

---

# 软件项目产品化之路

作者：阎伟

日期 2008-10-28

---

# 目 录

<b>1. 前言 .....</b>	<b>5</b>
<b>2. 产品化之路 .....</b>	<b>6</b>
2.1. 困惑 .....	6
2.2. 光明之路还是不归之路? .....	6
2.3. 光明之路也是荆棘之路 .....	7
<b>3. 软件产品化解决之道.....</b>	<b>8</b>
3.1. 企业经营.....	8
3.1.1. 有所为有所不为.....	8
3.1.2. 市场可行性分析.....	8
3.2. 企业管理.....	9
3.2.1. 组织结构变革.....	9
3.2.1.1. 由销售为中心转化为以市场为中心.....	9
3.2.1.2. 技术部门的细分和演进 .....	10
3.2.2. 项目和产品之间的关系 .....	10
3.2.2.1. 开发过程中的项目目标与产品目标.....	10
3.2.2.2. 辩证处理项目与产品的关系 .....	11
3.2.3. 对人员的激励和考核.....	11
3.3. 软件技术.....	12
3.3.1. 软件技术方案与框架.....	12
3.3.1.1. 稳定技术方案和框架 .....	12
3.3.1.2. 技术方案和框架的演进 .....	13
3.3.1.3. 技术方案和框架的常见误区 .....	13
3.3.1.4. 技术传播和转化 .....	14
3.3.2. 行业领域知识.....	15
3.3.2.1. 要比客户还要专业 .....	15
3.3.2.2. 要基于业务又要高于业务 .....	15

---

3.3.2.3.    要重视对领域的管理 .....	17
3.3.3.    软件开发过程 .....	17
3.3.3.1.    产品化开发过程 .....	17
3.3.3.2.    软件开发过程选择 .....	19
3.3.4.    软件开发组织机构 .....	20
3.3.5.    软件生产工业化 .....	21
3.3.5.1.    传统工业化内容 .....	22
3.3.5.2.    软件生产工业化 .....	22
3.3.5.3.    实现软件生产的自动化 .....	23
3.3.6.    软件产品质量 .....	24
3.3.6.1.    客户全程参与 .....	25
3.3.6.2.    人机交互体验 .....	25
3.3.6.3.    业务场景模拟实验室 .....	26
3.3.6.4.    自动化技术测试实验室 .....	27
3.4.    软件人才 .....	27
3.4.1.    人才结构问题 .....	27
3.4.2.    对关键人员的依赖问题 .....	28
3.5.    本章小结 .....	29
<b>4.    案例分析 .....</b>	<b>30</b>
4.1.    案例背景 .....	30
4.2.    企业经营与发展问题 .....	31
4.2.1.    企业发展战略 .....	31
4.2.2.    供应链管理系统 .....	32
4.2.3.    电子商务平台 .....	33
4.2.4.    专业化物流软件 .....	34
4.2.5.    小结 .....	35
4.3.    企业管理问题 .....	36
4.3.1.    企业文化 .....	36
4.3.2.    组织结构问题 .....	37

---

4.3.3.	业务发展	37
4.3.4.	技术管理	38
4.4.	软件技术	38
4.4.1.	软件技术方案和框架	38
4.4.2.	行业领域知识	39
4.4.2.1.	领域知识的管理	39
4.4.2.2.	物流运营模型	40
4.4.3.	软件研发管理实践	41
4.4.3.1.	软件开发过程	41
4.4.3.2.	每日晨会制度	41
4.4.3.3.	产品进度公告板	41
4.4.3.4.	测试人员驱动开发	42
4.4.3.5.	文档与代码	42
4.4.4.	软件生产的工业化	43
4.4.4.1.	模型驱动开发	43
4.4.4.2.	自动化生成技术	43
4.4.4.3.	可复用组件技术	43
4.4.5.	软件产品质量	44
4.4.5.1.	客户全程参与	44
4.4.5.2.	测试数据工厂	44
4.5.	软件人才	44
4.5.1.	技术团队的建立	44
4.5.2.	技术团队的培养	45
4.5.3.	技术团队的使用	45
4.5.4.	技术团队的保留	45
4.6.	小结	46

## 1. 前言

---

本文阐述了软件项目产品化过程中所面临的困难和问题,对问题的根源进行了分析。在此基础上提出了软件产品化的解决之道,并针对某物流软件项目产品化的案例进行了具体分析。

在第2章对产品化道路中常见的问题进行了故事性描述。第3章对软件产品化解决之道进行了具体理论阐述,并围绕着企业经营、企业管理、软件技术、软件人才四个方面进行了详细讨论,以及对软件产品化的过程中存在的问题进行了系统化的分析。第4章选择了一个物流行业的具体软件产品化的案例,通过对案例的细致分析,对第3章的理论进行了具体解说。

本文适合软件行业从业人员、IT企业中高层管理人员、IT行业研究人员、计算机专业学生阅读。本文涉及企业管理、IT技术、IT管理、市场营销、工程制造等多个领域方面的话题和知识,需要读者进一步去了解这些方面的知识。

本文所阐述的内容,都是作者多年的知识沉淀和工作经验积累,希望能给读者一定启示作用。由于本人知识、阅历和能力所限,所表达的观点和所涉及的知识领域如有错误,欢迎读者指正。

未经作者书面许可,任何人不得以任何形式复制、转载、散布、引用、变更、传播或出版该文章之全部或局部内容。

---

## 2. 产品化之路

---

### 2.1. 困惑

软件项目产品化是大量软件企业，特别是应用型软件研发企业所必须面临的问题。不论是小型的软件公司和中大型的软件企业，在面对软件项目和软件产品，都有诸多困惑。到底是做项目还是做产品？

### 2.2. 光明之路还是不归之路？

在企业发展过程中，往往开始是项目驱动型，有一个好的项目，一个公司就这样发展起来了。有了这个项目做基础，公司自然会接到越来越多的项目。但当同类项目越接越多，人员队伍开始扩展，项目周期无法保证，产品质量问题、用户需求把握不准确等一系列问题都接踵而至。

这时候，企业开始考虑将项目进行整理和整合，进行产品化开发。这将是十分痛苦的过程。因为一开始软件就没有按照产品的思路来设计，为了赶时间进度，满足客户的一些“独特的”个性化需求，代码的可维护性很差，文档基本上没有，就算是有也是过时的。开发平台也是一个项目一个样，这个管理系统 Java,那个是业务系统是 VB.NET,门户网站是 PHP,客户要求（没办法啊）……。除了对原有代码推倒重来，基本上没有其他可行的道路。

历经几次痛苦的过程，产品终于相对成型了，整个技术构架和功能都相对于原有项目都有较大提高，开发语言和平台也基本统一，代码的质量相对从前有较大提高。但相关的问题又随即产生。

问题 1（小型公司更为常见）：在产品销售和实施的过程中，客户的差异性很大，不论是技术方面还是业务方面都有很多个性化的需求。现有产品如果去满足客户的要求，改动成本相当大，很多模块的业务代码都要重新开发，部分需求，产品现有框架不能满足；如果不改，现有产品又不能满足客户要求，将失去订单。怎么办？

公司从业务考虑，往往是先把项目接下来再说。但难题就摆在研发人员面

前。如果按照客户要求来做，基本上和做项目没啥区别；如果按照产品方式来做，在客户要求的时间要求上，基本上不可能。由于项目压力，只好先做出来再说。所谓产品化，只好先扔在一边。毕竟公司考核你的是能否完成客户的项目，大家的绩效奖金和此息息相关。阳春白雪（产品化）虽然好，但关系到切身利益，下里巴人（顾自己腰包）才是实实在在的。

问题 2（中大型公司更为常见）：由于产品化往往是专门的部门来负责，产品的内部行销往往成为问题。具体项目开发部门和项目组，常反感自己的项目成为产品的试验田，都不愿意做白老鼠，因此推行很困难。产品所涉及的基础技术框架，需要有较长的学习成本，培训、推广等都很花时间，大家手上项目都很忙，哪有时间来学这东西。另外，往往技术比较优秀的开发人员，个性都比较强，对别人做的东西，看不上眼，不买账，有抵制情绪。产品化在企业内部进行推广，步履维艰。

最终，产品化热闹了一阵以后，大家又恢复到项目方式，产品化，只剩下房间角落柜子中的一堆文件和光盘而已，公司为了眼前利益，对此也是睁着眼闭只眼，产品化也没有人提起了。

幸福的家庭往往都是相似的，不幸的家庭各有各的不幸。上面的这段情节，年复一年的在一家家软件公司中重演，演绎着各自的故事。

软件产品化到底是光明之路还是条不归之路？

## 2.3. 光明之路也是荆棘之路

软件项目产品化是企业发展到一定阶段，必然碰到的问题，这一问题不可能回避。由于软件需求的个性化差异，特别是应用型软件的研发，从本质上决定了软件产品化道路的困难性。因此，很多企业在面临业务迅速发展 的情况下，项目化开发是问题多，产品化开发问题更多

产品化是软件企业在发展过程的必经之路。解决的好，公司业务就会上到一个新的台阶，解决不好，公司因此停滞，甚至倒下都是很有可能的。所以，产品化道路是条光明之路也是荆棘之路。

因此，如何解决项目软件产品化过程中存在的问题，找到一条通往成功的道路，是摆在所有软件公司面前的课题。

## 3. 软件产品化解决之道

软件产品化所面临的困难和挑战，不仅仅是技术层面的问题，而是一个牵涉到企业整个发展过程各个方面的综合性问题。因此，解决软件的产品化，也不仅仅是从技术层面来解决这一问题，而是要从企业经营、企业管理、软件技术、软件人才四个方面综合性的进行分析和解决。

### 3.1. 企业经营

很多公司产品化失败，表象上是软件产品的质量和功能问题，其实究其根本性原因，往往是一开始就是错误的。项目开始起初，对项目所涉及到业务的发展没有进行过认真、科学的分析和论证，是导致后续苦果的始作俑者。

#### 3.1.1. 有所为有所不为

在企业接到一个项目的时候，为了生存或为了利益，往往对项目没有选择性。有所为有所不为，这句话说的容易，当面临生存压力或利益诱惑的时候，做起来却很难，特别是对于一些中小型企业和还在初创期的企业。

有所为有所不为，汝今能持否？

不考虑自身的特长，自身的积累，只要有项目就做，有钱就赚，这是很多公司，特别是创业之初的中小型公司所常犯下的错误。

因此，项目一开始，经营管理人员就要认清自我，能有做到有所为有所不为，才能为企业发展奠定良好基础。

#### 3.1.2. 市场可行性分析

项目一旦成功，这时候企业所常常犯的第二个毛病往往就要出现了：头脑一热，在没有冷静分析的基础上，就一头扎进去。

**软件产品化的第一个问题，不是如何进行产品化，而是是否需要产品化。**

在进入该产品市场之前，必须认真做好产品市场的可行性分析。在产品化



之前，最好不要仅仅只根据一两个单个项目来决定进入某个市场，而是要在进行这些项目的过程中，不断的进行产品的市场调查和分析，来论证产品化的可行性。

- 项目所涉及的目标市场是否有足够大，而有必要进行产品化？
- 这个市场是处于怎样一个竞争情况？是现金牛，还是瘦狗？
- 在这个市场是否有其他竞争对手，他们的实力如何？
- 我们是否有进入这个产品细分市场的技术研发能力？
- ……

