

第三章

发展中的 信息系统

目标

- 描述能力成熟度模型（CMM）。
- 区分系统的使用寿命一个周期和一个系统的开发一套方法
- 描述中的部件框架分类问题，机会，和指令。

目标

- 描述该项目的基本阶段
系统发展对于每个阶段，描述它的目的、输入和产量
- 通过系统开发的基本阶段描述典型的替代“路线”。

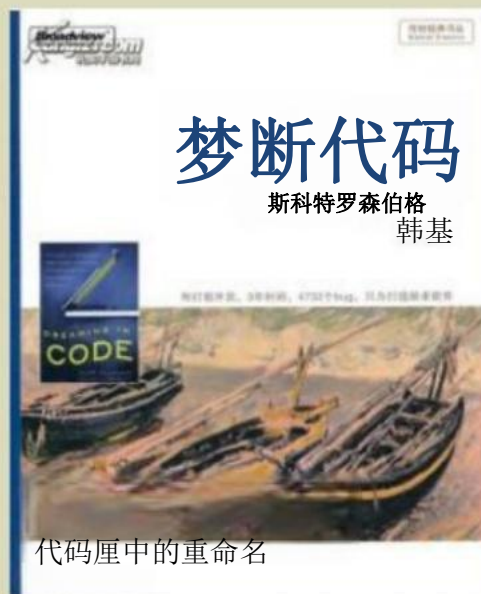
3.1 系统开发过程

系统开发过程—一系列活动，
利益相关者（第1章）用于开发和持续改进信息系
统的方法、最佳实践、可交付成果和自动化工具
软件（第1章和第2章）。

使用一致的系统开发流程：创建效率，允许管理人员在
不同项目之间转移资源
生成一致的文档，从而减少
维护系统的生命周期成本
提升质量

困难的软件难做

- Scott Rosenberg见证了软件世界中无数个悲惨的故事, 并得出结论: 无论是跨国公司、政府机构, 还是军工大鳄, 都曾一头撞上过代码的冰山.



