



普通高等教育“十二五”规划教材
西北工业大学研究生高水平课程体系建设丛书

软件产品线工程

——原理与方法

李伟刚 李 易 编著



科学出版社

普通高等教育“十二五”规划教材
西北工业大学研究生高水平课程体系建设丛书

软件产品线工程——原理与方法

李伟刚 李 易 编著

科学出版社

北 京

内 容 简 介

软件产品线工程是有计划地进行大规模软件复用的工程方法,它能使软件产品大幅度地缩短上市时间、降低成本和提高质量。本书介绍软件产品线工程的基本原理和方法,内容包括软件产品线工程的概念、软件产品线工程的商业动机、软件产品线工程过程、软件产品线的可变性、软件产品线需求工程、软件产品线的设计和软件产品线工程的组织管理等。

本书可以作为大专院校软件工程专业或计算机软件与理论专业高年级本科生和研究生的教材,也可以作为软件工程专业人员的参考书。

图书在版编目(CIP)数据

软件产品线工程:原理与方法/李伟刚,李易编著. —北京:科学出版社, 2015.6

(西北工业大学研究生高水平课程体系建设丛书)

普通高等教育“十二五”规划教材

ISBN 9 78-7-03-044758-8

I. ①软… II. ①李… ②李… III. ①软件开发-高等学校-教材
IV. ①TP311.52

中国版本图书馆 CIP 数据核字 (2015) 第 124339 号

责任编辑:李 萍 杨向萍 纪四穗/责任校对:刘亚琦

责任印制:赵 博/封面设计:红叶图文

科 学 出 版 社 出 版

北京东黄城根北街 16 号

邮政编码:100717

http://www.sciencep.com

文林印务有限公司印刷

科学出版社发行 各地新华书店经销

*

2015 年 6 月第 一 版 开本:720 × 1000 1/16

2015 年 6 月第一次印刷 印张:15

字数:302 000

定价:50.00 元

(如有印装质量问题,我社负责调换)

前 言

软件产品线是拥有公共特征的一组软件密集型系统的集合，这些公共特征满足特定市场和特定任务的需求，在一组公共核心资产的基础上按照预定的方式开发而来。软件产品线正在发展并成为一种可行的、重要的软件开发范型，它能使软件企业在产品上市时间、成本、生产率、质量和其他方面取得前所未有的进步。软件产品线工程还能实现快速切入市场、灵活反应并具备大规模定制的能力。

软件产品线能够帮助软件企业克服资源短缺的问题，无论企业规模如何，都可以施行软件产品线战略，用来提升它们的竞争力。按照卡耐基·梅隆大学软件工程研究所（SEI）的统计^[1]，引入软件产品线工程，可以使得软件企业获得如下进步：

- （1）将生产率提升至 10 倍以上。
- （2）产品质量提高 10 倍以上。
- （3）降低 60% 的成本。
- （4）对人工的需求降低 87%。
- （5）产品上市时间最大可缩短 98%。
- （6）在数月内即可进入新市场，而不是以前的数年。

虽然不知道这些数字是如何统计出来的，但是软件产品线工程是有计划地进行大规模软件复用的助推器是毋庸置疑的。近年来，软件产品线工程在国际软件产业界和学术界得到了高度的重视、深入的研究和广泛的应用，许多软件密集型企业和组织（如飞利浦、惠普、美国海军水下作战中心、美国国家侦察办公室等）都在积极实践这一工程方法，并取得了巨大的成功。

在国内，软件企业面临异常激烈的竞争。一方面，受到国际大型或专业化软件公司挤压；另一方面，国内同业竞争甚至恶性竞争依然存在。如何实现软件产业技术升级，改变手工作坊式的或欠缺严密规划的软件生产模式，提高软件产品应对细分市场的能力，改善软件复用程度，从而降低研发成本、提高产品质量、缩短产品响应市场的时间成为一道难题。对于软件从业者，学习软件产品线工程方法，对解决这个难题大有裨益。

鉴于国内高校缺乏软件产品线工程方面的教材和指导书的现状，我们在长期从事软件工程方法研究和工程实施所积累的成果和经验的基础上，编写了这本教材。全书分成 8 章，涵盖了软件产品线工程的基本原理和主要方法，每章后附思