因此，做好产品所涉及的市场可行性分析，扎实认真做好市场调研，科学严谨的进行市场分析，是软件产品化能顺利成功的前提保证。

## 3.2. 企业管理

项目软件产品化的过程，往往也是企业迅速发展的时期，在这个过程中，如何有效的进行管理方法和手段的调整，解决企业由项目向产品转化过程中碰到的问题，对于能否顺利进行产品化，是重要的保障。

### 3.2.1. 组织结构变革

由于软件产品化的需要，原有的以项目为导向的企业组织结构往往不能适应这一变化，需要进行重新调整。在这一过程中，会碰到很多组织管理性的问题，需要经营管理者来调整和理顺。

#### 3.2.1.1. 由销售为中心转化为以市场为中心

项目型的软件公司，是以项目为驱动的，所以在整个项目过程中，往往是以销售为中心的运作方式。在项目型软件公司，销售部门的话语权是最大的，因为他们是给公司创造利润的核心。而在软件产品化的过程中，企业会逐步从以销售为中心转化为以市场为中心。这时候，市场部成为公司的核心部门，销售和技术等部门都要以产品为单元进行运作和管理。

因此，由销售为中心向以市场为核心的转变，是企业要过的第一关。这一

关如果没过好，会对企业的经营产生重要影响，导致产品化的进程受阻甚至失败。

在这个转化的过程中，需要调整好销售、市场等其他部门之间的利益冲突，调整好人员的工作职责和重心，要转变原有的工作习惯和方式，这些都是转化的过程中需要特别注意的。在这个转化的过程中，改变员工的思维方式和做事习惯是最难的，但也是最为重要的。

### 3.2.1.2. 技术部门的细分和演进

在软件产品化的过程中，技术部门为了适应产品化的要求，也存在进一步细分和演化的需要。

原有的单一的研发部门往往不能同时满足项目开发和产品研发的双重需要，因此需要成立负责项目开发和产品研发的单独部门。

项目开发部门负责项目的具体开发和产品的实施过程，产品研发部门负责将在项目开发部门在开发过程中积累下来的软件资产进行产品化，并帮助和指导产品在项目过程中的实施。

因此，项目开发部和产品研发部是相互协作，相互依存的，并不是割裂和独立的。在项目的实施和产品的研发活动中，都需要两个部门的协作，只是二者角色不同、职责不同。

在研发队伍的不断扩大的过程中，涉及到项目管理、质量控制与质量保障、基础技术与构架、项目维护和服务、组织过程改进等活动，都需要逐步成立专门的部门进行专业化分工和管理。如何在产品化的同时，完善和提高公司技术管理水平，是摆在每个软件公司面前的课题。

## 3.2.2. 项目和产品之间的关系

### 3.2.2.1. 开发过程中的项目目标与产品目标

在项目软件产品化的过程中的开发活动，往往不是仅仅具有项目目标或产品目标，而是综合了项目目标和产品目标的。在项目软件产品化的过程中，这种双重目标性，将是软件开发活动的重要特征。根据具体开发任务的不同，

可以分为以项目为重点和以产品为重点两种。

以项目为重点的开发，主要发生在外部项目中。重心是以按照项目时间、进度、成本完成项目。因此产品化的重点是产品在项目中的具体应用。在碰到产品不能很快解决的问题的时候，往往不能等产品研究解决方案，而是需要先解决问题优先。

以产品为重心的开发，主要发生在内部产品开发中。这时候的重心是产品的功能、技术构架优化、新技术的引入，项目需求的融入等。在产品开发的过程中，对在项目开发过程中积累的产品的不足或新的需求，在产品开发的过程中，进行进一步的分析，选择有普遍性和创新性的需求，重点引入到产品功能中去。

因此，除了一些面对终端消费者的套装软件，可以采用纯产品的方式进行开发，绝大多数具有行业特点的产品软件，都需要在实际实施的过程中进行定制化开发，因此就形成了上述项目和产品双重目标的研发过程。

#### 3.2.2.2. 辩证处理项目与产品的关系

产品化并不是否定项目的价值和重要性，要看到其对立统一的辩证关系，处理好其相互转化和相互影响的过程。项目在积累到一定时间和数量后，会逐步向产品化进行过渡和转化，而产品在所服务的目标客户和市场发生变化时，也会按照项目的方式进行演进。

#### 3.2.3. 对人员的激励和考核

原有以项目为导向的员工考核指标和激励办法，在进行产品化的过程中变得不能适应新的需求。因此，对销售人员、市场人员、技术研发人员的考核标准都要进行调整，以适应软件产品化的要求。

解决绩效考核和激励机制的问题，具有十分重要的意义。如果在这一点上没有解决好，产品化将是一句空话。当涉及到人的切身利益的时候，如果企业目标和个人利益发生冲突，不可否认的事实是，绝大多数人都会选择后者。

由于产品相对于项目具有较长的生命周期，因此，在绩效考核和激励机制上，既要考虑到近期利益，也要考虑到远期影响。绩效考核指标要包含项目

方面还要包含产品方面，绩效的兑现时间要分为项目结束后和结束后一段时间，包括项目完成情况和对产品化发展的贡献。

### 3.3. 软件技术

在软件产品化的过程中，软件技术水平的发展和提高是一个必然过程。由于软件以项目运作的方式所带来的技术问题，是产品化必须面对和解决的。

#### 3.3.1. 软件技术方案与框架

由于项目型软件开发，开发语言和平台往往五花八门，开发人员在各个项目中疲于奔命。这个项目用 C++，换到另外一个项目组用 Java，同时进行的另一个项目用 ASP.NET，这就是一些软件公司的真实写照。稍微好一点的公司，技术平台相对稳定，例如使用 .net 或 java，但所谓的重用，就是把原有的项目代码拷贝过来，小的改动就修修改改，大的改动就干脆重写，根本上谈不上所谓的产品化。

上述情况，往往就是摆在我们面前的真实写照，是软件产品化必须面对和解决的问题。

##### 3.3.1.1. 稳定技术方案和框架

产品化首要一点，就是要根据产品的特性，选择并建立一套相对稳定的技术方案和框架，并在后续产品研发和项目实施过程中不断演进。要注意到没有一套技术方案和框架能解决所有问题，要根据业务的不同和对技术的要求，选择适合的技术方案和框架。在选择技术方案和框架时，要考虑其可扩展性，能应对不同层次和不同方面的应用要求。当公司扩展产品范围和业务方向时，要对新的产品方向进行系统化评估，以决定是否在现有框架上演进，还是建立新的技术框架和平台。

技术方案的发展，切忌不能受技术高层管理人员的变更带来的影响。不能来一个和尚，念一个经，这种调整，如果没有充分的理由，将是对企业的巨大伤害。

### 3.3.1.2. 技术方案和框架的演进

技术方案和框架并不是一成不变的，而是要在产品研发和项目实施的过程中，不断检验、不断改进、不断提高。

一套技术方案和框架往往在初期，可以适应项目的需要，但随着产品的发展，提出更高的性能要求，提出更大的数据容量，提出更高的处理能力，提出更多的业务需求的时候，往往就不能满足项目要求了。这时候需要对技术方案和框架本身要进行发展和演进。

因此，技术方案和框架的演进，是和软件产品化过程交织在一起的迭代式演进过程，通过产品和项目的实践和反馈，不断提升技术方案和框架的技术能力与成熟度。

由于需要处理的业务问题和技术要求不同，因此存在多套技术方案和框架是很正常的。通过提供不同层次、不同方面的技术方案和框架，来灵活应对复杂多变的业务需求，才是软件产品化的正确道路。在这方面，汽车、通讯等行业有很多成熟的经验可供软件行业学习和借鉴。

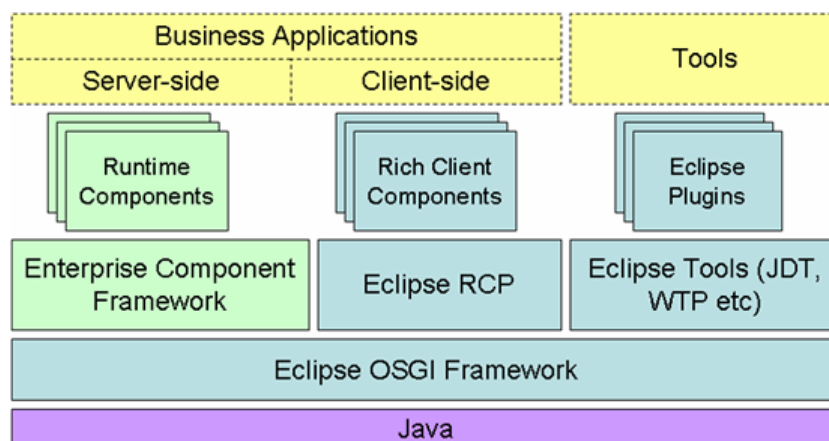
### 3.3.1.3. 技术方案和框架的常见误区

一提到技术方案和框架的狂热，莫过于 Java 语言的应用开发。不论是 J2EE 型的重量级解决方案，还是轻量级的 SSH (Spring + Hibernate + Struts)，好像一说技术方案和框架，无非就是这三板斧。难道技术方案真的就是这样的吗？

技术方案和框架要解决的问题不是具体性业务问题，而是要解决一类通用性问题。因此，技术方案首先要对原有的一系列项目的业务进行分析，分析其共通性，要从具体业务现象中抽象出其本质活动，将需要解决的问题进行归类 and 抽象化。然后，在技术框架的基础上，对框架进行扩展和实现，既保证技术方案的可扩展性，又能满足实际应用开发过程中的个性化需要。

技术方案和框架既具有一定的业务独立性，又同时具有行业性特点。只有真正能和行业特点相结合的技术方案和框架，才有真正的价值和意义。单纯技术层面的技术方案和框架，不是真正意义上的技术框架，而是技术方案和框架的实现方法和细节。

在这一方面，一个很好的一个例子，就是 Eclipse 平台。作为开源的 IDE（集成开发环境）平台，类似于 IBM、Borland、Adobe、Oracle、Bea 等越来越多的开发工具厂商，抛弃原有私有的 IDE，选择 Eclipse 作为其开发工具的平台就是最好的证明。Eclipse 的众多开源项目，给开发 IDE 工具提供了良好基础支撑，而 Eclipse 基于 OSGI 规范的插件化体系结构，也保证了其平台的开放性、可扩展性和先进性。



图表 1 Eclipse OSGi Framework 1

#### 3.3.1.4. 技术传播和转化

在从技术方案与框架到产品化软件再到具体项目实施，这个过程中涉及到技术知识的传播、学习和掌握、以及再反馈的过程。在一般较为正规的产品型软件公司，基础技术的研发、软件产品研发、软件项目实施往往是由不同部门负责，因此，如何有效的将技术、产品在实际项目中进行实施，是软件产品化的关键问题。