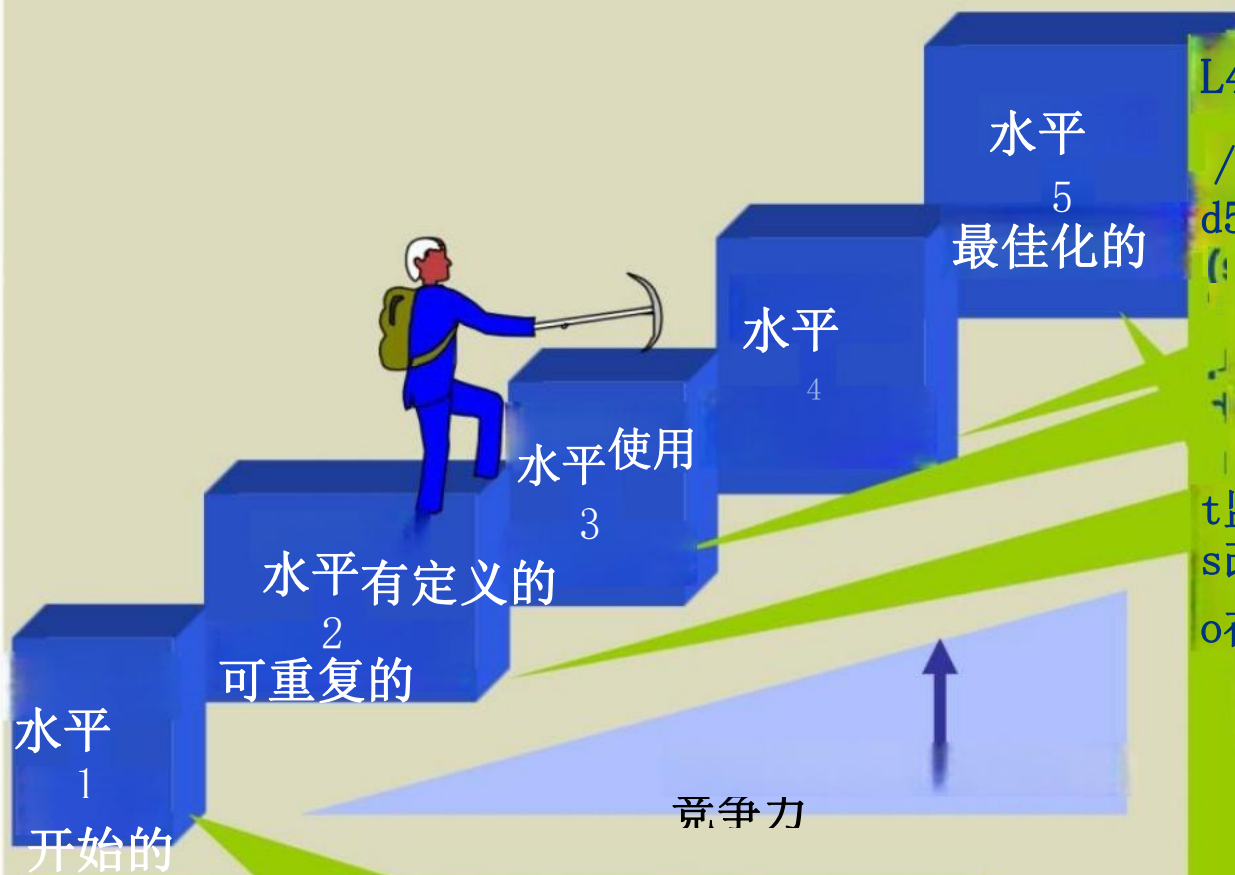
示例

- FBI投资4亿美元，花费10年时间，开展的名为Trilogy的计算机现代化项目，当进行到耗资1亿7千万美元的第三个模块时，撞上了冰山，1亿7千万美元也就此打了水漂。
- 美国国内税务局（IRS）是另一个活生生的例子。IRS在过去40年里，曾3次尝试改造计算机系统。
- 联邦航空管理局（FAA）的先进自动系统（AAS）项目，从1981年启动到1994年宣告终止，花费了几十亿美元。
- 麦当劳的“Innocvate”项目耗资1亿7千万美金，也以失败告终。
- 福特公司耗资数亿美金的“Everest”采购系统，也成了一个历时5年的黑洞。

3.1.1C MM过程管理模型

能力成熟度模型（CMM）-一个标准化的模型
评估a成熟度水平的框架
组织的信息系统开发和开发
管理流程和产品。它由五个
成熟度：

- 1级-初始：系统开发项目规定的过程。
- **第2级-可重复的：项目管理流程和**
为跟踪项目成本、进度计划和功能
- 第3级一定义：购买或开发标准系统开发过程（方法）。所有的项目都使用这个过程的一个版本。
- **4级管理：可衡量的质量目标**
生产率建立。
- 第5级——最优化：标准化的系统开发过程不断被监控和改进
在第4级中建立的措施和数据分析。



L4级管理:

/Maacurahle nale fnr
d5级优化

标准化的
系统开发过程是
连续不断地
t监控和
s改进基于
o在第4级中建立的措施
和数据分析。

第1级-初始阶段: 系统开发

3-8个项目没有遵循规定的程序。

关键工艺区域

- 关键过程域：指一系列相互关联的操作活动，这些活动反映了一个软件组织改进过程时必须集中精力改进的几个方面。换句话说，关键过程域标识了达到某个成熟度等级时所必须满足的条件。
- 在 CMM 中每个成熟度等级(第1级除外)规定了不同的关键过程域，一个软件组织如果希望达到某一个成熟度等级，就必须完全满足关键过程域所规定的不同要求，即满足每个关键过程域的目标。
- 在 CMM 中一共有18个关键过程域，分布在2~5级中。

- 第2级(可重复级) 有6个关键过程域, 主要涉及建立 软件项目管理控制方面的内容。

- 需求管理 (需求管理, RM)

- 软件项目计划 (软件项目规划, SPP)
监督, SPTO)

- 软件子合同管理(软件分包合同管理公司
SSM)

- 软件质量保证 (软件质量保证, SQA) • 软件配置管理(软件组态管理,
SCM)

第3级(定义级)有7个关键过程域，主要涉及项目和组织的策略。使软件组织建立起对项目中的有效计划和管 理过程的内部细节。

- 组织过程焦点（组织流程重点，OPF）
- 组织过程定义(组织流程定义，OPD)
- 培训程序（培训计划，TP）
- 集成软件管理（集成软件管理，ISM）
- 软件产品工程（软件产品工程，SPE）
- 组间协调（组间协调，IC）
- 同级评审（同行评审，PR）

第4级(管理级) 有 2 个关键过程域，主要的任务是为软件过程和软件产品建立一种可以理解的定量的方式。

- 定量过程管理（定量过程管理，QPM）
- 软件质量管理（软件质量管理，SQM）

第 5 级（优化级）有3个关键过程域，主要涉及的内容是软件组织和项目中如何实现持续不断的过程改进问题。

- 缺陷预防（缺陷预防，DP）
- 技术变更管理(技术变革管理公司，
终端设备至计算机多路转接器
- 过程变更管理(流程变更管理公司，PCM)

