

软件体系结构原理、方法与实践

严禁复制

软件产品线-引入



新的开始，新的起点，让我们一起为梦想而努力。

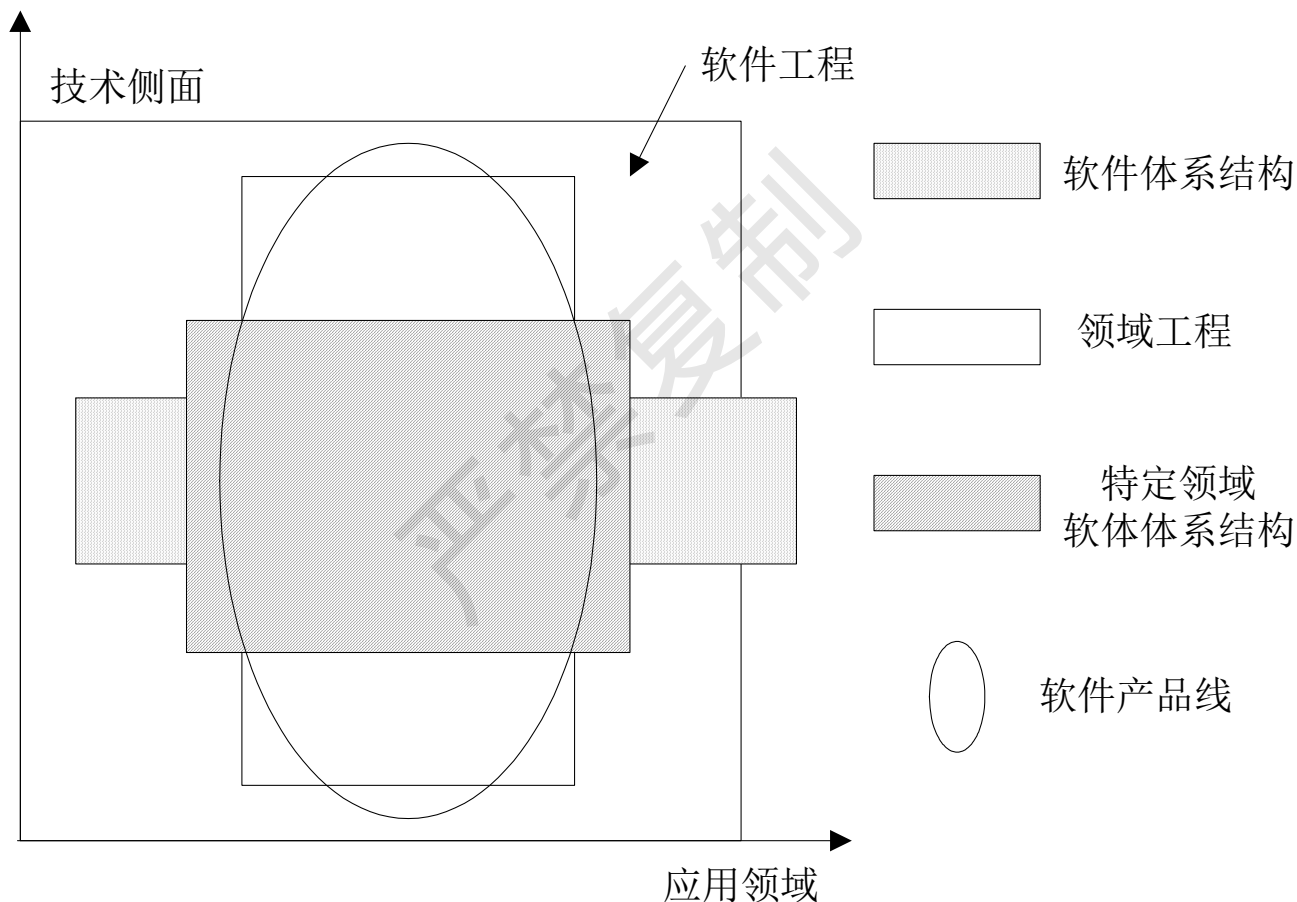
❁ 软件产品线-综述

- 软件产品线(software product line)是一个适合专业软件开发组织进行软件开发的方法，能有效的提高软件生产率和质量、缩短开发时间，降低开发成本；它也是一个新兴的，多学科交叉的研究领域，研究内容和范围都相当广泛。
- 软件产品线与软件体系结构有着密切的关系。软件体系结构的开发是大型软件系统开发的重要环节，体系结构的开发在软件产品线中起着至关重要的作用，在这种开发过程中，基于同一体系结构，可以创建具有不同功能的多个系统。在软件产品族之间**共享体系结构和一组可重用的构件**，可以降低开发和维护成本。

❁ 软件产品线的出现和发展

- 产品线的起源可以追溯到1976年Parnas对程序族的研究。软件产品线的实践早在20世纪80年代中期就出现
- 据HP公司1996年对HP、IBM、NEC、AT&T等几个大型公司分析研究，他们在采用了软件产品线开发方法后，使产品的**开发时间**减少1.5-2倍，**维护成本**降低2 - 5倍，**软件质量**提升5 - 10倍，**软件重用**达50% - 80%，**开发成本**降低12% - 15%
- **软件产品线的**发展得益于**软件体系结构**的发展和**软件重用技术**的发展。