在这一技术传播和转化过程中，加强部门的沟通和协作十分重要。这一沟通，不仅仅是指已经做好的技术方案的推广、已经完成的产品的培训工作，而是指在技术方案与框架、软件产品的研发的整个生命周期内，都需要各个部门的参与，形成跨部门的项目型小组进行运作。只有通过全过程的沟通和协作，才能最大程度上消除部门之间的隔阂，充分吸收各个部门的知识，发挥各个部门的特长，减少推行过程中的阻力，而形成一个良好的产品研发环

1 图片来源于 [www.eclipse.org](http://www.eclipse.org) 网站

境，建立有效率的产品研发机制和流程。

### 3.3.2. 行业领域知识

在一个企业开始某个项目，往往是进入到一个新的行业或业务领域，在项目型软件开发过程中，往往是开发方一般没有这方面的专业领域知识，往往是跟着客户学，客户说啥就做啥。很多软件公司都存在忽视业务知识的积累的过程，做了很多项目，但行业专业知识仍然是非常薄弱，往往人员变动和流失，这些知识就不知不觉的流失了，结果到头来，往往还是重新开始。

软件产品化的一个重要标准就是领域知识的积累和完善，这是软件产品化的重要步骤和关键要素。一个公司要想成功完成产品化转变，就要不但让自己成为技术专家，还要让自身成为行业专家。

#### 3.3.2.1. 要比客户还要专业

真正优秀的行业性产品和解决方案提供商，例如 IBM、Oracle、SAP、BPCS、Infor 等厂商，不仅仅只是卖产品，而是通过自身的专业行业知识、帮助客户建立适合他自己的业务解决方案，这是软件产品化的更高阶段，是软件产品化成功的重要标志。

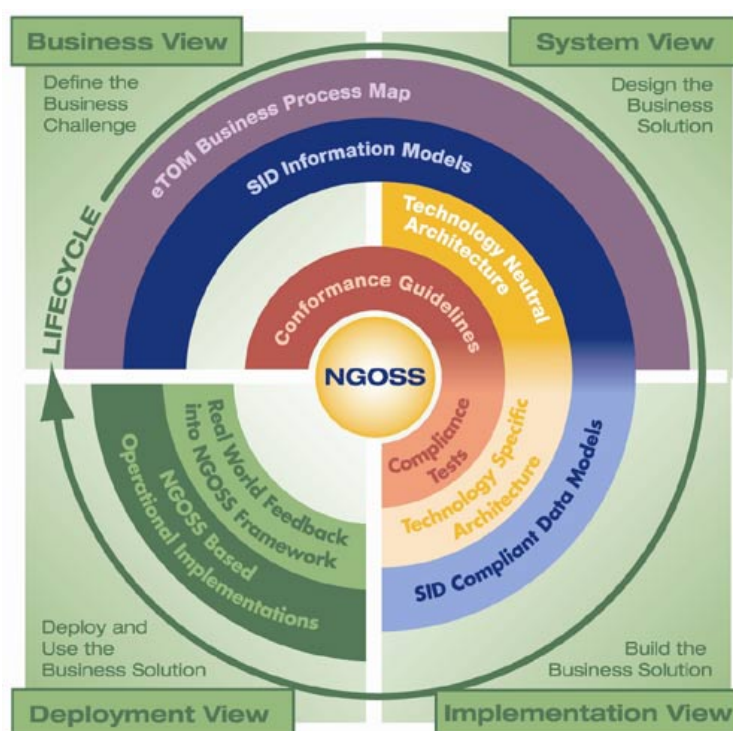
在上述成功企业当中，都有一支专业化的行业专家和业务顾问，帮助客户诊断自身企业问题，提出业务流程改造方案，并结合 IT 技术进行实施，而到达企业信息化的目的。因此，软件产品化的一个重要措施，就是要培养和建立一支懂技术、懂业务的专家队伍，通过不断建立和完善这样一支队伍，真正成为行业性专家，做到比客户还要专业。当然，如果企业不具备建立这样一支行业专家队伍的条件，也可以考虑和一些行业咨询顾问公司进行合作，来弥补自身在业务知识领域的不足。

#### 3.3.2.2. 要基于业务又要高于业务

做行业型产品应用，常常陷入的误区就是就项目而产品，没有跳出客户需求本身来思考问题，造成的结果就是所谓的产品是客户需求的超大合集，往往不成体系，并没有体现其产品的行业先进性。

因此，在做软件产品化的过程中，必须跳出原有项目的视角和局限，站在行业的角度和思考，参考一些国际性规范和标准，并结合实际业务进行实践，这样才能开发出真正优秀和领先的产品化软件。

以电信行业为例，其 IT 系统的建设过程中，这一点是特别突出的。以电信的各类业务运营支撑系统为例，就有“电信管理论坛（TMForum）”提出的下一代运营软件和系统（Next Generation Operation Software and System，简称 NGOSS）。



图表 2 NGOSS下一代运营软件和系统 2

下一代运营软件和系统（NGOSS）是TMF提出的新一代OSS体系。NGOSS从系统（即插即用规则）、过程（企业事务过程模型）、信息（关联处理公用数据）、产品四个方面保证OSS体系具备标准化、能够逐步演化、保证互连互操作（开放）、实现端到端的管理和高度自动化的特点。NGOSS提出一系列的文档、信息模型和代码，分析研究企业核心业务流和信息技术，提出一套指导OSS建设的系统框架和设计即插即用的OSS组件方法，帮助开发商迅速开发支撑系统，满足电信运营商对OSS系统建设的需要，从而使OSS系统设计、开发从满足个别运营商的个体需求到分析电信运营商的整体需求的范围上来，

2 图片来源于 TM Forum 网站



进一步使OSS系统的设计、开发进入到一个崭新时代。贴近运营商需求，使系统开发变得更迅速、更灵活、成本更低是NGOSS的目标。<sup>3</sup>

由此可见，充分吸收行业先进知识和理念，对于项目软件产品化具有十分重要的意义。在具体吸收和借鉴的过程中，不要盲目崇拜，要认真分析和对待这些理念和标准，秉承拿来主义的思想，有选择的进行吸收。

### 3.3.2.3. 要重视对领域知识的管理

对于行业领域知识，一般的软件企业虽然口头重视，但往往没有有效的管理和积累，造成领域知识的混淆、残缺和流失。即使对行业领域知识做了比较好的整理，而软件研发人员对行业知识不感兴趣，只关心技术，了解业务知识，只是为了工作而已，使得领域知识的传播效果很差。

因此，做好对行业领域知识的管理，并建立一支懂技术、懂业务的技术团队，是软件产品化的成功保障。

对于领域知识的积累和整理，需要有专人负责，对行业知识进行系统记录和整理。要建立长期培训和自我学习机制，请内部或外部的行业专家给研发人员进行培训，帮助研发人员系统化提高行业领域知识的水平。要鼓励和奖励知识分享，形成一个良好的学习和讨论氛围。

### 3.3.3. 软件开发过程

产品的生命周期往往要大于项目的生命周期，因此，对于产品化软件开发过程也不同于项目开发过程，需要对项目级别的软件开发过程进行修订和扩展，才能适应产品化软件开发的需要。

#### 3.3.3.1. 产品化开发过程

纵观软件开发过程其发展历程，不论是早期的瀑布开发过程，到后来的V模型，增量与迭代式开发过程，统一过程（UP），直到最新的强调敏捷思想的XP、Scrum、模型驱动开发（MDD）、测试驱动开发（TDD），都是围绕着

---

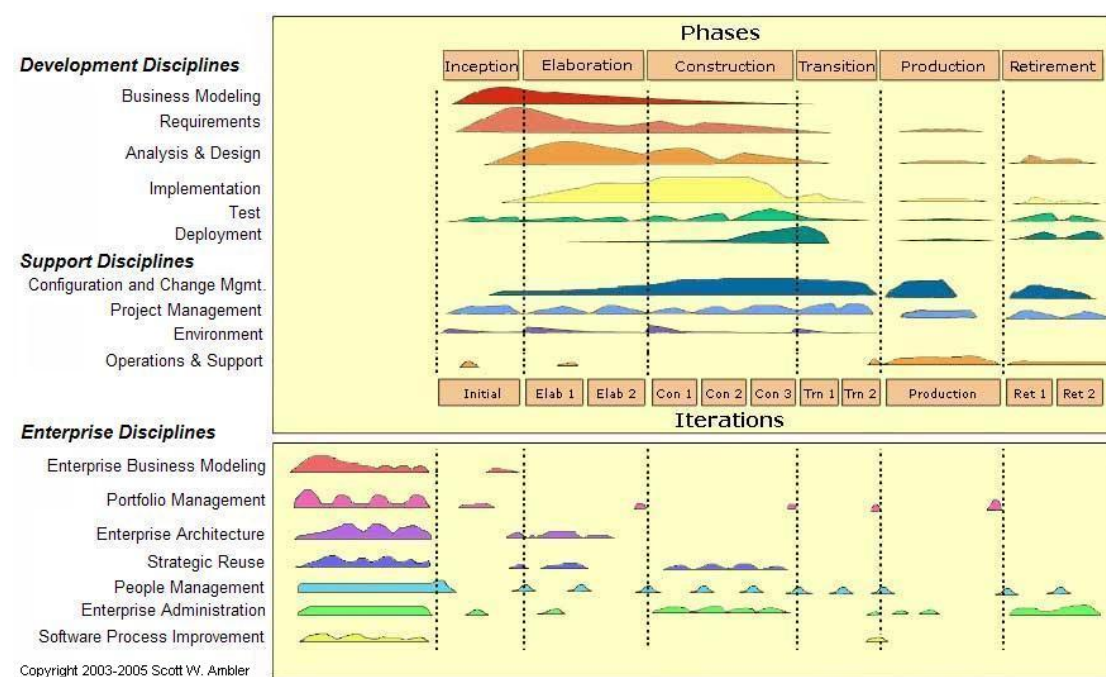
<sup>3</sup> 本段引自中国通讯网《NGOSS在业务支撑系统中的应用》一文。

项目级别的软件开发过程，而关于如何进行产品化开发，几乎没有涉及，缺乏具体的方法和行动指南。

为了弥补软件开发过程在产品化开发方面的缺陷和不足，一些行业专业人士也提出了相应的过程和模型。这中间比较著名和有参考意义的有 EUP（Enterprise Unified Process 企业统一过程）、Zachman 框架等等。下面以 EUP 为例子做一个简单介绍。

Scott Ambler 提出的 EUP 在 RUP 的基础上进行了扩展，对 RUP 的初始、细化、构造、交付四个阶段进行了增补，添加了产品化、退役阶段。对 RUP 的项目级别的活动进行了扩展，补充了项目级别和企业级别的活动。

- 项目级别的活动：运营与支持
- 企业级别的活动：企业业务建模、组合管理、企业架构、战略重用、人员管理、企业管理、软件流程改进



图表 3 EUP企业统一过程 4

EUP 中所涉及到的软件过程与活动，对于将软件项目产品化是很有帮助和意义的，在软件产品化的过程中应该进行吸收和借鉴。

当然 EUP 所涉及的范围对于软件产品化，仍然缺少部分过程，包括企业业务分析、产品设计、产品管理、产品组合管理等过程，需要在企业在产品

4 图片来源 [www.enterpriseunifiedprocess.com](http://www.enterpriseunifiedprocess.com) 网站