考与练习题，方便读者检查学习效果。各章的内容如下。

第1章阐述软件产品线工程的背景、发展历史及核心概念。

第2章讨论软件产品线工程带来的经济效应以及它对软件企业战略及产品管理的影响。介绍各种不同类型的产品线市场，并给出一种产品线经济模型。这一章还为厂商决策导入软件产品线提供若干建议。

第3章介绍软件产品线工程中的三大核心活动和两大开发过程，即开发核心资产活动、开发产品活动和管理活动，以及领域工程过程和应用工程过程。三大核心活动体现了软件产品线工程的关注点从单个产品向整条产品线的转移，而两大开发过程则将开发可复用工件的过程与使用可复用工件开发软件个体的过程明确分离，从而产生了领域工程和应用工程两个特殊的工程领域。本章最后还讨论软件产品线工程方法如何应用于标准软件产品开发和基于项目的软件开发。

第4章主要讨论可变性的基本概念及其在软件产品线工程中的作用，并介绍如何进行可变性建模，以及如何利用可变性模型在应用工程中进行产品开发。描述可变性并对其进行管理是软件产品线工程的核心，本章使用正交可变性模型为软件产品线工程中所有工件的可变性提供统一的描述方法，还介绍多种视图下识别可变性的方法。

第5章论述软件产品线的需求工程。首先介绍传统需求工程的概况，而后分析软件产品线需求工程的特点及其方法和原则。接着，通过用例分析及建模介绍实施产品线需求工程的主要方法，着重讨论如何在用例中对可变性建模。最后介绍特征建模技术及其在软件产品线可变性建模中的作用，并对特征模型与用例模型之间的映射关系进行总结。

第6章详细讨论软件产品线设计方面的问题。先介绍架构设计的一般方法，然后以 UML 为工具，说明面向对象分析与设计方法在软件产品线中的应用，主要包括静态分析与设计、动态分析与设计和可变性设计。

第7章描述在应用工程中如何通过调整和配置领域工程的输出工件来生成一个特定的软件应用。在软件产品线特征模型上进行裁剪，得到所需软件的功能架构，然后根据所选特征对用例模型、分析模型和设计模型进行修改，得到软件的各种分析和设计模型，实现领域工程向应用工程的过渡。

第8章讨论软件产品线工程的组织方面的问题。首先讨论设立一个好的组织对于软件产品线工程取得成功的重要性，然后说明产品线中的特定角色及其职责，接着讨论如何建立一个与这些角色相适应的组织结构，最后对如何保证组织中沟通的顺畅进行探讨。

在编写本书时，参考和借鉴了学术界与工业界有关学者、专家的最新研究成果，特别是本书所采用的微波炉软件产品线的案例，是在 Gomaa 设计的例子基础上改编的，在此向他们表示诚挚的谢意。西北工业大学研究生院和软件与

微电子学院对本书的编写给予了关注，并提供资金支持和课程改革方面的便利条件。科学出版社的李萍编辑对本书的出版倾注了大量的心血，这里一并表示感谢。

由于作者水平有限，经验不足，书中不足之处在所难免，恳请读者批评指正。谢谢！

李伟刚

2015年6月

目 录

前言

1 软件产品线工程基础	1
1.1 产品线工程	2
1.1.1 来自制造业的案例	2
1.1.2 基于平台定制产品	3
1.1.3 定制产品的工程化	5
1.1.4 产品线工程的益处	7
1.2 软件产品线工程	8
1.2.1 发展历史	9
1.2.2 公共性和可变性	11
1.2.3 领域工程和应用工程	12
1.2.4 组织、过程和技术	14
思考与练习	16
2 软件产品线的导入	17
2.1 软件产品线工程的商业目标	17
2.2 软件产品线市场	18
2.2.1 软件产品定义策略	19
2.2.2 市场战略	19
2.2.3 软件产品线生存周期	20
2.2.4 产品线市场战略与产品线工程的关系	22
2.3 软件产品线的经济学	22
2.3.1 软件产品线工程带来的经济效应	22
2.3.2 简单的软件产品线经济模型	24
2.3.3 软件产品线的其他经济属性	25
2.4 产品的管理和范围界定	26
2.4.1 产品组合管理	26
2.4.2 领域潜能分析	27
2.4.3 资产范围界定	29
2.5 小结	30