❁ 软件产品线在软件工程中地位



❁ 软件产品线的基本概念

- 将利用了产品间公共方面、预期考虑了可变性等设计的产品族称为产品线(Weiss和Lai)。
- 产品线就是由在系统的组成元素和功能方面具有共性和个性的相似的多个系统组成的一个系统族。
- 软件产品线就是在一个公共的软件资源集合基础上建立起来的，共享同一个特性集合的系统集合(Bass,Clements和Kazman)。
- 一个软件产品线由一个产品线体系结构、一个可重用构件集合和一个源自共享资源的产品集合组成，是组织一组相关软件产品开发的方式(Jan Bosch)。



❁ 软件产品线的基本概念

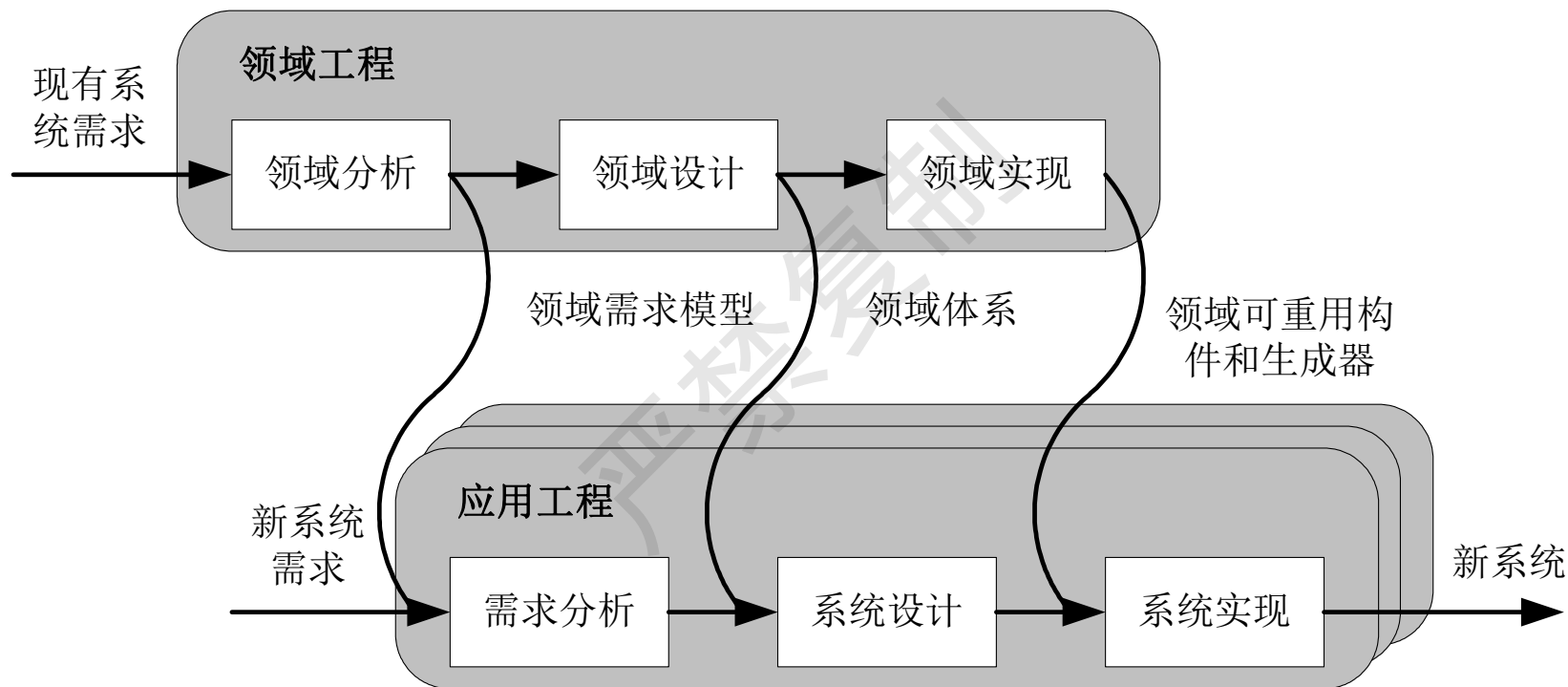
- CMU/SEI: 产品线是一个**产品集合**，这些产品共享一个公共的、可管理的特征集，这个特征集能满足选定的市场或任务领域的特定需求。这些系统遵循一个预描述的方式，在公共的核心资源基础上开发的。
- 根据SEI的定义，软件产品线主要由两部分组成：**核心资源、产品集合**。**核心资源是领域工程的所有结果的集合，是产品线中产品构造的基础。也有组织将核心资源库称为“平台”。**
- **核心资源必定包含产品线中所有产品共享的产品线体系结构，新设计开发的或者通过对现有系统的再工程得到的、需要在整个产品线中系统化重用的软件构件。**

关于软件产品线下列说法正确的是:

- ☒ A 软件产品线主要由核心资源与产品集合组成
- ☒ B 软件产品线的发展得益于软件体系结构的发展和软件重用技术的发展
- ☒ C 产品线是一个产品集合，这些产品共享一个公共的、可管理的特征集，这个特征集能满足选定的市场或任务领域的特定需求
- ☐ D 产品线就是由在系统的组成元素和功能方面具有共性的相似多个系统组成的一个系统

提交

❁ 软件产品线的过程模型 – 双生命周期模型



❁ 软件产品线的过程模型 – 双生命周期模型

领域工程阶段的主要任务：

领域分析：利用现有系统的设计、体系结构和需求建立领域模型。

领域设计：用领域模型确定领域产品线的共性和可变性，为产品线设计体系结构。

领域实现：基于领域体系结构开发领域可重用的资源(构件，文档，代码生成器等)