化的过程中进一步的完善和补充。

### 3.3.3.2. 软件开发过程选择

随着 XP/Scrum 等敏捷软件开发过程的不断兴起，相对与传统重型软件过程（CMM/CMMI、MSF、RUP 等），越来越多的人进入到敏捷软件开发过程中来。对于进行软件产品化，哪种过程更合适呢？

相对于 CMM/CMMI 等重量级过程，产品化软件开发，应该更多的吸收敏捷软件过程思想和精髓，注重产品的长远规划和近期目标，采用增量迭代的方式，逐步演进的进行开发。

需要注意的是，在产品的生命周期的不同阶段，是应该采用不同的开发过程的。在产品的初创阶段，应倾向采用 XP/Scrum 等更加敏捷的方法，鼓励发散性思维，强调产品的创新性；在产品进入快速发展期的时候，应该倾向采用迭代增量式的开发方法，强调有计划的逐步产出；在产品的稳定期，由于各项需求都比较明确，可以考虑采用传统的瀑布模型或 V 模型；在产品衰退阶段，需要对原有产品进行二次创新，则应鼓励采用 XP 这类极端敏捷的软件过程。

因此，软件产品化开发不存在唯一的、一成不变的软件过程，应该更多的吸收权变理论的思想，根据具体客观情况的不同，灵活掌握和应用。

另外，一般而言，产品的规模越大，牵涉人员越多，牵涉的外部协作方越多，项目的干系人越多，对过程的重量级也要求越高。当然，但这一点并不是绝对的。

以波音公司研发波音 777 客机的过程为例，其研发过程中就抛弃了传统“按图纸生产”过程，而采用同步工程方式<sup>5</sup>，大量使用 CAD/CAM 技术，实现了无纸化生产，试飞一次成功，比传统方法节约时间近 50%。<sup>6</sup>

在设计初期。波音公司和一些航空公司进行了广泛深入的讨论以确定和开

---

<sup>5</sup> 同步工程（concurrent engineering），又称并行工程，是对整个产品开发过程实施同步、一体化设计，促使开发者始终考虑从概念形成直到用后处置的整个产品生命周期内的所有因素（包括质量、成本、进度和用户要求）的一种系统方法。它把目前大多按阶段进行的跨部门（包括供应商和协作单位）的工作尽可能进行同步作业。

<sup>6</sup> 本段和同步工程定义引自 [《汽车研发中的现代化项目管理》](#)一文。

发新飞机的结构布局、这些航空公司包括：美国联合航空公司、全日空航空公司、英国航空公司、日本航空公司和香港国泰航空公司，它们在航线结构客流量和服务频率方面全方位地代表了各航空公司现有的营运水平，均成为了开发 777 时的顾问航空公司。这些航空公司的参与保证了产品最大限度地满足全世界航空公司的需要。为了满足航空公司的要求，波音在 777 上采用了多项新科技。波音公司把这套哲学称之为「一起工作」(Working Together)，因此 777 被称为是波音「以客为本」的产品之一。<sup>7</sup>

可以看出传统制造行业的最先进的产品制造过程，也不再是我们常认为的严格按照阶段进行的瀑布式生产过程，而是引入了和敏捷过程思想有很多共通特点（现场客户、并行工作、可抛弃原型等等）的同步工程方法。反之，我们在设计和研发产品化软件的过程中也要更多的吸收其他行业的成功经验，要将内外部的协同、协作的思想贯穿到底。

### 3.3.4. 软件开发组织机构

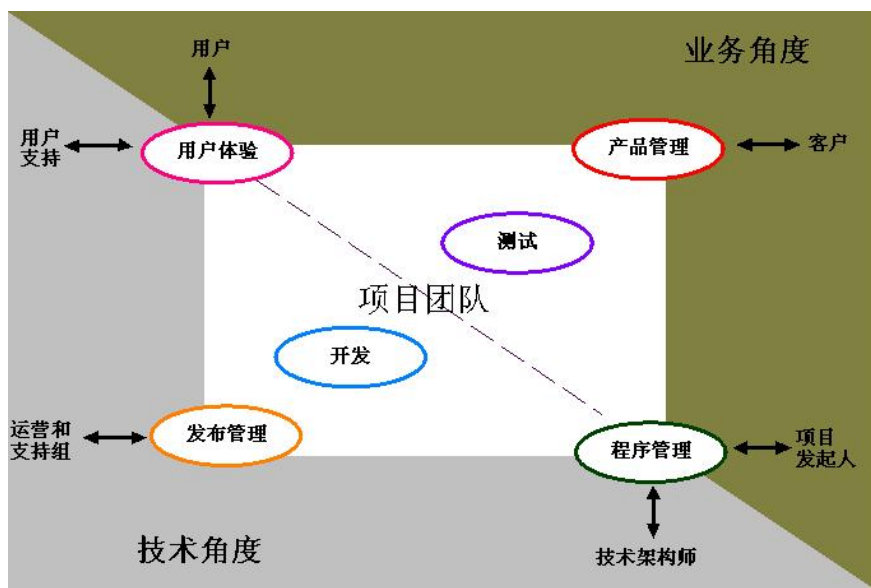
产品化软件开发与商业化套装软件开发的最大不同，产品化软件开发必须同时处理项目和产品之间的关系，而商业化套装软件的开发基本上只考虑产品本身。由于产品化软件的销售基本上是以解决方案的表现形式，即既包括可以重复使用的软件产品，又包括客户化定制部分。因此，如何组织产品及项目的开发，并协调两者之间的关系，就是一件非常困难和非常重要的工作。

由于产品型软件开发，基础技术、产品研发、项目实施、产品管理常常分属不同的部门，而产品研发或项目实施都需要不同部门的人员共同参与配合，因此采用传统集权式或层峰结构组织，都会造成部门割裂，官僚作风的产生。

针对产品型软件开发，应该采用矩阵式管理或更为灵活的自组织团队的管理方式，消除部门壁垒，加强人员沟通。在这方面可以借鉴微软的 MSF 软件过程。MSF 软件过程的组织结构采用以特性 (Feature) 小组方式组成，每个小组包含产品管理、程序管理、开发、测试、用户体验以及发布管理等六个角色。

---

<sup>7</sup> 本段引自百度百科词条：[波音 777](#)



图表 4 MSF项目团队和外部环境的沟通 8

针对产品研发和项目实施，也建议采用跨部门的多职能小组，根据是产品还是项目，采用偏重产品或偏重项目的管理方法进行管理。

### 3.3.5. 软件生产工业化

项目型软件向产品化发展，不仅仅是指软件本身的发展，更为重要的一点，就是软件本身的生产过程，要向工业化的生产水平的发展，而一点常常被人们所忽视。如果软件本身的生产过程和技术水平仍然停留在单个项目的水平，那就算是你同类项目做了几十个，并把他们的需求整理成一套软件，完成客户的需求也相对变快，这并称不上产品，只不过是像欧阳修的所描写的卖油翁一样，“我亦无他，惟手熟尔”。

因此，软件的产品化不仅仅意味着软件产品的可复用，而是软件的生产过程也要进化到工业化产品生产水平。通过技术上的创新，解放开发人员，将其从简单重复性的工作中解放出来，来处理更高级的和更关键的问题。

通过将软件生产提升到工业化生产水平，将大幅提高软件生产效率，提高软件产品质量，形成规模化经济效益。

8 图片引自edmond\_wang 博客《[MSF团队模型](#)》

### 3.3.5.1. 传统工业化内容

传统工业化内容的定义，主要包括劳动资料、工业管理、劳动者和管理者的知识结构、工业部门结构、主要技术经济指标等方面的内容。

劳动资料的现代化。在主要工业部门实现生产过程的机械化、自动化，采用新技术、新材料、新工艺和最新科技成果。

工业管理的现代化。在工业管理中，采用现代化的管理体制、合理化的生产组织、科学化的管理方法和电子计算机化的管理手段，达到优化资源配置，取得最佳经济效益的目的。

劳动者和管理者知识结构的现代化。劳动者和管理者的科学文化水平和专业知识普遍提高，在业职工构成发生变化，即在整个工业人员中，技术人员的比重将不断上升，直接生产工人的比重将逐渐下降；在工业管理人员中，具有专业知识的人员会迅速增加，而非专家职业的人数会减少。

工业部门结构的现代化。技术密集型工业的比重日益提高、新兴工业部门的建立与发展，是工业部门结构高度化的主要标志。

主要技术经济指标达到当代世界先进水平。例如，主要工业产品产量、劳动生产率、主要原料和能源的消耗水平、资金占用、先进机器设备的自给率等一些指标，都应该达到当代世界先进水平。

劳动资料的现代化是工业化的核心，工业管理的现代化与劳动者和管理者知识结构的现代化，是实现劳动资料现代化的基础，工业部门结构的现代化<sup>9</sup>和主要技术经济指标达到当代世界先进水平，是工业化在结构上和功能上的综合反映。<sup>10</sup>

### 3.3.5.2. 软件生产工业化

参考传统工业化内容范畴，可以进一步得出软件工业化的重要标志，即研发方法的现代化、研发管理的现代化、研发人员和管理人员知识结构的现代化、研发部门组织结构的现代化和主要产品研发技术指标的达到世界先进水平

---

<sup>9</sup> 作者注：原文为“高度化”，应理解为“高度结构化”。为和前文统一，改为“现代化”。

<sup>10</sup> 本段内容引自[智库百科](#)词条：[工业化](#)

平。

研发方法的现代化。主要包括软件生产的自动化和智能化，采用最新的研发工具、测试技术与工具、产品技术框架、开发方法、建模语言、编程语言等等。

研发管理的现代化。在研发管理中，采用现代化的软件过程与思想，科学组织研发管理团队，使用信息化的管理工具手段，人性化的管理，并为员工提供良好的学习环境和成长空间，从而达到尽可能大的发挥研发团队能力，激发员工的主动性，创造更大的经济价值。

研发人员和管理人员知识结构的现代化。研发人员和管理人员的专业知识、研发能力、管理水平普遍提高，人才结构发生变化，及高素质人才、专家、通才在整个员工比例中不断上升；而基础的编程、测试、维护人员数量逐步下降。

研发部门组织结构的现代化。从单一的研发部门，到专业化的部门分工，例如基础技术研发部门、产品研发部门、项目实施部门、产品发展与创新部门、质量控制部门、质量保证部门、项目管理办公室、流程改进工作组等都。专业化部门分工，将标志着软件研发部门组织结构向现代化的迈进。

主要产品研发技术指标的达到世界先进水平。例如缺陷数/千行代码、代码产出率、无故障运行时间等，都应该达到当代世界先进水平。

研发方法的现代化是软件生产工业化的核心，研发管理的现代化与研发人员和管理人员知识结构的现代化，是实现研发方法现代化的基础，研发部门组织结构的现代化和主要产品研发技术指标的达到世界先进水平，是软件生产工业化在结构上和功能上的综合反映。

### 3.3.5.3. 实现软件生产的自动化

软件产品生产工业化的核心就是研发方法的现代化。借鉴工业化的发展的历程，其劳动资料的现代化的核心体现就是生产过程的机械化、自动化。那么针对已经高科技行业的软件行业其“机械化、自动化”又体现在哪里呢？