思考与练习	30
3 软件产品线工程过程框架	31
3.1 软件产品线工程中的基本活动	31
3.1.1 三大基本活动	31
3.1.2 开发核心资产	33
3.1.3 开发产品	37
3.1.4 管理	38
3.1.5 整合三大基本活动	39
3.2 统一软件开发过程	40
3.3 软件产品线工程过程	42
3.3.1 为复用而开发	42
3.3.2 两个开发过程	44
3.3.3 过程框架	45
3.4 领域工程	48
3.4.1 核心 workflow	49
3.4.2 主要阶段	52
3.5 应用工程	53
3.5.1 核心 workflow	54
3.5.2 主要阶段	56
3.6 产品驱动和方案驱动的软件产品线工程模式	57
3.6.1 两种软件产品线类型	57
3.6.2 产品驱动的软件产品线工程	58
3.6.3 方案驱动的软件产品线工程	59
3.7 小结	60
思考与练习	61
4 可变性	62
4.1 基本概念	62
4.1.1 可变性是什么	62
4.1.2 软件产品线工程中的可变性	64
4.1.3 可变性的分类	66
4.2 可变性元建模	70
4.2.1 解析可变性	71
4.2.2 可变性元模型	72
4.2.3 正交可变性	80
4.3 多种视图下的可变性	82

4.3.1 C-A-F-C-R 视图	82
4.3.2 多视图下的可变性	89
4.4 应用工程与可变性	95
4.5 小结	96
思考与练习	96
5 从用例分析到特征模型	97
5.1 需求工程	97
5.1.1 概述	97
5.1.2 软件产品线中的需求工程	102
5.2 在领域工程中进行需求分析	106
5.2.1 领域工程中需求分析的主要工作内容	107
5.2.2 公共性需求	107
5.2.3 可变性需求	109
5.3 软件产品线的用例建模	111
5.3.1 用例模型	111
5.3.2 软件产品线中的用例模型	113
5.3.3 在用例中对可变性建模	116
5.3.4 对用例开发的进一步探讨	130
5.4 软件产品线的特征建模	132
5.4.1 有关特征模型和正交可变性模型的讨论	132
5.4.2 特征建模技术	132
5.4.3 扩展特征建模技术	134
5.4.4 用 UML 进行特征建模	139
5.5 小结	149
思考与练习	149
6 软件产品线设计	151
6.1 架构设计	151
6.1.1 架构设计的作用	151
6.1.2 架构设计的范围	152
6.1.3 软件产品线架构设计	153
6.1.4 架构评估	155
6.1.5 架构演化	155
6.2 基于 UML 的软件产品线设计	157
6.2.1 软件产品线设计中的静态建模技术	157
6.2.2 软件产品线动态建模: 交互图法	168

6.2.3 软件产品线动态建模：有限状态机法	180
6.2.4 特征与类的耦合	193
6.3 小结	200
思考与练习	201
7 应用工程	202
7.1 应用工程的开发策略	202
7.2 应用工程过程实践	203
7.2.1 特征模型	203
7.2.2 用例模型	204
7.2.3 静态模型	205
7.2.4 动态模型	205
7.2.5 特征-类依赖关系和类结构	207
7.3 小结	208
思考与练习	208
8 组织	209
8.1 组织在软件产品线工程中的作用	209
8.2 角色和职责	210
8.2.1 产品经理	211
8.2.2 领域需求工程师	211
8.2.3 领域架构师	211
8.2.4 领域开发工程师	212
8.2.5 领域测试工程师	212
8.2.6 领域资产管理	212
8.2.7 应用需求工程师	212
8.2.8 应用架构师	213
8.2.9 应用开发工程师	213
8.2.10 应用测试工程师	213
8.3 组织结构	213
8.3.1 面向产品的组织	214
8.3.2 面向过程的组织	216
8.3.3 矩阵型组织	217
8.3.4 测试	218
8.3.5 资产管理	219
8.3.6 产品管理	220
8.4 地域分布	222