系统开发的影响 质量的“过程”

CMM项目统计数据的一个项目导致200,000行代码

组织机构 在CMM上 水平	普罗杰 克 t 杜拉蒂 在… 上 (mo nt hs)	项 目 人 员 月 份	数 量 的 缺 陷 已 发 货	中 位 数 成 本 (\$ 百 万)	最 低 的 成 本 (\$ 百 万)	最 高 的 代 价 (\$ 数 百 万)
1	30	600	61	5.5	1.8	100+

2	18. 5	143	12	1. 3	. 96	1. 7
3	15	80	7	. 728	. 518	. 933

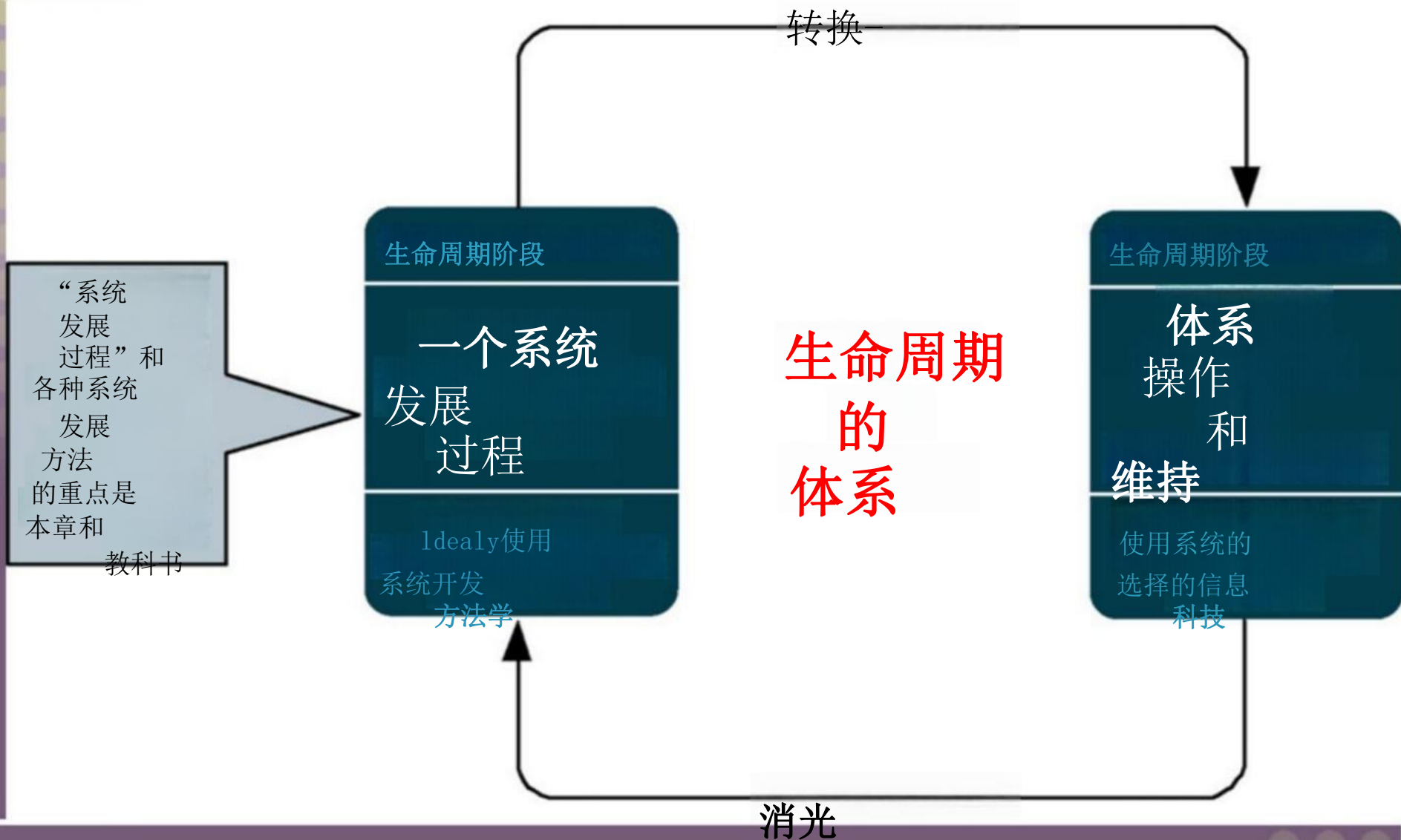
ISO9000 和 CMM

对比项	ISO9000	CMM
适应对象	全部产品(含软件)	只有计算机软件
服务对象	企业市场各方	承制方
性质	强调可接收的企业质量管理体系的最低标准	软件过程能力成熟度标准
用途	认证合格与否	评价能力等级
目的	向顾客提供信任	降低风险或进行过程改进