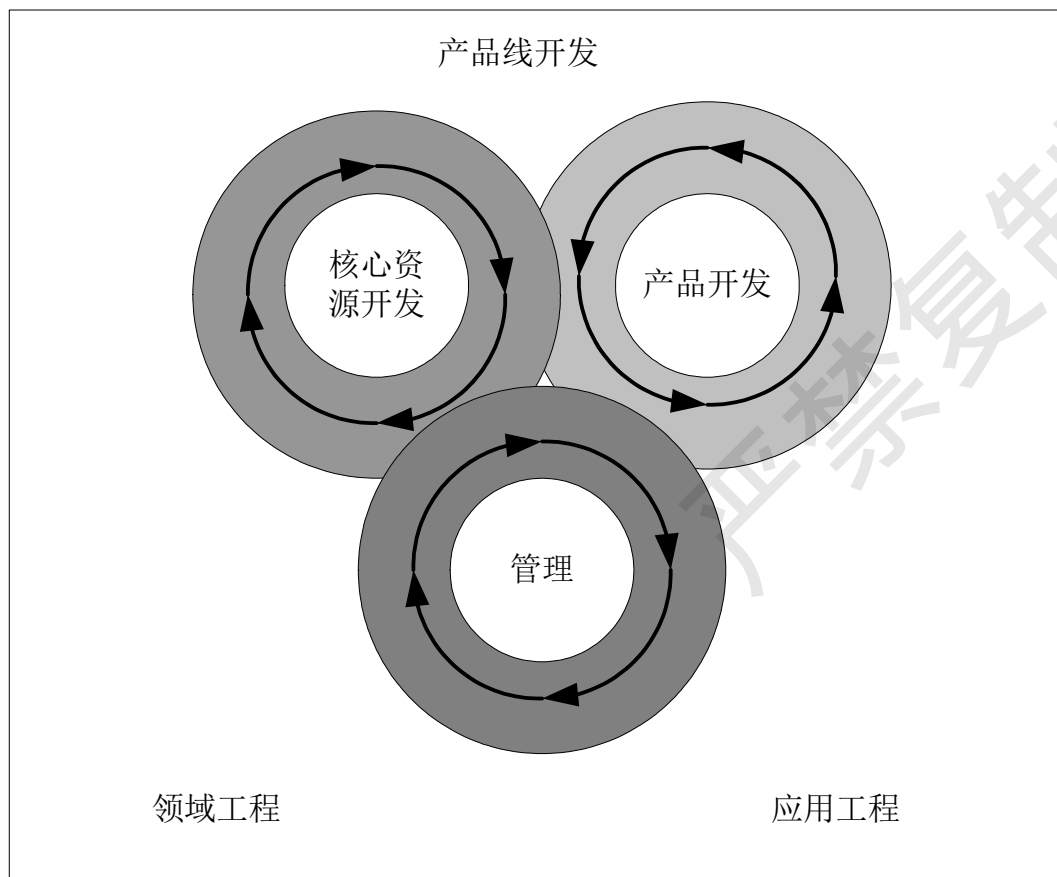
应用工程生成新产品阶段

需求分析：将系统需求与领域需求比较，划分出领域公共需求和独特需求，得出系统说明书。

系统设计：在领域体系结构基础上，结合系统独特需求设计应用的软件体系结构。

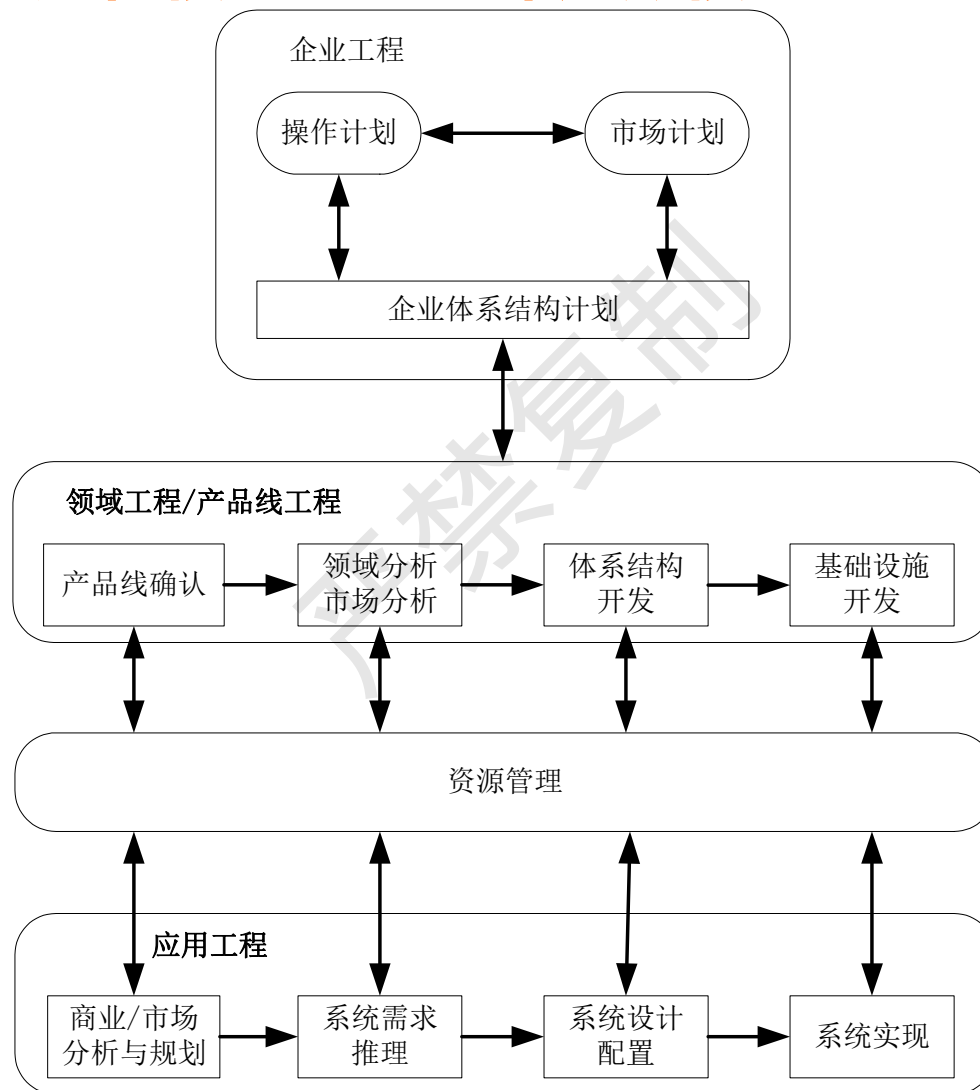
系统实现：遵照应用体系结构，用领域可重用资源实现领域公共需求，用定制开发的构件满足系统独特的需求，构件新系统。