8.5 协作模式	222
8.6 小结	223
思考与练习	223
参考文献	224

1 软件产品线工程基础

随着软件系统的规模和复杂度的增加，在软件工程中人们试图通过提高软件资产复用程度来达到降低开发成本、提高开发效率、缩短软件上市时间的目的。因此，软件复用相关的方法和技术得到了空前的关注，成为学术界和产业界研究与应用的热点。软件的模块化、抽象和组合机制等思想和技术层出不穷。软件复用意味着只有在软件资产具备足够的可变性时，才可能被应用于不同的产品中。因而，支持可变性^[2]成为现代软件开发实践的重要特征，软件产品线（software product line, SPL）^[3]机制以及产生式编程（generative programming^①）^[4]等方法应运而生。这些机制与从零开始编码，创建独一无二的软件不同，在一定程度上实现了自动化的软件开发。通过引入可变性技术^[5]，可创建出高可复用的代码库和构件，从而达到降低开发成本和缩短上市时间的目标。

软件产品线是一个软件系统的家族，它们共享一组公共的和可管理的特性集，来满足特定市场或特定项目的个性化需求；它们基于一组公共的核心软件资产，用配置的方式开发出来，核心资产在这个产品家族中得到最大程度的复用。核心资产可以是文档、模型、代码和算法等，组成了产品的路线图、需求、项目计划、系统架构、设计模型和软件构件等工件（工件是指在软件过程中被产生、修改和使用的信息片段，通常以某种信息载体的形式呈现，如文档和代码等^[6,7]）。

软件产品线提供了在一套产品中管理可变性的系统化方法，主要的优点如下。

（1）可以标识产品线家族中的个体（称为变种），描述各个变种之间的关系，追溯变种与核心资产之间的关联。

（2）充分考虑了对产品家族的未来扩展。

（3）既达到了产品实现的灵活性，又保持了模块化。

（4）能够管理产品家族中多产品之间的演化，保证其一一致性。

在一个特定领域，需要建立的应用软件越具有相似性，越应该考虑使用软件产品线方法重用以前的开发成果。然而，软件产品线方法的应用仍有一些障碍，表现如下。

（1）规模的限制：大规模产品线环境中存在大量的产品变种，它们之间形成指数级数量的关联及依赖关系。从需求工件到实现工件可变性模型逐步细化，所以当追溯变种之间的交互时，会非常困难。

① 一种通过泛型、类、原型、模板、方面和代码生成器等自动生成源代码，以提高软件生产率的软件编程范式。

(2) 产品中引入可变性会大大增加整个软件产品线架构设计的难度。

(3) 软件产品线经常用于不同的业务环境中, 每种环境都具有独特的复杂性, 设计统一的产品线模型非常困难。

本书将系统地阐述软件产品线工程的基础理论, 介绍该领域的最新研究成果, 并举例说明软件产品线工程相关的工具及其实际应用。

1.1 产品线工程

在工业界, 产品线是指一群相关的产品, 这些产品可能功能相似, 能满足同类需求而规格、型号和花色等不同, 一般经过相同或相似的销售途径销售给同一顾客群。

1.1.1 来自制造业的案例

制造业是率先引入产品线技术来降低产品研发成本、缩短产品上市时间, 同时又满足用户多样化需求的。然而, 在 20 世纪初期, 由于技术和管理水平的限制, 工业产品尚不丰富, 企业还无法为用户提供足够多的产品选型。美国福特汽车公司生产的 T 型车就是典型的案例^[8]。

福特汽车公司成立于 1903 年, 当时的汽车工厂都是处于作坊式的手工生产状态。这种生产方式使得汽车的产量很低, 成本居高不下。当时, 一辆汽车在美国的售价大约是 4700 美元, 相当于一个普通工人好几年的收入。福特公司创始人亨利·福特认为, 要想把汽车市场变成一个能够创造巨大利润的市场, 就必须把汽车变成普通人也买得起的消费品, 而要想做到这一点, 就得大幅降低价格。于是, 1908 年, 福特汽车公司推出了 T 型车, 它广泛使用机器替代工人的手工劳作。当时, 其他公司装配出一辆汽车需要 700 多小时, 而福特仅仅需要 12.5 小时。这使得福特公司最初推向市场的 T 型车定价只有 825 美元, 远远低于市场价格水平。