3. 1. 2生命周期

- 系统生命周期-将信息系统的生命周期分解为两个阶段：(1)系统开发和(2)系统的运行和维护。

系统生命周期



3.2. 1T他碎片问题 解决框架

P需要提高性能

1需要改进信息(和
数据

E改善经济效益的需要，提高控制能力
成本，或增加利润

C需要改进控制或安全性

E需要提高人们的工作效率
和过程

我们需要改善对客户、供应商、合作伙伴、员工等的服务。

各部分的框架和检查清单

- 性能：吞吐量响应时间 • 信息（和数据）

输出输入存储数据

- 经济方面：成本利润 •

- 控制（和安全）

安全措施或控制能力太少

控制措施或安全措施过多

- 效率

数据被冗余地输入、复制、处理。 • 服务

该系统不容易使用和学习

该系统与其他系统不兼容

- 这些作品的类别
框架既不详尽，也不相互排斥——它们是重叠的。任何给定的项目通常都有一个或多个类别
给定的问题或机会可能会出现
对不止一个类别都有影响。碎片是一个
实践的框架，而不仅仅是一个
学术练习。

样例

唯品会
品牌特卖

1. 改进性能的需要 P

- 网站的展现速度有点慢，挑战访客的耐心，仍需加强改进。在等待的这段时间里访客估计又可以搜索到另一个同类网站了。
- 高峰期很慢，有点瘫痪。从购买到收到需要1周。

2. 改进信息(和数据)的需要 I

- 唯品会系统追踪不到订单
- 唯品会系统品牌供应商申请上线不能在线申请，只能

EMAIL: BRAND@VIPSHOP.COM

传真: 020-22330164(只接受传真)

地址: 广州市荔湾区花海街20号唯品会(醉观公园旁) 邮编: 510370

3. 页面设计、排版 I

4. 提升对产品描述的吸引力 I

5. 改进经济、控制成本或增加收益的需要 E

- 库存费用，库存成本仍然较大
- 物流系统费用高
- 宣传与广告费用大

6. 改进控制或安全的需要 C

- 发货发错
- 抢购唯品会商品遇系统问题，数据库录入错误，拍下产品后却没货。
- 安全控制系统不够，如可以通过条形码在网上进行在线检验，避免顾客对产品的真伪怀疑。