仔细分析软件产品的生产过程可以看到，虽然软件行业属于高科技行业，但其生产过程，从需求分析、设计、编码、测试、配置管理、部署、项目管

理等诸多活动，可以看到其大部分的工作仍然是手工作业为主，虽然有很多工具在这过程中进行使用，但其核心生产活动（分析、设计、编码、测试）绝大部分仍然是手工作业，而更加关键的部分是这些活动之间是相互独立的，很少有自动化的过程。因此，相比较汽车、飞机、计算机硬件生产等高度发达的制造行业，其机械自动化程度、柔性生产能力、无人操作流水线水平，软件行业的自动化生产技术和能力还处在相当原始的水平。

因此，要实现研发方法的现代化，其核心就是要解决如何让研发的各个过程能够无缝的进行衔接，既能提高自动化水平，又有足够的灵活度，达到柔性生产的目的。

为了达到这个目的，不少软件开发工具提供商也在这方面做了很多的探索和实践。从目前最新的进展来看，MDA、DSL 将是实现这以目标的重要发展方向。MDA 以 UML 为核心，采用自顶向下的方式来实现软件开发的自动化；而 DSL 则是以基于一定宿主语言的领域特定语言为核心，由下而上来逐步实现从分析到编码的无缝连接。

另外，一些厂商在一些特定领域，将自动化开发过程推进到更为实用的阶段，并提供了相对完整的开发环境。例如 Eclipse 的 EMF+GMF+GEF 和 DevExpress 的 eXpressApp Framework。他们共同特定是，建立了一套可以自动化从领域模型到可执行代码双向转化的机制，可以较好的实现自动化生产的目的。

通过实现软件生产的自动化，将很大程度上降低具体应用开发的门槛，从而让软件生产的成本下降，质量提高，而开发人员则被解放去进行更加富有创造力和有价值的开发和设计工作去。

### 3.3.6. 软件产品质量

软件质量低下，缺陷数量多，软件界面粗制滥造，没有良好的用户帮助文档，软件功能和实际需求不一致，软件的性能、稳定性、可扩展性差……这些问题在项目性软件开发过程中常常见到。当然，并不是说产品化软件就不存在这些问题，但产品化软件做的相对较好的软件企业，上述问题出现的数量和程度要小的很多。而项目型软件的开发，往往受开发时间期限以及人



员素质和数量的限制，往往很难将上述问题都能做好。

因此，软件产品化不仅仅是软件功能的丰富和完善，软件质量的提高是衡量软件产品化是否成功的重要检验标志。

产品的质量保障，重点在产品的软件的验证与确认技术(即通常所说的 Validation & Verification 认证，简称 V&V)。而软件的产品化，从质量角度，就是要在验证和确认技术上提高其技术水平。V&V 的方法和手段，从产品化软件开发角度来看，重点要在客户全程参与、人机交互体验、业务场景模拟实验室、自动化技术测试实验室等方面进行重点提高。

### 3.3.6.1. 客户全程参与

客户全程参与，是指拥有实际业务知识和经验的客户参与到整个开发过程中来，这一点在前面波音 777 飞机制造过程中所倡导的“Working Together”理念是一致的，和极限编程（XP）所倡导的“现场客户”也是相通的。

首先，产品化软件开发，必须建立一支有实际操作经验和业务知识的业务团队，不论是邀请客户的相关人员进入团队，还是自行招聘相关业务人员。这支团队是保证产品符合客户和用户的真实需求的重要保证，也是研发人员和客户沟通的重要桥梁。

其次，这支业务团队，不是像在项目型软件开发过程中常见到的，只是在需求调研之初和产品验收之时才出现，而是要参与产品的整个研发流程，持续性的与研发团队进行业务交流和反馈，对研发过程中的产出物进行评估和测试，及时将信息反馈给研发团队。

因此，客户的全程参与，对于保证产品能最大幅度的贴合实际用户需求，具有举足轻重的作用。

### 3.3.6.2. 人机交互体验

人机交互体验过程，往往是被软件研发组织所忽视的部分，极少有公司（特别是国内企业）有设立专门的用户体验部门，并具有专业人机交互体验设计知识和经验的设计人员和工程人员。在绝大多数的项目型软件中，可以毫不客气的说，根本就没有人机交互设计，人机交互设计的好坏，只是凭着开发

人员和美工的个人感觉和个人能力。

优秀的产品化软件，在人机交互体验上往往做的相对较好，其用户界面的友好性、操作的便捷性、用户帮助系统的完备性上都做的不错。

因此，发展产品化软件，不可忽视的一点就是要在软件的人机交互体验上大力提高自身水平，建立一支专业化的设计团队，这是体现产品化软件质量的一个重要方面。

### 3.3.6.3. 业务场景模拟实验室

传统项目型软件的业务功能测试，基本上都是靠测试人员依据测试脚本进行业务功能的测试，但当软件所设计的业务越来越复杂，这种测试越来越体现出其脆弱和不足。

首先，由于测试人员本身对业务和技术两方面都比较欠缺，其编写的业务测试脚本根本无法覆盖到业务的真实场景情况，很多复杂业务情况、异常业务处理流程，在需求调研的时候客户根本不会有意识的进行系统总结，需求分析人员自然也不得而知。实际的情况往往是挤牙膏式，只有你实际问到了，用户才会把问题提出来。因此，测试人员和开发人员在软件开发的过程中根本不会知道这些。

其次，就算是了解和分析到各种业务场景和流程，如何有效的进行反复测试，是一个十分棘手的问题。因为，某一特殊业务场景，往往是在十分特殊的条件下才会产生，而要模拟这一场景，需要准备的数据是相当大的，特别是对于一些具有复杂业务的应用型软件。

因此，必须建立一个专门的业务场景模拟实验室，并配备相应的业务人员、测试人员和技术人员，并开发相关的业务场景模拟软件，自动化的模拟产生相关场景数据，才能有效的在研发阶段，在实验室环境中模拟真实的业务运行过程。虽然建立这样一个专门的实验室，是需要有相当程度的人力投入和资金投入的，但这也是提高软件质量的极为重要的保证，这一点对中大型软件公司，非常重要。

### 3.3.6.4. 自动化技术测试实验室

对于产品化软件，特别是大型产品化软件，为了测试产品的性能、负载、稳定性等技术指标，如果不采用自动化技术测试手段，仅仅靠人工，根本无法进行测试。

因此，针对一些需要专业化技术测试的领域，发展自动化测试技术，对于产品化软件开发，是十分必要的，在一定场合下，甚至是评判软件产品能否合格的重要保障。

自动化技术测试实验室和前面基于场景的业务测试实验室一样，需要建立专门的测试队伍，并配备相关的测试人员和技术人员。

需要注意一点的是，由于软件本身的复杂性和特殊性，市面上通用的软件技术型测试工具并不能完全满足测试要求，在很大程度上，需要开发专用的测试平台和工具。这一点在大型软件产品研发过程中，是十分必要的。

## 3.4. 软件人才

软件研发其核心是人，拥有优秀的开发人员和管理人员，是实现优秀软件产品的基础保证。而众所周知的是，研发人员的流动性是相当高的，这在项目型软件开发中特别常见。往往一个项目快要结束的时候，也是人员流失最频繁的时候，往往项目一结束，人也走了一半了，特别是优秀的软件研发人员和管理人员。而产品化软件的开发，其产品生命周期要比项目长得多，如何保持软件研发队伍的研发与管理能力处于一个稳定且逐步上升的过程中呢？当一个产品下市以后，如何对现有研发团队进行调整，避免人才的流失呢？如何使得开发人员在产品多个版本的研发过程中，持续保持激情和动力呢？这都是产品化软件研发，需要面对和解决的问题。

在产品化软件开发的要求下，需要解决的问题很多，下面重点谈一下人才的结构问题、和对关键人员的依赖问题。

### 3.4.1. 人才结构问题

产品化软件开发，很重要的一点，就是要合理配置人才结构。这一点看似

简单，但实际上往往被人们所忽视。软件生产现代化的一个重要特征就是研发部门组织结构的现代化，这是软件生产工业化在结构上的重要反映。而实现研发部门组织结构化，就涉及到软件行业人员专业分工，需要不同岗位的专业从业人员。这个在项目型软件开发当中，开发人员占核心位置是有着本质区别的。

因此，有效的配备各个软件岗位的人员，特别是产品方面、业务方面、质量方面、管理方面的人员是实现软件产品化发展的基础保证。

### 3.4.2. 对关键人员的依赖问题

项目型软件开发，最为常见的问题是，对核心人员的过分依赖。由于项目型开发在时间和人员上的约束，存在的现实状况往往是，一个团队中，真正熟练和有能力强的人员（技术熟练，有创新精神，能解决关键性问题）只是少数几个，而大部分人员都是相对中等（工具和语言较为熟练，但思维能力和钻研精神平平）甚至初级的（只是对开发语言和工具只有基本了解，但实战能力和理论基础都很薄弱）。这样情况在很多研发项目中都是普遍存在的，而且也是在很长一段时期内无法缓解的。如果关键人员发生变动，整个团队将受到很大影响，团队战斗力和士气将会急剧下降。

在从项目型软件开发转变到产品化软件开发过程中，由于产品化开发对研发人员的素质提出了更高的要求，因此人才的矛盾问题就越发突出。如何解决既要吸引关键性人才，又要减少对关键性人才的依赖问题？

借鉴传统工业化的发展历程，从小作坊式手工制品到大规模机械化生产的转变可以看出，以往需要具有十多年经验的专家才能掌握制作出的精美物品，通过工业化过程，只需要简单培训过的操作人员就可以大批生产出同等质量的产品。这个过程中，可以看到，将专家型人员的知识、技能、经验，尽可能转化为机械化、自动化的生产机器和工具，是解决对关键性专家人才的依赖的根本办法。

因此，要通过技术方案和框架的实施、通过自动化的开发工具，将复杂的开发过程，转化为简单的操作，将关键人员的知识和经验，不仅仅是通过文档保存下来，更重要的是转化为技术方案和框架、领域知识、自动化工具，才

---

可以真正解决对关键性人员的依赖问题，真正进入到软件行业工业化制造的时代。

### 3.5. 本章小结

本章，通过对企业经营、企业管理、软件技术、软件人才等四方面的论述，详细阐述了在项目软件产品化过程中所需要关注的各个方面、所需要解决的问题和一些解决思路与方法。

可以看到，项目软件产品化不是一个简单的技术性问题，在这过程中将碰到方方面面的问题。如果一个企业在这方面认识不够，就想当然的项目做得多了就产品化，是要吃大亏的。

另外，企业应该看到，在产品化的过程中，虽然产品化带来的好处是巨大的，但需要花费的资源 and 成本也是相当大的，而且也是存在着一定的风险的。在这过程中，企业应该做好充分的思想准备，并做好资金和人才储备，并要容忍在产品化过程中所走的弯路和犯的错误，才能真正穿过这条荆棘之路，到达光明的前方。

---

## 4. 案例分析

---

本章通过对 A 物流公司内部运营管理系统项目开发及产品化过程进行讲述和分析，便于读者更好的理解项目软件产品化的过程，并了解到其中所面临的困难和问题。本案例在项目软件产品化过程中，做了很多有益的实践和探索，希望能给读者一定启发，在解决自身具体问题上有一定的帮助作用。