为了进一步提高生产率, 1910 年, 在已经生产了 12000 辆 T 型车之后, 亨利·福特将工厂从底特律的皮科特搬迁至海兰公园, 并在那里首创了 T 型车生产流水线。这样, 单辆汽车的装配时间进一步缩短至 93 分钟。随着流水线的不断改进, 十几年后, 福特公司的生产效率提高到了惊人的程度: 10 秒钟就可以生产出一辆汽车。与此同时, 福特汽车的市场价格不断下降, 1910 年降为 780 美元, 1911 年下降到 690 美元, 1914 年则大幅降到了 360 美元, 并且最终降到了 260 美元。

福特公司先进的生产方式为它带来了极大的市场优势。1908 年, T 型车的产量达到 10660 辆, 创下了汽车行业的纪录。到了 1921 年, T 型车的产量已占世界汽车总产量的 56.6%, T 型车的最终产量超过了 1500 万辆, 福特公司也成了美国最大的汽车公司。可以说, 福特创造出了现代工业史上的奇迹。

福特的 T 型车开创了汽车工业的革命, 使得人们能够很容易地购买到一辆汽

车。随着基本需求的满足,人们逐渐对汽车多样性的要求越来越迫切。虽然福特公司把单一型号大规模生产的潜力发挥到了极致,但是当市场转变成了买方市场后,简陋而千篇一律的T型车虽然价廉,但已经不能满足消费者的需求。面对福特汽车的价格优势,竞争对手通用汽车公司转而在汽车的舒适化、个性化和多样化等方面大做文章,以此来对抗廉价的福特汽车,推出了新款雪佛兰汽车。雪佛兰一上市就大受欢迎,严重冲击了福特T型车的市场份额。

然而,面对市场的变化,福特仍然顽固地坚持生产中心的观念,不相信还有比单一品种、大批量、精密分工和流水线生产更经济、更有效的生产方式,甚至不愿意生产除黑色以外的其他颜色的汽车。福特宣称:“无论你需要什么颜色的汽车,福特只有黑色的。”

无视用户核心需求的变化,不在产品可变性上下工夫的福特T型车,最终在1927年宣告停产。之后,福特公司面临着产品转型的问题。但是,过去几乎长达30年的时间福特一直只生产这一种型号的汽车,产品转型异常艰难:所有的设备和工艺都只能用于生产T型车,想要转产其他车型,就要在全面停产的条件下,花费大量的资金和时间全面更新这些设备和工艺。1927年开始,福特公司被迫停产,重组生产线,更换1.5万台车床,重新设计制造2.5万台机床。这些庞大的调整工作耗用了1亿美元的资金和16个月的时间。等到新车型投产时,福特公司已经从全美第一大汽车公司降至第二位。新车型仓促上市,许多部件的技术并不成熟,加之随后被迫更换发动机,福特不得不再一次停产。通用汽车公司等竞争对手趁机抢占市场。1933年,福特的新车重新上市。这时,福特公司落在了通用汽车公司和克莱斯勒汽车公司之后,成为美国第三大汽车公司。直到今天,福特公司也没有能够恢复昔日美国最大汽车公司的地位。

这个案例带给我们许多启示,仅从产品生产的角度分析福特公司所犯的错误,主要包括如下几个方面。

- (1) 没有及时把握用户需求的变化,并按新需求改善产品战略。
- (2) 产品型号单一,在设计中没有引入任何可能形成产品变种的机制。
- (3) 生产方式固定,生产工艺和生产工具无法适应形成产品变种的需求。
- (4) 同样,产品质量控制方法只能适应T型车一种产品型号的生产需求。

由此可见,现代流水线生产方式,与面向单用户的手工生产方式相比,在大幅提高生产效率和降低成本的同时,却降低了生产多样化产品的可能性。

工业生产在面临个性化需求时,能否达到优质的服务与高效生产和较低成本的平衡呢?人们逐渐发明了大规模定制的生产方法。

1.1.2 基于平台定制产品

大规模定制(mass customization, MC)是以大规模的方式生产产品,但能满

足每个用户不同需求的生产模式。1970 年美国未来学家 Toffler 在 *Future Shock* 一书中提出了一种全新的生产方式的设想：以类似于标准化和大规模生产的成本与时间，提供用户特定需求的产品和服务^[9]。1987 年，Davis 首次将这种生产方式称为“mass customization”^[10]，即大规模定制。大规模定制的核心是大幅度提高产品品种的多样化和定制化水平，而不相应地增加成本；是个性化定制产品的大规模生产，其最大优点是提升制造企业战略优势和经济价值^[11]。