7. 改进人与过程的效率的需要 E

- 在唯品会网页端把商品加入购物车不方便

样例

唯品会
品牌特卖

8. 改进对客户、供应商、合作伙伴、雇员等的服务需要 S

- 唯品会退款不及时，客服称是系统问题
- 下单一月货未收到唯品会自动确认收货
- 当重视用户购物体验，出现问题应及时处理，注重效率

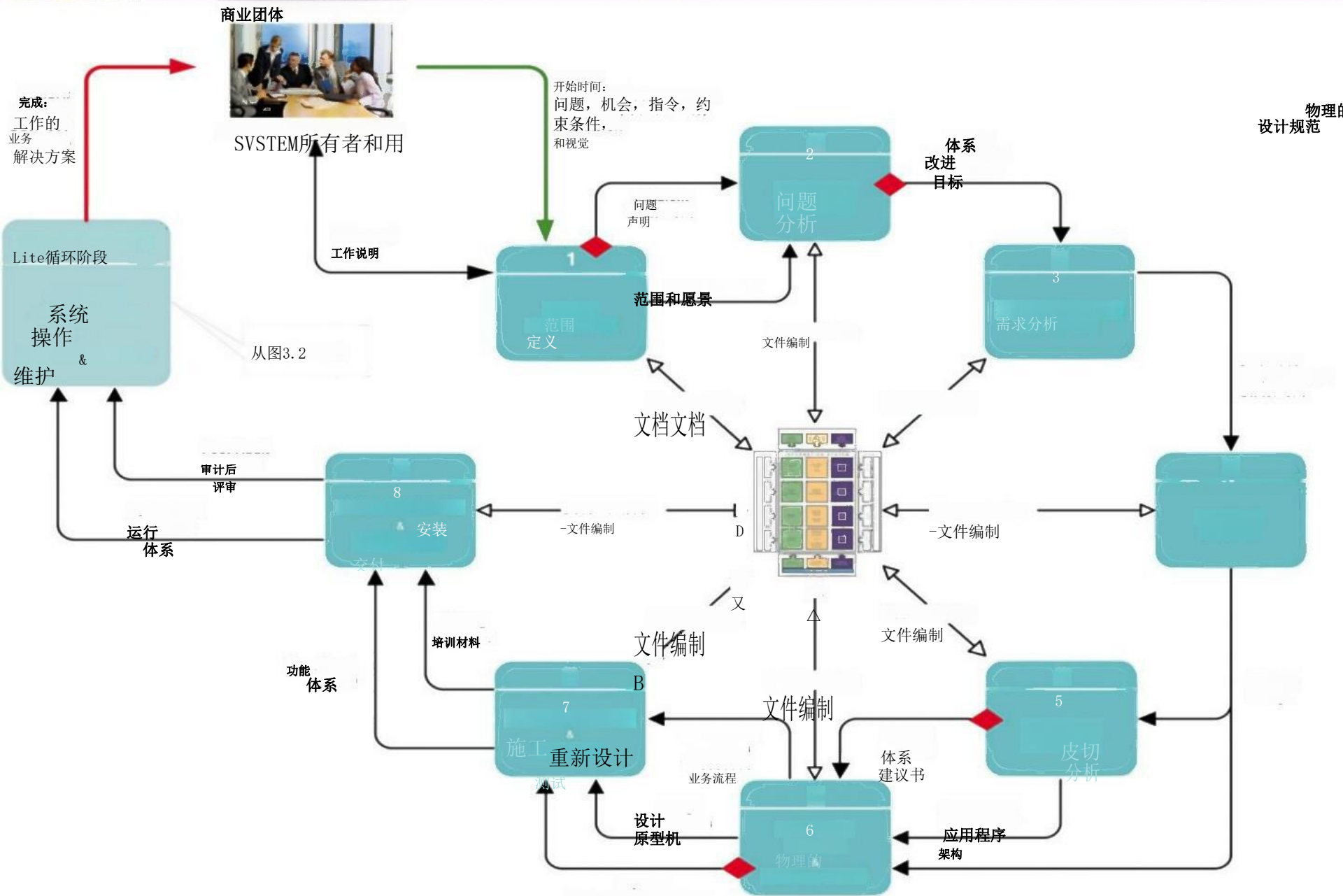
3.2.2 项目阶段

- FAST-（系统思维应用框架）使用的一种假设方法
在整本书中展示了一个具有代表性的系统开发过程。

每种方法都将使用不同的项目阶段。

快速阶段	经典阶段（源自第1章）			
	项目 启动	系统分 析	体系 设计	体系 实施
范围定义	X			
问题分析	X	X		
需求分析		X		
逻辑设计		X		
决策分析			（系统分析过渡阶段）	
物理设计与集成			X	
施工和测试			X	X
安装和交付				X