❁ 软件产品线的过程模型 – SEI模型



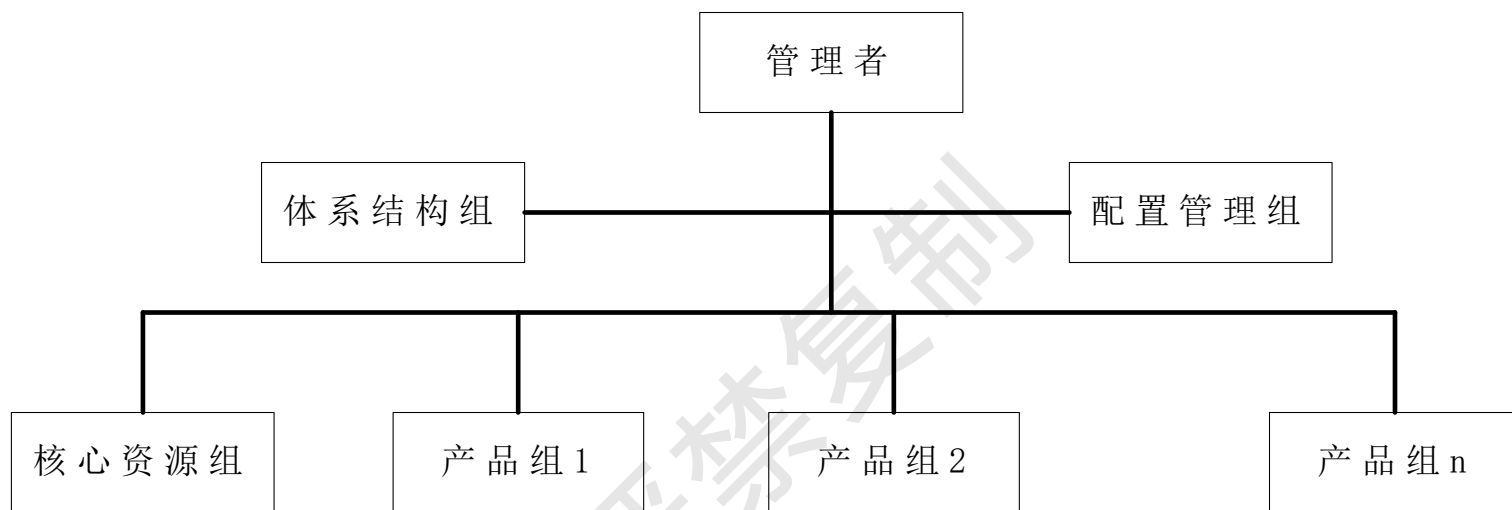
- 循环重复是产品线开发过程的特征，也是核心资源开发、产品线开发以及核心资源和产品之间协作的特征；
- 核心资源开发和产品开发没有先后之分；
- 管理活动协调整个产品线开发过程的各个活动，对产品线的成败负责；
- 核心资源开发和产品开发是两个互动的过程，三个活动和整个产品线开发之间也是双向互动的。

软件产品线的过程模型 – 三生命周期模型



新的开始，新的起点，让我们一起为梦想而努力。

❁ 软件产品线的组织结构 – 典型结构



典型产品线开发组织结构

体系结构组监控核心资源开发组和产品开发组以保证核心资源和产品能够遵循体系结构,同时负责体系结构的演化;配置管理组维护每个资源的版本体系结构组、核心资源开发组与负责独立产品开发的小组互相独立

❁ 软件产品线的组织结构 – SEI产品线组织结构

- 市场人员是产品线和产品能力、客户需求之间的沟通桥梁；
 - 核心资源组负责体系结构和其他核心资源的开发；
 - 应用组负责交付给客户的系统的开发；
 - 管理者负责开发过程的协调、商务计划等。
-
- ✓ 设有独立核心资源小组的组织结构通常合适于至少由50到100人组成的较大型的软件开发组织，设立独立的核心资源小组可以使小组成员将精力和时间集中在核心资源的认真的设计和开发上，得到更通用的资源。
 - ✓ 不设立独立的核心资源小组，核心资源的开发融入各系统开发小组中，只是设立专人负责核心资源开发的管理。这种组织结构的重点不在核心资源的开发上，所以比较适合于组成产品线的产品共性相对较少，开发独立产品所需的工作量相对较大的情况。也是小型软件组织向软件产品线开发过渡时采用的一种方法。