本案例取材于真实项目，为了避免涉及企业敏感信息，对项目涉及的公司、项目、人员都隐去了其真实名称。项目所涉及到活动都是真实记录项目的真实情况，所提供的数据都是根据项目真实统计数据所得出的结论。

### 4.1. 案例背景

A 物流公司（以下简称 A 公司）是国内一家大型物流运输企业，在行业内属于前 3 名的领先地位。面对物流行业的残酷竞争，为了提高企业的竞争能力，早在 90 年代就开始进行尝试信息技术在企业中的应用。通过多年的不断的信息化建设，其企业信息化水平已经有了长足进步。为了进一步提高企业管理水平，加速业务服务速度，提高物流服务质量，公司组织了近三十人的研发队伍开发了新一代物流运营管理系统。通过历时近 1 年半的开发，形成了新一代的物流运营与管理系统（以下简称 L 物流软件），涉及营业、运输、财务、管理、质量、运作、企业决策分析等各个方面，极大提高了运营速度和管理效率。

作为 A 物流公司，建立自己的一支研发队伍，其目的不仅仅是要为内部服务，更为重要的目的，是要进入到软件研发行业，为达到整个集团立体化发展做战略布局。

在这个过程中，所涉及到的问题就是如何完成公司的战略定位，在软件行业生存和发展的的问题，以及如何将供内部使用的物流管理软件产品化，进入软件行业市场的问题。

## 4.2. 企业经营与发展问题

为了更好的进行软件研发和管理，在内部系统开发的阶段，A 物流公司就成立了一家 B 软件公司（以下简称 B 公司），将原有研发人员全部转入新的公司进行运作。

作为 B 公司，如何确定自身的发展目标，是首先摆在 B 企业面前的首要问题。这其中有两个关键性的问题需要解决。

其一，如果将供 A 公司企业内部使用的 L 物流软件，进行产品调整后直接进行销售，就等于给 A 公司培养竞争对手，因为 A 公司已经是行业内的领先企业了，业内很多公司都是以 A 公司为标杆。但如果不把 L 物流软件进行市场化，B 公司因为没有成熟的系统和成功的案例，则在激烈的市场竞争上，很难打入其市场。

这就让 B 公司进入了一个两难的境地，该怎么办？如何让自身发展建立在 A 公司的优势之上，却又不和 A 公司的利益发生冲突，并且和 A 公司实现双赢的效果呢？

其二，B 企业本身属于初创阶段，虽然有 A 企业的资金支撑，但想实现快速发展，并在行业中站稳脚跟，是一件非常困难的事情，没有客户、没有市场，技术人员队伍还比较弱小，如何发展？

### 4.2.1. 企业发展战略

B 公司根据自身的特点，最终确定如下的三大发展战略：以制造业企业为目标客户，重点发展企业供应链管理系统（SCM）；大力发展电子商务技术，为企业客户建立具体自身特点的电子商务平台；打造专业化物流软件，为第三方物流企业提供专业化物流方案。

通过上述三个方面的业务，帮助客户达到商流、物流、信息流、资金流的四位一体式管理，形成立体化业务发展。在 B 企业发展过程中，将充分借助于 A 企业在物流行业的原有优势，并通过自身业务的发展，反过来进一步促进 A 企业的业务发展，从而达到双赢的目的。

需要清楚的是，这三方面的业务不是孤立的，而是相互联系，相互促进的。

通过三方面的业务有效配合，从而形成 1+1+1 远远大于 3 的效果。

当 B 企业发展到一定规模之后，不但是靠 A 企业来帮助 B 企业的发展，更进一步，B 企业为 A 企业的发展，从技术、市场、客户等带来巨大的支撑和帮助。

B 企业的发展，如果只是靠其自身的发展和积累，很难生存和发展。通过与 A 公司的有效协作，其整个集团企业发展不仅仅是做加法的过程，而是做乘法，这样才会给整个集团发展带来质的飞跃。

#### 4.2.2. 供应链管理系统

首先，通过将原有 A 企业的客户进行分析，选取优质企业，进行业务发展。因为这些客户都是 A 企业的重点客户，对 A 企业有业务依赖和认同感，对于 B 企业开展业务会有很大的便利性。选取 SCM 作为企业应用的切入点，是因为，企业供应链管理和第三方物流都属于物流范畴，具有很多的相似性和相关性，因此，可以充分利用 A 公司的业务人员的专业知识，为 B 公司打开该市场提供帮助。另外由于其业务的相似性，原有 L 物流系统的很多模块，特别是领域模型部分，都是可以进行复用，或在其基础上进一步进行发展。由于客户来源、业务知识、软件系统都可以从原有资源中进行重用，因此 SCM 系统是 B 公司发展的良好选择。

由于这些客户都是 A 公司的客户，通过将 SCM 系统和 L 物流系统进行互联互通，通过电子数据交换平台进行数据交换，可以将 SCM 和 L 物流系统联成一个整体。这样既提高了 SCM 产品的竞争的同时，还为客户提供了更为透明的供应链管理，提高了供应链的管理能力，也将客户有效的锁定到 A 公司，减少了客户流失。更进一步，通过由于供应链管理的需要，将客户的供应商也进一步纳入到 A 公司的物流通道中来，进一步提升了 A 公司的业务量。通过为客户建立 SCM 系统，可以进一步将其供应商发展成 B 公司的 SCM 产品的客户，从而形成滚雪球效应，这样就逐渐打开了 B 公司 SCM 系统的市场。

综上所述，通过上述业务发展战略，可以有效的形成企业发展的正反馈效应，形成良好的企业发展生态环境，从而到达双赢的结果。SCM 系统将是 B 企业在企业级市场发展的重要战略发展目标。



### 4.2.3. 电子商务平台

A 公司现有代收货款业务已经通过自身的发展，已经具有相当的规模，月代收货款交易金额已经过亿元，具有非常广泛的客户群体。

针对这一特点，B 公司将结合 A 公司的业务，重点发展电子支付、网上支付和电子商务平台。

通过发展电子支付、网上支付，可以将 A 公司现有的代收货款业务的实际运营，纳入到 B 公司的业务范畴中。并且由于采用了网上支付和电子支付，可以进一步提高 A 公司的服务范围和质量，并能够进一步提高资金安全，降低人工成本。

在为客户提供代收货款的业务同时，进一步为企业自主拥有的个性化的电子商务平台，提供产品展示、网上订单、网上支付、物流服务、订单跟踪与管理等一体化集成服务。这一服务和现有的阿里巴巴、淘宝、eBay 等著名电子商务网站有着很大的不同。

因为，以 B 公司的实力，即使有 A 公司的市场和资金的支持，如果建立和阿里巴巴、淘宝网同质竞争形式的电子商务交易平台，基本上是以卵击石。不论是从资金、市场、人才、技术上，B 公司和这些行业领先者都不在一个数量级上，因此必须发挥蓝海战略的思想，突破现有的业务模式，才可能有突破性的发展。

现有的电子商务平台，基本上是借鉴超市模式（这也是为什么淘宝网把沃尔玛作为其最大的竞争对手）。超市模式的特点是，商家实现电子商务的门槛很低，可以很快享受到电子商务的好处，而其不足也在于，其电子商务平台同质化严重，没有企业特色，无法体现其品牌特点。因为，当企业发展到一定规模以后，其品牌意识逐步增强，这样建立专属的电子商务平台就成为其潜在的用户需求。这一点，从阿里巴巴、淘宝网的客户群体就可以看出，其大部分的客户都是中小型企业和个人。一些有实力的企业和个人，往往都有自己的网站，但其水平、功能、技术能力都有很大的局限，因为他们往往都不是专业从事软件开发的，没有能力靠自身力量建立这样的系统。他们不可能以自身实力，建立类似 Dell 直销网站式的专业化电子商务服务平台。

因此，基于前面的市场分析，B 公司采取的是品牌店模式。通过为其客户

建立其专属的电子商务平台，着重其品牌推广，并提供完整的电子商务服务功能，包括网上商场、电子支付、物流服务与跟踪、订单管理、采购管理等，为客户提供了一个更加专业、个性化和完善的电子商务服务平台。

而建立这样的系统，对于 B 公司而言，正是其长项。通过和国外的 Magento、国内的 ShopEx 等网上商店软件企业联合，可以快速的建立专业化的电子商务网站，在其基础上进一步扩展出完整的电子商务平台。通过将客户的电子商务平台和 A 公司的 L 物流系统进行对接，可以进一步将客户转化并锁定成 A 企业的客户，也为 A 企业发展运输、快递、仓储、配送等多方面、多层次的物流综合服务提供了发展契机。

因此，在电子商务方面的发展，是实现 B 企业快速发展的重要战略目标。

#### 4.2.4. 专业化物流软件

为 A 公司提供长期的物流软件的开发和服务，是 B 公司的基本责任之一。通过给 A 公司提供物流软件的同时，也积累了丰富的物流软件行业知识和专业技术人才。因此，将这些知识、经验、技术进行社会化服务，形成产品化软件，是 B 公司发展的重要方面。

在对 L 物流软件开发之初，就已经对未来的发展就有了前面的战略考虑，因此，在 L 物流软件的开发上，从技术层面和业务层面就已经进行了相关的准备，这一点在下面的部分会具体展开阐述。因此，L 物流软件的产品化在技术和人才方面应该是有一个比较好的基础的。

在产品化的过程中，需要解决如何为内进行项目化的服务和对外进行产品化推广的矛盾。

在产品化 L 物流软件的过程中，如前文所述，不可避免的存在与 A 公司发展所涉及的矛盾和冲突问题。因为从 A 公司企业利益的角度上来看，L 物流软件是其多年运营和管理经验的结晶，是 A 企业在其行业内能成为领军企业的核心竞争能力，是不希望被人家所拷贝和复制的。这一点是和 B 公司将 L 物流软件市场化目标相矛盾的。

另外，由于优秀技术人才的缺乏，特别是具有综合素质的技术人才的缺乏，是在很长一段时期内都存在和不可避免的，因此，如何将有限的人员合理安

排，协调在对外进行产品服务的同时兼顾对内的项目研发，是一件重要的事情。如果不能解决这个问题，B 企业由于外部市场的压力和产品与项目的进度压力，往往会削弱企业对 A 公司项目开发在人力和精力上的投入和支持力度。

因此，为了解决上述两方面的问题，B 公司在以下方面做出努力。

首先，充分利用在为 A 公司开发 L 物流软件所获得的业务知识与积累的技术能力，沿着 A 公司主营业务及其附属业务的发展，有选择的进入对应的行业市场。A 公司所处的物流领域目前主要集中在运输行业，可以看到，从物流行业的细分市场上看，还要国际运输、快递、仓储、配送等很多尚未进入的市场。而这些市场不但是 B 公司可以进入和发展的市场空间，同时也是 A 公司为了自身发展的需要，可以进一步进入的物流服务领域。