大规模定制带来的经济价值可以用图 1-1 来描述^[12]。虽然大规模生产早期需要在购买设备和工具、建立流程、培训等方面投入大量的资金，但是这些投资可以随着产量的增加分摊到单个产品个体上，结果是销售的产品越多，单个产品越能够维持较低的成本。因此，厂商可以降低产品定价而获得更多的订单。图 1-1 中的大规模生产成本曲线表明了这一现象。然而，用户一段时间内对大规模生产的标准化产品可能是满意的，但随着需求级别的提高，用户需要更多个性化的产品来满足不同的需求。例如，拥有一辆城市轿车的家庭，可能希望购买第二辆汽车用于旅游，他们倾向于选择 SUV 车型。虽然他们对以前的轿车品牌非常满意，但是这家厂商无法提供他们的新选型，这意味着厂商将丢掉一个潜在的用户。

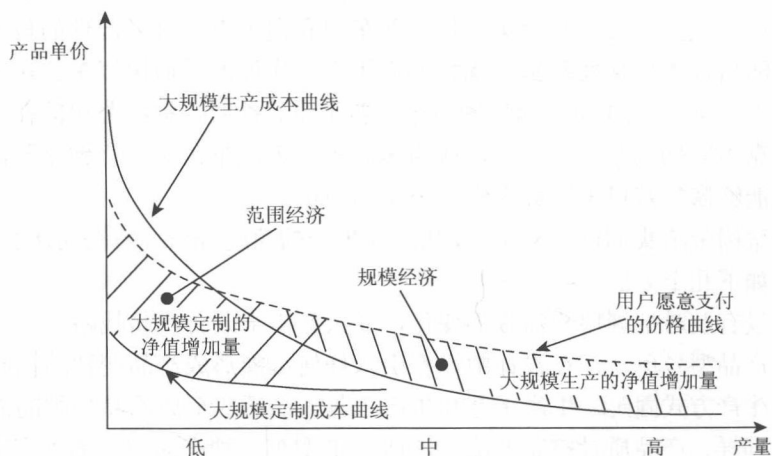


图 1-1 大规模定制的经济价值

为了满足用户需求的多样性，厂商必然无法对所有的产品大规模生产，产品成本升高但是用户愿意为此付出更多，因为他们的特殊需求得到了满足。为了尽量降低成本，缩短交货期，厂商通过大规模复用的方式争取到规模经济性，获得更高的利润，这就是大规模定制。由图 1-1 可见，大规模定制的成本曲线较为平缓，它与用户为了实现个性化需求而愿意付出的价格曲线达到最佳匹配，厂商的

效益较优，取得了范围经济^①。

大规模定制如何实现呢？工业界，特别是汽车制造业，是通过预先设计，为不同种类的产品引入通用平台，达到将部件应用于不同种类产品的目的。下面仍以汽车制造业为例，说明平台在大规模定制中的应用。

平台产品开发就是产品群一次规划，车型分步开发的模式，以实现零部件共用化与产品个性化的最佳平衡，构成削减成本与满足用户需求多元化的解决方案。

引入平台的汽车产品开发将形成平台开发、车型开发和总成开发的三个层次，开发资源将向前期的平台开发倾斜，单个车型开发周期缩短、效率提高，有利于满足对用户需求变化的快速响应。

平台为汽车主要部件提供了一个结构，它确定了车身体积、发动机和传动装置的类型及大小。组成平台的零部件，从设计和生产准备费用来看，通常是最昂贵的子系统。使用平台制造不同种类的汽车，使得平台旗下车型实现了大规模的零部件通用和生产线通用，显著降低了零部件开发/设计成本、模具开发/设计成本、人力成本和固定资产（厂房/机器人/标准设备等）投资等。