FAST项目阶段



业务
要求声明

逻辑的
设计

逻辑的
设计

范围定义阶段

问题语句和分类

问题、机会和指示；也可能包括约束条件和初步设想。同义词包括初步研究和可行性评估的部分。--

约束-任何可能限制a的因素、限制或限制解决方案或解决问题的过程。

范围蠕变（范围蔓延）-一种常见的现象
对一个项目的需求和期望通常会增加而不考虑对预算和时间表的影响。

工作声明-与管理层签订的合同

通过用户社区来开发或增强一个信息系统；
定义视觉（目标）、范围、约束、高级用户需求、时间表和预算。同义词包括项目章程、项目计划和服务水平协议。

问题分析

- 里程碑：系统改进目标
- 定义任何新系统所采用的业务标准

将评估。

——减少订单处理和运输之间的时间

三天。

——减少45%的不良信用损失

检查点

- 如果问题被认为不再值得解决，则取消-批准继续进行下一步工作
- 减少或扩大范围（包括预算和计划修改），然后被批准继续进行下一步操作

需求分析阶段

- 新系统应该具备什么功能为其用户提供？
- 必须捕获和存储哪些数据？
- 预期的绩效水平？
- 不同项目的优先事项是什么所需的東西为什么P57

逻辑设计阶段

逻辑设计-将业务用户需求转换为一个系统模型，只描述业务需求，而不描述任何可能的技术设计或实现这些要求。常见的同义词包括概念性的设计和基本设计。

系统模型-代表现实或期望现实的系统的画面。系统模型便于改进系统用户、系统分析师、系统设计人员和系统构建者之间的通信。

分析瘫痪-一个讽刺（讽刺）术语创造了描述一种常见的项目条件，在此条件下，过度的系统建模显著降低了预期系统解决方案的实现进度。

决策分析阶段

- 根据以下方面评估候选解决方案：
 - 技术可行性-该解决方案在技术上是否可行？我们的员工是否有设计和构建这个解决方案的技术专长？
 - 操作可行性-该解决方案是否能满足用户的需求
所需的東西到什么程度？这个解决方案将会如何改变呢
用户的工作环境？用户是如何看待这样的一个
解决办法
 - 经济可行性-该解决方案是否具有成本效益？
 - 计划的可行性-该解决方案能否在可接受的时间内进行设计和实施？
 - 风险可行性-使用该技术和方法成功实施的可能性是多少？