因此结合 A 公司本身的发展需要，B 公司确定了专项物流、航空、快递为其重点发展的领域。这些领域的业务知识和原有 A 公司的业务具有一定的相似性，B 公司的业务人员和技术人员可以快速学习和掌握，而且其业务系统也具有相似性，可以充分利用原有在 L 物流软件积累的技术和功能。更为重要的一点，A 公司在这些领域也正在进行开拓和尝试，B 公司的产品化开发天然就拥有了一个忠实客户，可以和 A 公司的发展一同进行发展。另外，B 公司在为外部客户提供产品服务的同时，也可以充分了解到行业内其他客户的一些宝贵经验和优势。而这些内容，通过为 A 公司在 L 物流软件的基础上建立新的业务系统得以体现，对于为 A 公司开展其新业务，也提供了直接的帮助。

其次，为了保证 B 公司对 A 公司的持续投入，A 公司每年也支付一定数量的软件服务费，将 A 公司的系统建设外包给 B 公司，而 B 公司也组织一支长期的技术队伍为其服务。这样就从机制上解决对内服务和对外服务的矛盾，利于 A 公司和 B 公司的长期发展和相互协作。

#### 4.2.5. 小结

综上所述，可以看到通过对 B 公司业务战略规划和市场分析，明确了 B 公司的发展方向和道路，为 B 公司的软件产品化发展指明了方向，也奠定了

良好的基础。也可以看到，软件项目产品的市场化和产品化，不是一件简单的事情，需要综合考虑各方面的因素，认清自身的优势和不足，才能选择好一条适合自身发展的道路。软件项目产品化的第一个部分，就是要从企业经营部分，解决企业经营发展的问题，这样才能为软件项目产品化的成功建立基础保证。

### 4.3. 企业管理问题

为了实现软件项目产品化的目标，A 公司从基本经营上将这一块的业务独立出来，成立了 B 公司,是非常关键的一步，为 B 公司今后的经营管理打下了一个良好基础。

#### 4.3.1. 企业文化

由于 A 公司是一家具有十多年历史的物流企业，因此形成了自身独特的企业文化。B 公司是属于一家高科技行业公司，而由于行业性的不同，不可避免的是，需要对原有的企业文化进行有选择的扬弃，才能适应 B 企业自身的发展需要。

通过以下几个具体措施，为保障 B 企业的企业文化的确立起到良好的帮助。

首先，企业高层领导的支持。为了 B 企业的发展，需要让 A 企业集团的领导理解 B 企业发展的所必须拥有的企业文化非常重要的。公司集团高层的理解和支持是 B 公司良好发展的具体保证。

由于 B 企业是一家高科技企业公司，因此，决定了其企业文化所倡导是人性化管理、灵活敏捷的管理方法和制度、对员工的流动和新陈代谢有良好的心态、重视对员工的培训、强调学习型组织和自组织团队等等。这些都是和 A 公司的传统物流行业是很不一样的。

其次，在办公地点和环境上的投入。原 A 公司所在地点，由于物流行业的特点，选择比较偏远的城乡结合部，具有良好的交通便利，但由于其办公地点的偏远，无法吸引优秀的高科技的优秀人才在其公司长期工作。

为了解决这一问题，A 公司将 B 公司的办公地点迁至市中心附近的写字

楼，使得 B 公司有一个相对独立的空间，这样就解决了一个基本的办公地点和环境问题。

### 4.3.2. 组织结构问题

为了适应 B 公司的发展战略目标，需要建立完善和健全的组织结构。B 公司原有的人员主要是技术人员为主，因此，需要在产品、市场、销售、售后服务等方面，完善其组织结构，并建立和发挥相关功能。

### 4.3.3. 业务发展

根据企业发展的三个重要战略方向，根据其业务发展的先后，在业务方面，确定了首先突破企业 SCM 市场，同时快速建立电子商务平台，伺机开展专业化物流软件领域的战术。

在技术投入上，以 SCM 为重心，因为其业务复杂度和技术复杂度是最高的，但同时也是技术壁垒最高的；电子商务系统部分，可以充分整合现有的资源，通过利用外部的一些成熟产品和技术，达到自身的快速发展。例如网上商场部分，可以使用目前最新的最好的开源的网上商店系统（Magento）或国内成熟的 ShopEx 平台；电子支付部分，在 L 物流系统已经和多家银行建立了网上支付系统的互联，因此具有良好的技术基础，并且可以进一步引入第三方支付系统，从而达到快速发展的目的。电子商务的后台运营管理部门，可以充分利用现有 L 物流系统的积累，并借鉴行业内的一些领先企业，可以在短时间内快速发展起来。

在 B 企业发展的初期，将重点挖掘 A 公司的现有客户。因为掌握 A 公司具体的运营数据，整理出占 A 企业业务量前 500 名的客户名单，这些客户都是 B 企业发展的重点客户。对这些客户进行了行业性分类和地区分类，整理出前 10 个重点行业范围，并重点挑选出围绕着北京、上海、深圳等重点城市及其周边地区城市的企业。通过这一整理，基本确定了约 200 家重点企业。

这 200 家企业是 B 企业重点的业务发展对象，对 B 企业的初期及以后的发展具有重要意义。

#### 4.3.4. 技术管理

原有 L 物流软件开发的约三十人的团队，都转入了 B 公司的研发部门。这些人员是 B 公司技术发展的宝贵资产和力量。

在 L 物流软件开发和建设的过程中，就考虑到未来的技术发展需要，从而建立了一支相对完善的技术与管理团队。在这支团队中，包含系统构架、业务分析、软件开发、软件测试、业务团队、质量保障、项目管理等多方面的专业技术人才。

### 4.4. 软件技术

#### 4.4.1. 软件技术方案和框架

在软件技术方案和框架上，在 L 物流软件设计之初，就已经考虑到其以后的可扩展性和可维护性，在技术平台的选择和技术方案的确立上进行了反复对比和研究，为以后的长远发展打下坚实基础。

在研发平台的选择上，选择了 Java 为其技术平台。作为企业级应用，Java 在企业级应用上具有明显优势，人才和技术资源都是相当丰富的。

由于物流企业应用软件具有广泛的跨地域特点，地点相对偏远、作业实时性要求高的要求，因此采用了分布和集中并存的方式，以满足复杂的业务需求和不足的网络环境。

在客户端系统上，为了满足其操作的方便、快捷和复杂性需求，又同时考虑到系统的可扩展性，选择 Eclipse RCP 平台(Rich Client Platform)作为其开发的基础平台。通过 Eclipse 平台所提供的基于 OSGI 体系构架的插件化体系结构，为系统的可扩展性打下了良好的基础；通过 SWT/JFACE 技术，可以为客户提供和 Window 本地应用感受一致的用户界面、也避免了 Java 的 Swing 技术的在客户端系统开发的不足问题。

在服务器段，采用目前最为流行的 Spring+Hibernate 轻量级应用服务器技术，通过 Spring 的本身开放式体系构架，和基于 IoC 和 AOP 的技术的优势，为服务器端提供了良好的可扩展性。另外，通过 Hibernate 这一业内领先的

ORM 技术，很好的实现了数据库应用和面向对象编程的匹配，为实现领域模型驱动式开发提供了良好基础。

在数据库方面，通过对业务数据进行基于组织机构的横向切分和基于功能模块的纵向切分，将数据库的读写压力分解到矩阵化的数据库服务器集群上，为业务发展提供了良好的可扩展空间。

在分布式应用上，采用基于 Remoting + HTTP + Binary 的技术方案实现了分布式体系结构，通过在传输过程中的压缩和加密技术，进一步提高了系统性能和安全性。

综上所述，可以看到，L 物流软件是建立在一套科学的、具有良好技术框架和可扩展性的技术平台之上的，其基础技术方案和技术框架，可以在以后的产品化研发过程中充分复用，为产品化研发打下了坚实的技术基础。

#### 4.4.2. 行业领域知识

由于 L 物流软件属于 A 企业核心业务运营支撑系统，而 A 企业又是一家全国范围具有上万人和一千多个分公司和办事处的企业，其软件所涉及到的业务范围包括前台营业、运输管理、运作管理、财务控制与管理、质量监控与管理、运营决策与分析等多个业务环节，覆盖了企业绝大部分的信息化需求，其业务复杂性和综合性可想而知。

##### 4.4.2.1. 领域知识的管理

在 L 物流软件的开发过程中，通过引入专业的业务人才、技术人才和先进的建模设计方法，将 A 企业的业务知识进行了系统化的分析、整理和建模，形成了宝贵的行业知识财富。

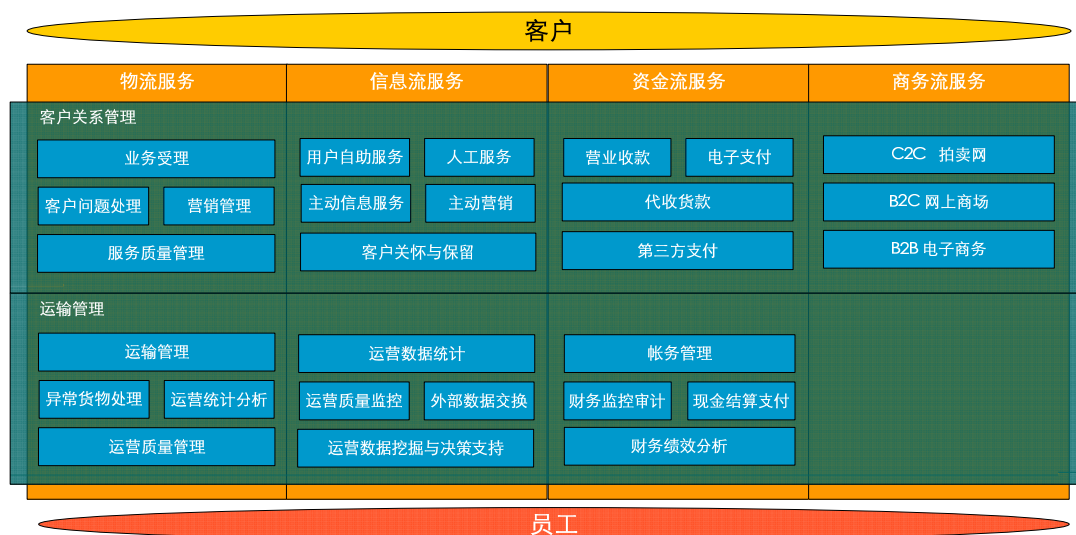
在 L 物流软件开发的过程中，采用了基于 UML 2.0 标准的建模规范，对业务领域进行了业务用例的建模和领域模型的设计，并整理了标准的适用于 A 公司业务情况的物流行业术语表。

在对 A 公司的业务用例进行系统分析的过程中，采用用例分析技术，对业务功能和场景，使用标准化的用例文本进行了规范化描述，从而便于开发人员进行相关的设计和开发工作。

为了方便领域知识的学习和传播，在内部的基于 Wiki 的知识管理平台上，专门建立了物流领域知识专题，由专人进行维护，供开发人员学习和了解。同时，也组织一线业务人员给研发人员进行业务培训，使其增强业务知识。另外，通过将研发人员派驻到 A 公司的各个部门一段时间，也使得开发人员进一步了解了一线的业务情况，增加了其对实际业务理解。

