | 系统    | 体系  |
|----|-------|-----|
| 范围 | 确定，已知 | 未知  |
| 规格 | 确定    | 变化  |
| 控制 | 集中    | 分布  |
| 演进 | 版本控制  | 不协同 |
| 测试 | 按阶段   | 连续  |
| 故障 | 异常    | 正常  |
| 技术 | 给定，确定 | 不确定 |
| 涌现 | 可控    | 偶然  |
| 开发 | 流程模型  | 未定义 |

图 1.2 说明了系统级特征和子系统级特征，这些特征通常被纳入体系的各种定义中。系统级特征将系统描述为一个整体，而子系统级特征主要从子系统视角来描述系统。每个特征都有一个相关的尺度，按照风险的增加从左到右排序<sup>[23,24]</sup>。

任何一个给定的系统通常位于不同尺度上的不同点，将尺度转换为测量系统表现出相关特征的程度。图 1.3、图 1.4 是两个示例<sup>[25]</sup>。

### 1.2.2 系统和体系开发运行环境的比较

理解系统和体系的开发与使用环境，对于理解如何在该环境中最佳地运用体系至关重要<sup>[26]</sup>。表 1.2 列出了成员系统与集合体系之间差异的比较结果。

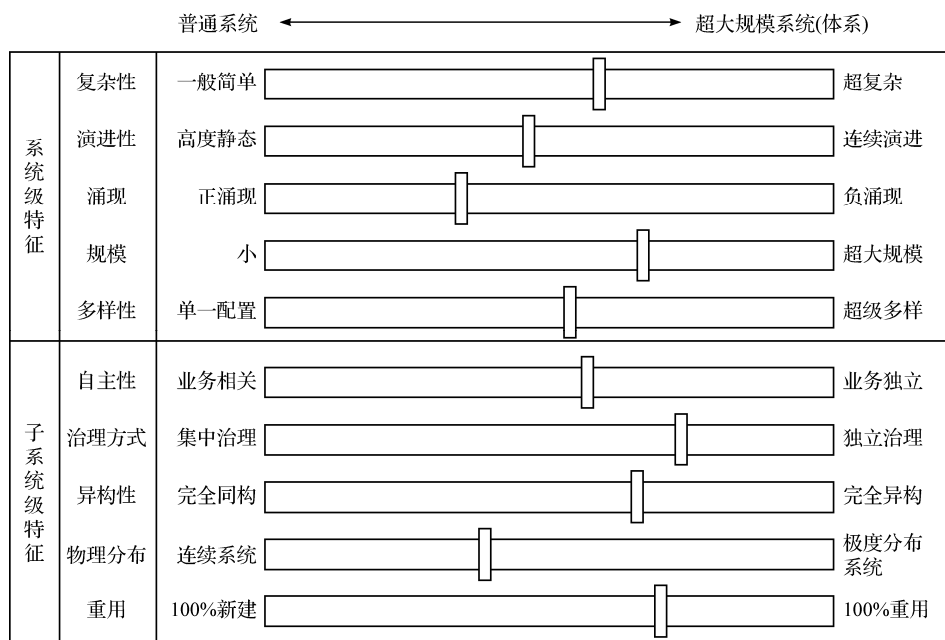


图 1.2 体系定义使用相关特征的度量

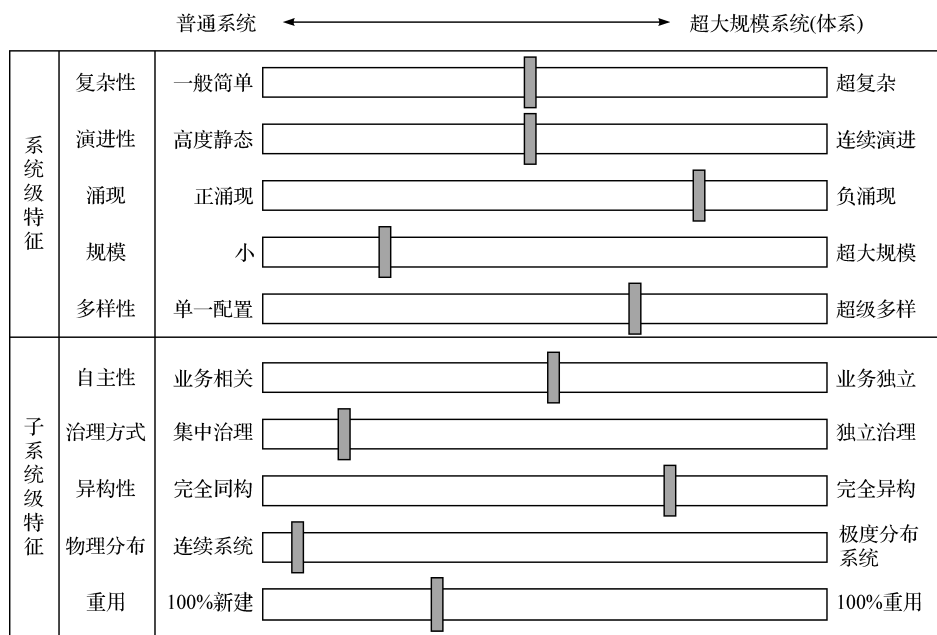


图 1.3 混合动力汽车的系统特征



利益相关者通常致力于自身，并在其系统工程中扮演特定角色。成员系统的系统工程过程的治理通常是分层的，由一个首席系统工程师(或首席工程师)负责。

对体系而言，体系和成员系统本身都有利益相关者。这些利益相关者有各自的目标和组织背景，这些目标和组织背景构成了他们对体系的期望。体系的利益相关者可能对各成员系统的约束和开发计划了解有限。在某些情况下，可能无法识别每个体系利益相关者。各成员系统的利益相关者可能对体系不感兴趣，可能给予体系需求较低的优先级，或者可能抵制其系统的体系需求。这些相互竞争的利益相关者的利益造成了体系开发运行的复杂环境。