❁ 软件产品线的组织结构 – Jan Bosch产品线组织结构

- **开发部门：**所有的软件开发集中在一个部门，每个人都可承担领域工程和应用工程中适合的任务，简单、利于沟通，适用于不超过30人的组织。
- **业务部门：**每个部门负责产品线中一个和多个相似的系统，共性资源由需要使用它的一个和几个部门协作开发，整个团体都可享用。资源更容易共享，适用于30-100人的组织，**主要缺点是业务部门更注重自己的产品而将产品线的整体利益放在第二位。**
- **领域工程部门：**有一个专门的单位——领域工程部门负责核心资源库的开发和维护，其他业务单位使用这些核心资源来构建产品。这种结构可有效的降低通讯的复杂度、保持资源的通用性，适于超过100人的组织。**缺点是难以管理领域工程部门和不同产品工程部门之间的需求冲突和因此导致的开发周期增长。**
- **层次领域工程部门：**对于非常巨大和复杂的产品线可以设立多层（一般为两层）领域工程部门，不同层部门服务的范围不同。**这种模型趋向臃肿，对新需求的响应慢。**

❁ 软件产品线的建立方式

	演化方式	革命方式
基于现有产品	<p>基于现有产品架构设计产品线的架构，经演化现有构件，开发产品线构件</p> <p>优点：风险最小，第一个产品面世时间早</p> <p>缺点：完成产品线核心资源的总周期和总投资较大</p>	<p>核心资源的开发基于现有产品集的需求和可预测的、将来需求的超集</p> <p>优点：开发一个不受现有产品集存在问题限制的、全新的平台，总周期和总投资较少</p> <p>缺点：风险大，第一个产品面世的时间会推后</p>
全新产品线	<p>产品线核心资源随产品新成员的需求而演化</p> <p>优点：先期投资少，风险小，第一个产品面世时间早，可以减少因经验不足造成的初始阶段错误的修正代价</p> <p>缺点：已有的产品线核心资源会影响新产品的需求协调，使成本加大</p>	<p>开发满足所有预期产品线成员的需求的核心资源</p> <p>优点：一旦产品线核心资源完成，新产品的开发速度将非常快，总成本也将减少</p> <p>缺点：风险大；对新领域的需求很难做到全面和正确，使得核心资源不能像预期的那样支持新产品的开发</p>

- 将现有产品演化为产品线
- 用软件产品线替代现有产品集
- 全新软件产品线的演化
- 全新软件产品线的开发

❁ 软件产品线的演化

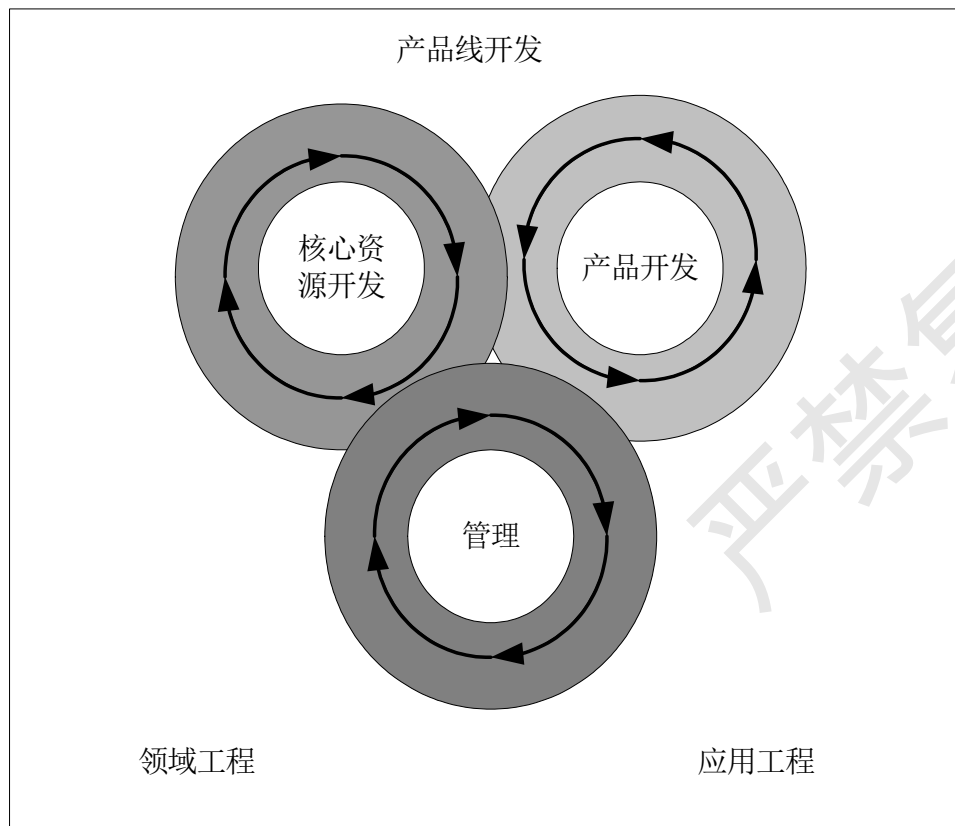
- 从整体来看，软件产品线的发展过程有三个阶段，开发阶段、配置分发阶段和演化阶段。
- 引起产品线体系体系结构演化的原因：产品线与技术变化的协调、现有问题的改正、新功能的增加、对现有功能的重组以允许更多的变化等等。
- 产品线的演化包括产品线核心资源的演化、产品的演化和产品的版本升级。这样在整个产品线就出现了：核心资源的新旧版本、产品的新旧版本和新产品等。它们之间的协调是产品线演化研究的主要问题。