物理设计与集成阶段

物理设计-业务用户的翻译
需求到描述技术的系统模型
实现用户的业务需求。常见的
同义词包括技术设计或实现模型。

物理设计的两种极端哲学（极端情况）

- *按规范设计-物理系统模型和详细设计*

规范是一系列书面的（或计算机生成的）施工蓝图。

通过原型设计进行设计——不完整但功能良好的应用程序或子系统
• （称为原型）是基于用户和其他设计者的反馈来构建和改进的。

逻辑vs。物理模型

逻辑模型-描述系统的状态或行为的图形表示。

物理模型-一个技术图片
描述系统是什么或做什么以及系统是如何实现的表示。

施工和测试阶段

- 构建和测试系统组件
 - Software
 - Purchased
 - 定制的
 - 数据库
 - 用户和系统接口
 - Hardware
 - 网络

安装和交付阶段

- 使系统投入运行
生产
- 提供用户培训
- 交付完整的文件
- 转换现有数据

系统操作和 维持

系统支持-正在进行的技术支持

对系统用户的支持，以及处理任何问题所需的维护

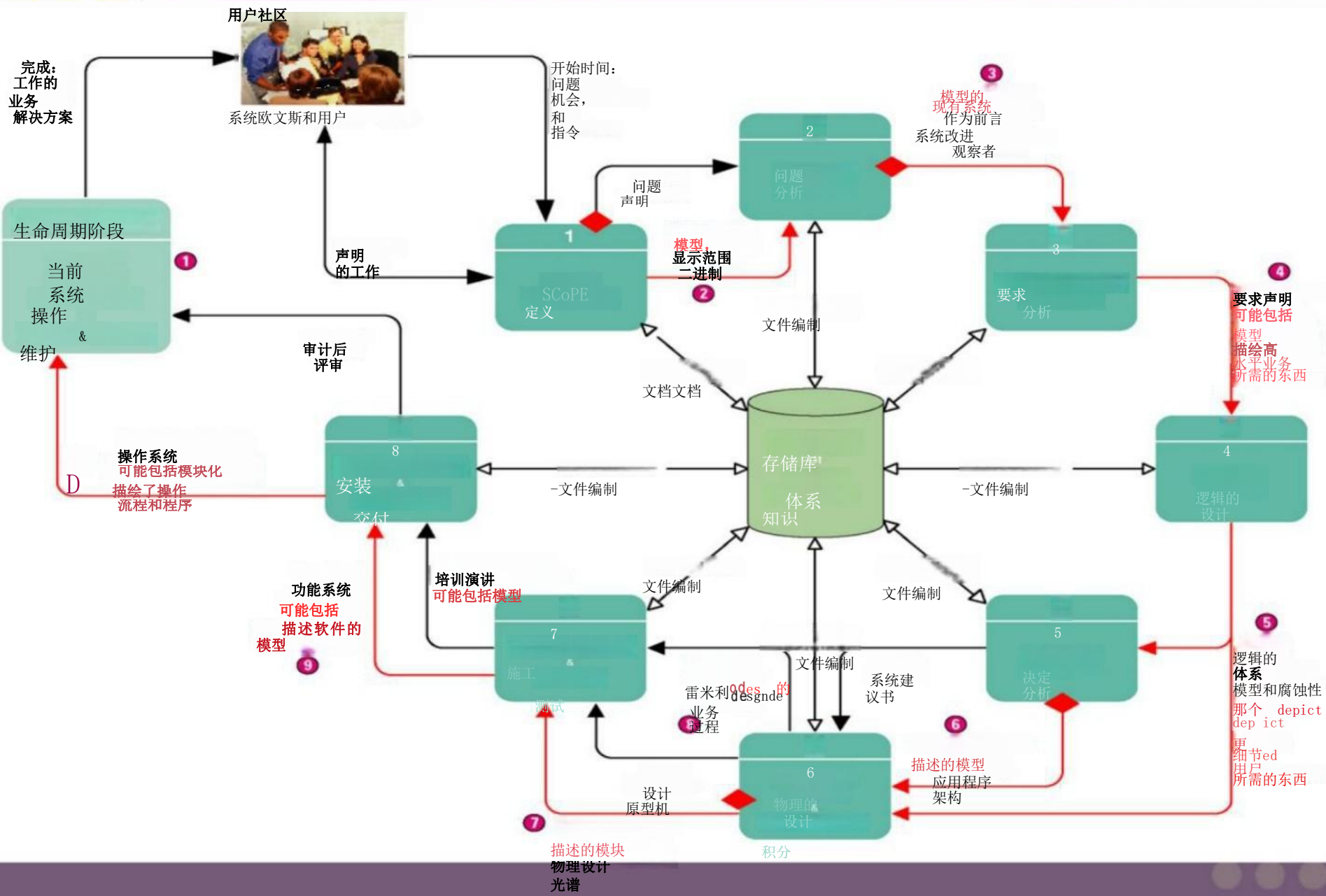
错误、遗漏、或新的要求

这可能会出现。

3.3.1模型驱动的开发策略

- 模型驱动的开发-一个系统的开发
强调绘制系统模型的策略
要帮助可视化和分析问题，请定义业务需求和设计信息系统。
 - 流程建模—一种以流程为中心的技术
通过结构化分析和设计方法所使用的
建立了业务流程的需求模型，以获得有效的结果
针对一个系统的软件设计。
 - 数据建模—一种用于建模的以数据为中心的技术
业务数据需求和设计的数据库系统
满足这些要求。
 - 对象建模—一种试图合并数据的技术
并将相关问题处理成被称为对象的奇异结构。
对象模型是一种用来记录系统的图表
它的对象和它们之间的互动。

模型驱动的开发策略



优点

- 要求往往更为彻底
- 更容易分析选择的余地
- 设计规范通常更稳定和更稳定柔韧的
- 系统可以构建更多第一次正确

缺点

- 耗时
- 模型只有一样好作为用户的理解所需的東西
- 减少了用户的角色，因为图片不是软件
- 可能不灵活

- 模型驱动的开发是最重要的有效的系统
需求已经被很好地理解和
它们是如此复杂，他们需要
大型项目团队要完成。这个
当满足用户的期望和质量更高时，该方法
也很有效
比成本和时间表更重要。