体系治理非常复杂。它涉及一套组织机构、权力结构及分配资源、协调或控制活动所需的协作。有效的体系治理对于体系中多个独立项目和系统的工作集成至关重要。尽管体系有一名管理人员及相关资源专门服务于体系目标，但成员系统通常也有自己的管理者、赞助商、资金来源、系统工程师和独立开发项目。一些系统可能是遗留系统，不会再进行设计改造。一些系统可能参与了多个体系。因此，体系的治理必然具有协作性质。图 1.5 说明了影响体系工程师的政策和管理环境。

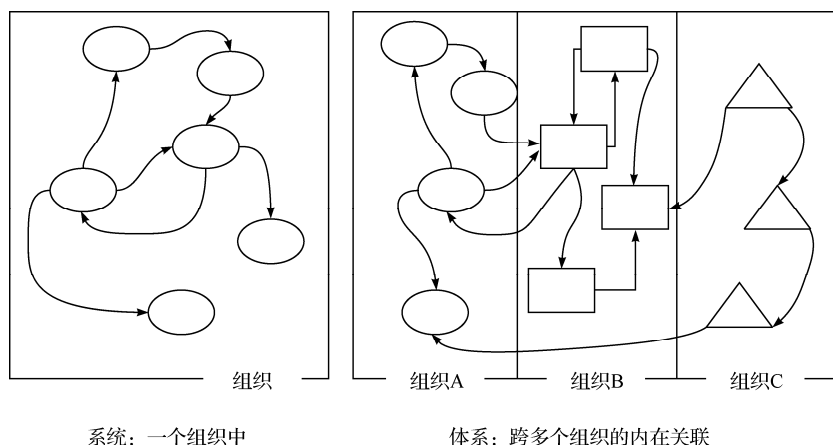


图 1.5 体系管理的挑战

体系管理人员可能无法控制所有的成员系统，成员系统利益相关者的利益也会超出体系目标，体系工程师需要在这种环境中工作。

## 2. 业务环境方面

对于业务环境中的成员系统，其任务目标是基于结构化需求或能力开发过程(包含业务概念和开发优先级)来确定的。需要保持一个明确、良好定义的运行重点，并将变更推迟到交付增量完成后实施。

对体系而言,其目的在于构建出超出成员系统可独立提供的业务能力(即涌现出“1+1>2”的能力)。这可能会对成员系统的功能或信息共享提出新的要求,而这些要求在成员系统的设计中并没有考虑。因此,在某些情况下,对成员系统的这些新需求可能与其原始目标不相称。

在根据现有系统构建新能力时,系统工程师需要考虑可能对业务用户产生直接影响的问题。各个系统之间的术语、符号、交互约定或任何其他人机界面变体的差异,都可能会对体系的可用性 & 体系使用培训造成挑战。积极地看,多个系统的综合作用可能会为业务人员提供机会,产生或启用一种最初没有计划的新能力。

总的来说,体系工程师必须对体系需求和成员系统需求进行平衡。

### 3. 体系实现方面

成员系统工程化的采办环境通常侧重于与采办类别(acquisition category, ACAT)里程碑和需求规格一致的系统生命周期。一般来说,可以对整个系统进行测试和评估,或者至少对与定义的任务和特定需求相关的子系统进行测试。

通常,体系涉及可能处于不同开发/维持阶段的多个系统,包括遗留系统、采办中的系统、寿命延长项目及与其他计划相关的系统。体系项目管理人员和工程师需要接受扩展或重新定义现有系统工程过程的挑战,以适应成员系统的独特考虑,并满足总体的体系需求。体系工程师的职责是在该过程中灌输技术纪律。体系能力的开发或演进通常不会仅由单个组织驱动,而是很可能涉及多个机构、项目管理人员及业务和保障团体。这会使体系工程师的任务复杂化,因为其必须指导成员系统的开发计划和开发优先级及其不同步的开发进度,以规划和协调体系向目标演进。除了这些开发挑战,根据成员系统的复杂性和分布,完全测试和评估体系能力可能是不可行的或非常困难的。

总的来说,体系在规划和实施时必须考虑并改进各个系统的开发计划。

### 4. 工程化和设计方面的考虑

从工程的角度来看,成员系统研制时需要重点考虑边界、接口、性能和行为等内容。系统的边界定义通常是一个静态问题,即确定系统边界内的内容(即系统)和系统边界外的内容(这是排除在系统开发项目之外的内容)。明确定义的边界允许直接识别边界点的需求,系统必须通过该边界点与不属于系统的元素交互。边界点的识别倾向于最小化系统对外部能力的依赖性,这些依赖性通过接口需求得到了很好的定义。以这种方式定义的成员系统的性能和行为往往是自主的。然而,通常存在一些外部依赖性,例如,通信或指挥控制依赖性。此外,即使是相对定义良好的系统也需要考虑范围更大的使用环境,并且可能需要预测设计变化以支持不断变化的用户需求。