下列说法正确的是:

- ☐ A 在双生命模型中领域工程阶段的主要任务有领域需求，领域分析，领域设计。
- ☒ B 在产品线组织结构中业务部门更注重自己的产品而将产品线的整体利益放在第二位
- ☐ C 软件产品线的发展过程有三个阶段，开发阶段、配置分发阶段和测试阶段
- ☒ D 产品线的演化包括产品线核心资源的演化、产品的演化和产品的版本升级

提交

❁ 产品线基本活动 - 概述



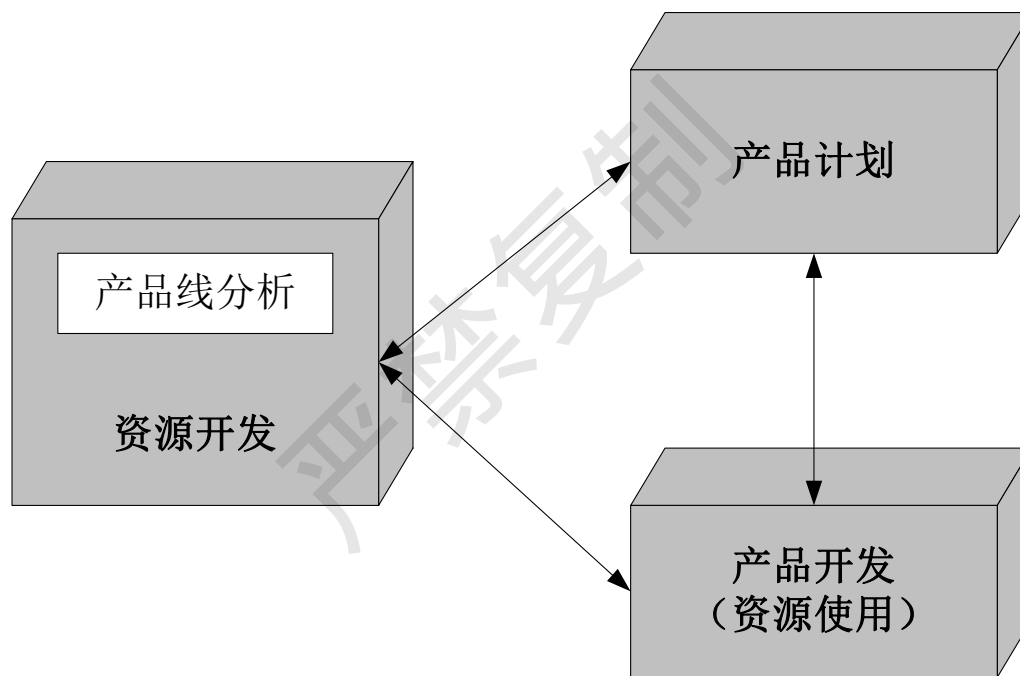
每个旋转环代表一个基本活动, 三个环连接在一起, 不停地运动着。三个基本活动交错连接, 可以任何次序发生, 且是高度重叠的。旋转的箭头表示不但核心资源库被用来开发产品, 而且已存在的核心资源的修改甚至新的核心资源常常可以来自产品开发。在核心资源和产品开发之间有一个强的反馈环, 当新产品开发时, 核心资源库就得到刷新。对核心资源的使用反过来又会促进核心资源的开发活动。另外, 核心资源的价值通过使用它们的产品开发来得到体现。

❁ 产品线基本活动 – 产品线分析 – 上下文

产品线分析是产品线的需求工程，是业务机遇的确认和产品线体系结构的设计之间的桥梁。产品线分析强调：

- (1)、通过捕获风险承担着的观点来揭示产品线需求。
- (2)、通过系统的推理和分析、集成功能需求和非功能需求来完成产品线需求。
- (3)、产品线设计师对产品线需求的可用性。