据统计，在 1980 年至 1991 年期间，平均每 3 年时间，使用平台策略的厂商的销售量上涨了 35%，而那些从头开始制造每一款新系列汽车的厂商的销售量下降了 7%^[13]。

基于平台大规模定制产品的策略在其他行业也屡见不鲜。例如，相机业的柯达公司，就是在 20 世纪 90 年代初，基于同一平台构建一系列有明显区别的不同相机型号，从而反超富士胶片，成为当时市场的主导厂商。波音公司前后研制的 757 和 767 客机，约有 60% 的零件是相同的。另外，家电、手机、计算机和办公耗材等都是成功使用平台进行大规模定制的行业。

平台在不同行业有不同的表现形式，但是其设计思路和作用大致相同，根据知名在线媒体公司 TechTarget 的定义如下。

定义 1-1 平台

平台是任何一种在其上可以构建其他技术或过程的技术基础。

平台开发的关键是企业内部制定强制性技术规范，明确通用化原则，未来采用同一平台的所有产品的开发与生产必须严格遵循这些规范和原则，这也是保证平台规模效益的前提。

1.1.3 定制产品的工程化

基于平台的大规模定制，使用平台作为产品研发的基础。这使得产品线中的

^① 即 economies of scope，如果某一个厂商同时生产多种产品的支出小于多个厂商分别生产这些产品的支出，就可以称为这个厂商获得了范围经济（亦可称为多产品经济）^[3]。

每个个体，都拥有一组能满足特定市场需求或任务需求的公共的、可管理的特性，并且是按预定义的方式，由平台开发而来的。称平台为核心资产，它们是构成产品线基础的资产。针对用户对个性化产品的需求，平台中还包含了大量可定制的特征。

创建平台是产品线工程化的关键。首先，关注哪些是所有产品共同的部分；其次，关注哪些是不同的。得到的工件适用于所有的产品的共性部分，同时必须具备一定灵活性，使其可以在不同的应用中得到复用，提供大规模定制的可能性。

这些工件的获得方式可以有三种。

(1) 厂商自己构建：或者从头构建，或者从老产品中挖掘。

(2) 购买：买来之后，直接使用。

(3) 委托第三方构建：与其他组织签订合同，委托其专门为自己构建。

一般，平台构建需要相当大的工作量，所以平台研发经常作为单独的开发项目。例如，汽车制造业中，平台是由专门的团队研发的，而基于平台开发单独的车型，则单独启动项目完成。经常是有多个产品系列，每个系列又构成了产品线家族中的子集。

实现灵活性要求找出和描述产品线上的产品在什么特征上不同，它们要实现哪些不同的需求，有时候这甚至意味着在底层架构上的调整。

例如，同一产品线的不同汽车可能有不同的刮水器和清洗器。所以在设计平台时需要为不同型号的汽车的不同刮水器和清洗器提供不同类型、不同尺寸的电机。并且，这种灵活性还受到产品线的具体个体的限制。例如，敞篷车就不需要后窗清洗器，以免将玻璃水洒到座位上。因此，定制敞篷车时，就需要对刮水器的灵活性加以限制，当敞篷车的顶篷敞开时，后玻璃清洗器是不可用的。

设计这种灵活性需要对各种产品的差异性有深刻认识，哪些差异可以通过在平台上引入配置项获得支持，哪些差异不能或者不必要在平台上获得支持（可能是成本、技术和管理等造成的）。这些确定下来后，也就确定了产品线中存在几种规格的产品，可以精确地定义各种产品究竟在什么部位不同，从而使其尽量具有更多的通用之处。

这里定义的灵活性即软件产品线工程中所谓的“可变性”，可变性是大规模定制的基础，将在第4章详细讨论可变性问题。

在工业界，共享同一平台，拥有相似特征的产品称为产品线或产品家族，平台和产品线具有很强的关联性。属于一个产品线的一批可管理的汽车基于同一个平台。但是，实际的汽车企业中情况复杂得多：在汽车开发的历史过程中，平台可能在不同的产品线中共享，新的产品线使用早期产品线的平台等。这样，平台和产品线之间的简单关联关系消失或淡化，平台和从其得到的产品之间的追踪信息需要仔细的管理。没有这些追踪信息，几乎不可能发现平台的哪个部件在哪个