#### 4.4.2.2. 物流运营模型

在对系统功能设计的过程中，在对 A 公司业务范围与内容进行分析的同时，进一步进行归纳、总结和抽象，总结出完整的业务领域模型。在这个过程中，参考了国外相关行业的标准和规范，最终形成了具有一定行业普遍意义的物流运营模型（Logistics Operation Model，以下简称 LOM）。



图表 5 物流运营模型（LOM）1 级模型

LOM 整体划分为四纵两横的整体结构。四纵由物流服务、信息服务、资金服务、商务服务四种服务组成，构成了为客户提供的立体化服务内容。两横由客户关系管理和运输管理组成，组成了企业运营的两个基础层面。四纵体现了以客户为导向的经营理念，两横体现了以客户为导向的客户关系管理和以内部运营为导向的运输管理为企业运营管理的活动。

通过 LOM，可以清楚的将企业内部的各个业务和运营活动进行分类和描述，便于企业在具体管理过程中进行参照，同时也是信息系统建设的指南和地图。

LOM 根据细化的层次不同，分为 0 级模型、1 级模型、2 级模型和 3 级模



型。不同级别的模型对业务模型进行了不同层次上的细化。

整个 LOM 模型参考了 TMForum 提出的 eTOM 6.0、美国供应链协会提出的 SCOR 8.0 模型，并结合 A 公司目前的实际业务内容，对 A 公司未来的业务发展和信息系统建设提供了纲领性和前瞻性的规划与指南。

### 4.4.3. 软件研发管理实践

#### 4.4.3.1. 软件开发过程

在整个项目管理的过程中，采用以 RUP 为基础过程框架，并充分吸收和采纳敏捷软件开发过程中的最佳实践，并在实际项目开发过程中，不断实践和创新，发展出有自身特色的研发管理办法和技术。下面选取部分内容予以简单介绍。

#### 4.4.3.2. 每日晨会制度

在项目的实际开展过程中，吸收了 Scrum 的每天早上的晨会制度的精髓，通过每天早上 15~30 分钟的沟通，快速汇报昨天的工作进展、昨天的碰到的问题和困难和今天的工作目标三项主要部分。通过每日晨会，也提高了开发人员的沟通速度，而且这也是一个对大家反复教育和培训的最好时间。通过对某个开发人员工作存在的问题或取得的经验进行快速分析和讨论，达到了对员工的最为便捷和迅速的培训和教育。

在每日晨会的过程中，研发人员对工作的汇报也需要进行量化，并有专人负责记录和统计，这样通过最为经济的手段，达到了对工作的量化度量 and 统计，也将统计工作对研发人员的日常工作的干扰降低最低。

#### 4.4.3.3. 产品进度公告板

为了将项目的最新进展情况实施反馈给研发团队，在研发团队的办公区域，专门悬挂了一块大屏幕液晶电视，随时将系统的最新进度，度量统计数据，持续集成状态，任务统计数据以图形化的方式，简单明了的展示给整个团队。通过自行开发相关系统，实现了统计数据展示的实时化和自动化。

#### 4.4.3.4. 测试人员驱动开发

在开发团队的组织上,将研发人员和测试人员按照 2 比 1 的比例进行组队,并在办公区域上将其就近安排,使其能充分进行协调和沟通。其中,对测试人员的要求,不仅仅只是通过功能需求的理解,撰写测试用例和软件功能进行,而是要求在功能分析、设计、编码和测试阶段都要全程参与其中,测试人员也要求要进行编码,即开发人员编写功能代码,测试人员编写测试代码。

测试人员要负责与业务人员一起进行测试场景的分析和模拟,编写测试数据,定义测试接口;开发人员与测试人员一起定义测试接口,编写功能用例,并根据测试接口,编写基于用例测试接口的单元测试对功能代码进行测试。

通过将测试人员引入到研发团队研发的全过程,有效的提高了测试人员和开发人员的配合默契程度;通过职责明确的工作分工,也对编码的质量提高有了很好的帮助。从实际实践效果来看,效果非常良好,代码的返工率成倍下降,一些配合默契度高的小组,整个编码过程通过这种方式,基本上可以一次通过测试,而且因为沟通问题和理解问题的错误基本上下降为零。

#### 4.4.3.5. 文档与代码

在 L 物流软件研发的过程中,并没有象传统软件开发那样,要求开发和设计人员在所谓的文档上花费大量功夫。传统软件开发所产生的文档到项目后期维护成本很高,由于更新不能及时,最终成为废文档和死文档。

在研发过程中, B 公司强调编写活的文档,不写无用的文档。通过使用 Javadoc 技术将文档和代码进行有效结合,从技术上根本解决了文档和代码的脱节,并且通过文档生成技术,最终形成详细的软件分析与设计文档。

在文档的编写上, B 公司有着严格的要求,文档力求简洁,但要清晰易懂,格式规范。文档的生成和检查有专人进行负责,定期会对文档的撰写进行抽查,并对抽查的结果进行公布,持续性的提高文档质量。

在这一过程中,包括领域模型的定义,数据库结构的定义、功能用例的表述、业务接口的定义、业务流程的描述、程序接口和参数的定义等等,从最高层次的需求到最低层次的程序方法和算法,在文档都进行了编写,并随时保持更新。

可以看到，通过上述实践，有效解决了文档编写问题，让开发人员养成了编写文档的良好习惯，也从技术层面解决了管理上的可操作与可执行问题。

#### 4.4.4. 软件生产的工业化

在 L 物流软件项目开发的过程中，也做了很多为了向工业化生产迈进的有益尝试，并取得了良好的效果。

##### 4.4.4.1. 模型驱动开发

在 L 物流软件中，通过以领域模型为核心，实现了领域驱动开发。通过对领域模型进行丰富，将领域知识、业务需求、数据库定义、业务规则、业务约束在领域层从分析层就进行了定义，并通过标注技术，直接体现在代码层上。通过这一技术实现，将模型驱动开发，在实用的角度上，进行了实际的具体实践，并取得了良好效果。

##### 4.4.4.2. 自动化生成技术

在 L 物流软件开发过程中，对模型层的代码，使用了基于元数据模型的自动生成技术，通过在领域模型中的分析和定义，自动化产生相关的类、接口、方法和注释、自动生成相关的配置文件，大大减少的重复性的手工劳动，降低了因人工操作带来的错误，有效的提高了生产效率。

##### 4.4.4.3. 可复用组件技术

可复用组件技术在 L 物流软件开发过程中也做了实际的应用，并取得很好的效果，这一点在基于 Eclipse RCP 平台的客户端软件开发过程中显得尤为突出。在开发过程中，开发人员逐渐积累和整理了一批可复用的组件和工具类代码，在其后的开发过程中，大大提高了开发速度和开发的代码质量。

## 4.4.5. 软件产品质量

### 4.4.5.1. 客户全程参与

引入现场的业务团队，对项目软件的开发具有十分重要的帮助。这些人员都是从一线临时抽调或直接调入到研发团队中来的，既有一线的操作人员、财务人员，也有具有一定管理经验的一线管理人员、电脑管理员等等。通过这样一支长期和研发团队工作在一起的业务团队，保证了业务系统的开发能最大程度符合和满足各级业务操作和管理部门的需求，能快速将业务一线的最新要求反馈给研发团队，减少了不必要的沟通成本，降低了返工成本。通过这一机制，研发的速度和质量相对于原有的开发方法，开发速度提高了近30%，缺陷率降低了约50%左右。因对业务需求的理解偏差造成的缺陷平均每个用例只有0~5个，而且在研发的早期活动中就被及时的调整和修正了。

### 4.4.5.2. 测试数据工厂

在3.3.6.3业务场景模拟实验室一章中，提到了对业务场景模拟的困难，在L项目开发过程中也实际碰到了这样的问题，特别是异常业务流程的模拟和数据准备工作相当繁重。

为此，通过自身的摸索和实践，逐步掌握和实现了一套基于流程和场景的自动化业务场景数据生成工厂技术。通过编程模拟真实业务环节，产生场景数据，业务测试人员只需要指定业务场景的编号，就可以自动化的初始化业务场景数据，就可以供业务人员进行各种业务场景的测试工作。

## 4.5. 软件人才

### 4.5.1. 技术团队的建立

在L物流软件开发之初，也碰到人才极度缺乏的问题。为了解决这一问题，B公司通过多种渠道和方式，不断的招聘和吸引优秀人才进入到公司。通过内部推荐、人才招聘网站、猎头服务、在一些技术论坛上直接联系和发

布招聘信息，通过开发人员的博客联系、通过培训学校招聘等等多种手段，才逐步建立和完善起来一支具有各方面专业技术和知识的技术团队。

#### 4.5.2. 技术团队的培养

技术人员进入团队后，都需要在专业技术方面和业务知识方面进行进一步的培训，而且随着项目本身的不断深入，需要有更多专业化的技术和问题需要进行攻克。

在技术人员的培训和培养上，B公司秉持“在游泳中学游泳”的思想，通过给技术人员分配适合其自身能力的问题，让其去主动学习、研究和解决，从而提高其学习能力和研究能力，在这个过程中不但为企业解决了具体问题，也锻炼了员工，增加了其知识，增强了其能力。

#### 4.5.3. 技术团队的使用

在整个研发过程中，将研发人员进行编组，两个开发人员和一个测试人员一组。在编组的时候需要考虑其组员的性格特征，使小组的配合能够和谐一致。开发人员在一段时间内，专攻一方面的技术，或服务器端，或客户端，形成“术业有专攻”；测试人员也要求具有一定的研发能力，才能和开发人员密切配合。从实际效果上看，虽然各个小组能力和配合程度都有一定差异，但总体水平都比各自为战的情况有很大提高，其团队作战能力普遍增强。

#### 4.5.4. 技术团队的保留

由于人员流动是不可避免的事情，因此必须正确面对和解决这个问题。在L物流软件研发过程中，特别是在后期，也碰到这方面的问题。在这方面主要进行了几个方面的工作。

其一，加强核心团队的稳定。对开发团队中，表现优秀，具有责任感和技术能力的人员委以重任，并保证其薪资待遇在市场上具有竞争力。

其二，长期进行人员招聘工作。要将人才的招聘工作作为一项日常工作，要挑选素质好，能力强的可造之才。

其三，对研发人员的情绪和问题进行及时沟通。通过密切了解开发人员的动向，采用走动式管理，经常和开发人员进行正式和非正式的沟通，及早掌握他们的想法，帮助解决其具体问题，将人才流动的问题化解在早期。

其四，准备好预备团队。在人员配置上，留有一定的空间，避免因人员变动造成产能不足。预备团队往往是从培训公司招聘的一些比较好的苗子，进行一定的培养，在合适的阶段就可以承担起具体的任务。

## 4.6. 小结

本案例对 B 公司在开发 L 物流软件项目过程中，所涉及产品化开发的工作进行了总结，对前文提出的软件产品化解决之道进行了具体的阐述和说明。

本案例中所涉及一些产品化实践、包括企业运营、企业管理、软件技术、软件人才等方面的具体实践，希望能对读者能有一定的借鉴和帮助作用。对于这些实践方法，读者不要盲目跟从，要搞清其应用的前提条件和环境，逐步实践和体会，并根据自身特点进行调整，才能起到更好的效果。