产品基本活动 – 产品线分析 – 上下文

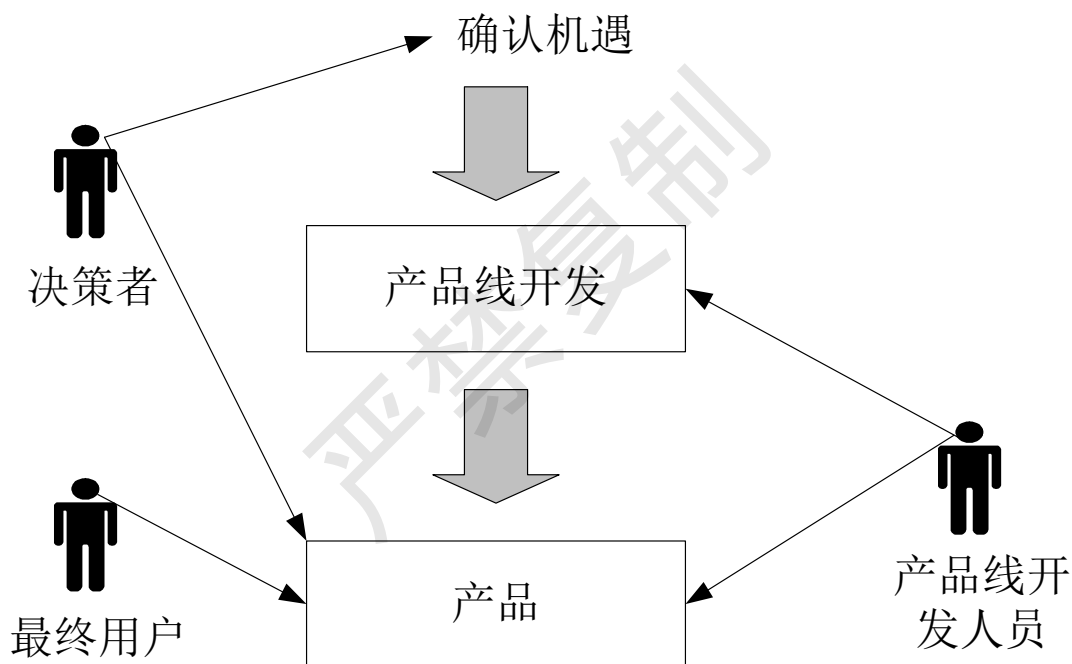


❁ 产品线基本活动 – 产品线分析 – 上下文

产品线分析是把对业务机遇的初步确认细化为需求模型，对正在开发的产品线而言，需要获取：

- (1)、组织的业务目标和约束。
- (2)、包含在产品线中的产品。
- (3)、最终用户和其它风险承担着的需求。
- (4)、大粒度重用的机会。

产品线基本活动 - 产品线分析 - 风险承担者观点

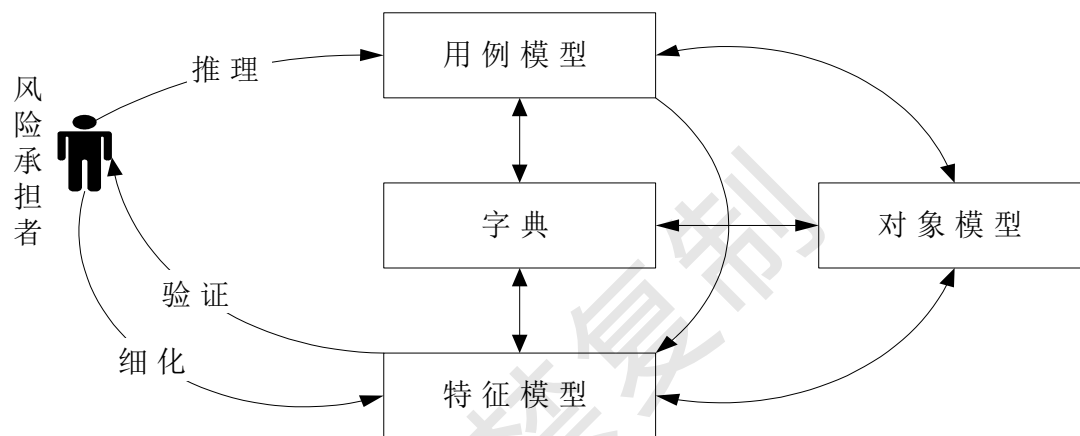


❁ 产品线基本活动 – 产品线分析 – 需求建模

在开始产品线分析时，要先确认以下问题：

- (1)、将要开发的产品线是否与组织的任务、业务目标和约束保持一致
- (2)、产品线将由哪些产品组成
- (3)、对组织而言产品线的开发是否有意义？与之相关的成本、风险和利润是什么？

产品线基本活动 – 产品线分析 – 需求建模



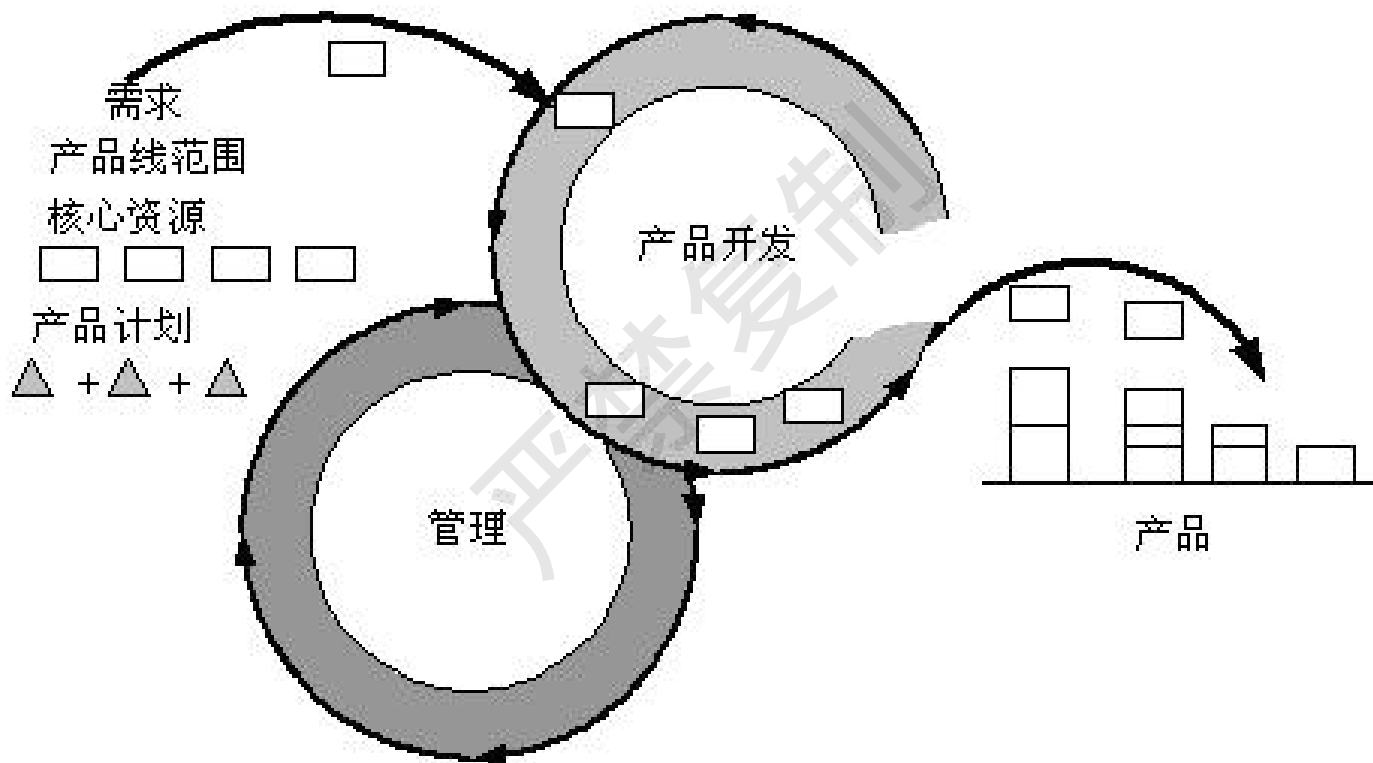
用例模型：产品线风险承担者和他们与产品线的关键交互，风险承担着将验证产品线的可接受性。

特征模型：描述产品线风险承担着的观点。它捕获产品的功能特征和产品线及产品的质量属性。

对象模型：描述产品线支持上述特征的功能，以及这些功能的通用性和可变性。

字典：定义用在产品中的，支持产品线需求的一致观点的术语。

产品线基本活动 - 产品开发



关于软件产品线的分析，下列说法正确的是：

- ☒ **A** 产品线分析是产品线的需求工程，是业务机遇的确认和产品线体系结构的设计之间的桥梁
- ☐ **B** 产品线需求模型是用例模型，场景模型，特征模型，对象模型相互联系的工作产品集合。
- ☒ **C** 产品开发活动取决于产品线范围、核心资源库、产品计划和需求输出。
- ☒ **D** 产品线分析是把对业务机遇的初步确认细化为需求模型

提交

❁ 软件产品线体系结构的设计 – 概述

- 软件产品线体系结构指一个软件开发组织为一组相关应用或产品建立的公共体系结构。
- 同领域模型一样，软件产品线体系结构中也可以分为共性部分和个性部分。
- 产品线体系结构是产品线核心资源的早期和主要部分，在产品线的生命周期中，产品线体系结构应该保持相对小和缓慢的变化以便在生命周期中尽量保持一致。产品线体系结构要明确定义核心资源库中软件构件集合及其相关文档。

❁ 软件产品线体系结构的设计 – 变化需求

- **产品线的产品有着不同的质量属性。**例如，一个产品需要高度安全但运算速度要求低，另一产品可能需要运算速度快但对安全没有特别要求，产品线体系结构需要足够灵活来支持两个产品。
- 产品之间的差异可能体现在各个方面：行为、质量属性、平台和中间件技术、网络、物理配置、规模等，**产品线体系结构需要对这些差异进行处理。**

❁ 软件产品线体系结构的设计 – 个性实现机制

- **继承**：用于对象方法在产品中的不同实现和扩展。
- **扩展和扩展点**：通过增加行为和功能扩展构件的某些部分。
- **参数化**：用于构件的行为特征可以抽象并在构件构造时可确定的情况，如宏定义和模板都是参数化方法的一种。
- **配置和模块互连语言**：用于定义系统构造时结构和构件的选择方式和结果。
- **自动生成**：用更高级的语言来定义构件的特征，并自动生成相应的构件。
- **编译时不同实现的选择**：用于构件的变化可以通过选择不同代码段实现的情况，如 “#ifdef” 。

❁ 软件产品线体系结构的设计 – 面临的主要困难和问题

- 没有熟练的体系结构设计师。
- 参数化问题：参数化是一个支持产品线体系结构变化的有效的方法，但要注意过于参数化易使系统难以使用和理解，参数化过少又会限制系统的变化能力。过早的参数绑定易使变化困难，绑定过晚（运行时刻的动态绑定）易导致性能降低。
- 必须有良好的领域分析和产品规则基础作保证，对技术发展趋势要做出准确预测，还要注意吸取相关领域的教训。
- 软件开发、管理和市场人员的管理和文化对基于软件体系结构开发的适应程度。
- 目前，支持软件体系结构设计的CASE工具较少。
- 产品线体系结构设计师和产品开发者之间的沟通。

❁ 软件产品线体系结构的设计 – 标准化和定制

- 产品线体系结构的设计有两种方式：使用标准体系结构和体系结构定制。作为标准，会有众多的软件开发组织遵循它，开发各自的应用或者为该体系结构提供基础构件和应用开发的辅助工具等。采用标准体系结构作为产品线的软件体系结构，可以获得第三方软件开发组织的支持、有效地缩短开发时间、提高产品的可靠性和与同类系统的可集成性等。
- 为适应应用的规模增大、复杂度提高，软件技术不断发展，相继出现了中间件技术、软件产品线等。
- 体系结构风格是一个使产品和产品线具有良好的可移植性的结构，产品和产品线通过最小的修改就可移植到一个新的平台上。

关于产品线体系结构的设计说法正确的是：

- ☒ **A** 软件产品线体系结构指一个软件开发组织为一组相关应用或产品建立的公共体系结构
- ☒ **B** 采用标准体系结构作为产品线的软件体系结构，可以获得第三方软件开发组织的支持
- ☒ **C** 体系结构风格是一个使产品和产品线具有良好的可移植性的结构，产品和产品线通过最小的修改就可移植到一个新的平台上
- ☐ **D** 缺少支持软件体系结构设计的CASE工具是软件产品线体系结构设计面临的主要困难和问题

提交