3.3.2快速应用程序 发展战略

- 快速应用程序开发（RAD）-一个系统

强调发展速度的发展战略

通过广泛的用户参与快速式、迭代式和增量式（持续不断增加的）来进行开发

构建一系列系统的功能原型，最终发展为最终的系统

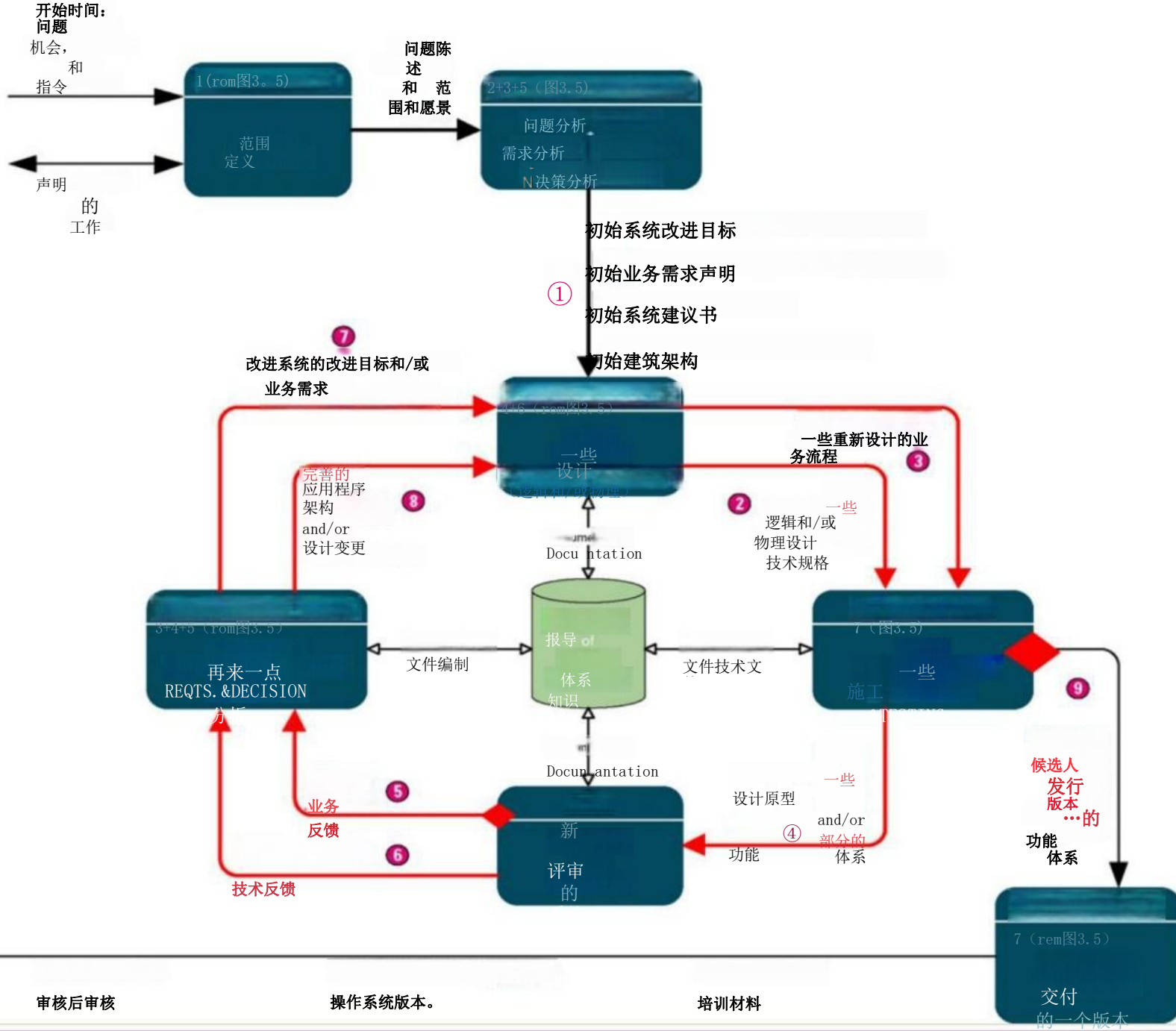
。

- 原型-用户需求的小规模、代表性或工作模型或建议的设计
信息系统

- 时间盒-不可扩展功能的实施（强迫接受）
一段时间，通常是60-90天，系统的第一个（或下一个）版本必须交付投入运行。



ss系统的所有者和用户



完成:
工作的
业务
解决方案

生命周期阶段

当前
系统
操作

维护

审核后审核

操作系统版本。

培训材料

交付的一个版本

- RAD最受中小企业的欢迎规模项目。

3.3.3 商业应用程序包实施策略（自学）

- 商业应用程序软件包-软件应用程序
这可以购买和定制，以满足业务
大量组织的要求或
特定行业。同义词是商业性的现成的
（COTS）系统。
 - 提案请求书（RFP）（建议申报书）-正式文件
它可以通信业务、技术和支持需求
为应用软件包的供应商可能希望
竞争应用程序包和服务的销售。
 - 报价请求书（RFQ）（报价申报书）-正式文件
它可以通信业务、技术和支持需求
为一个应用软件包到单个供应商，有
已被确定为能够提供该应用程序包
和服务。
 - 差距分析-业务和技术上的比较
对商业应用程序包的要求

一个特定的商业应用程序的功能和特性使用软件包来定义无法满足的需求。

用户社区



SVSTEM所有者和用户

科技产业



销售代表和
技术测试器

完成：
工作的
业务
解决方案

生命周期阶段

当前系统
操作

维护

操作系统

审计后审查

(图3.5)

交付

训练
材料

功能系统

40

7 b (rom图3.5)

施工

测试

开始时间：
问题
机会，
和
指令

工作说明

问题
陈述

范围
定义

范围和愿景

文件编制

文档文档

文件编制

文档文档

设计原型

重新设计的业务流程

定制的商业应用程序

技术市场研究

2

体系
改进
目标

1

需求分析

快速充盈期
或
请求报价

4

建议书
或
报价单

5

合同

0

业务需求
声明

决定
分析

特点和
能力

7

业务
过程
设计

6

自定义
要求

基线商业应用

对这条路线新
安装
自定义

添加/输入
软件要求

8

物理的
设计

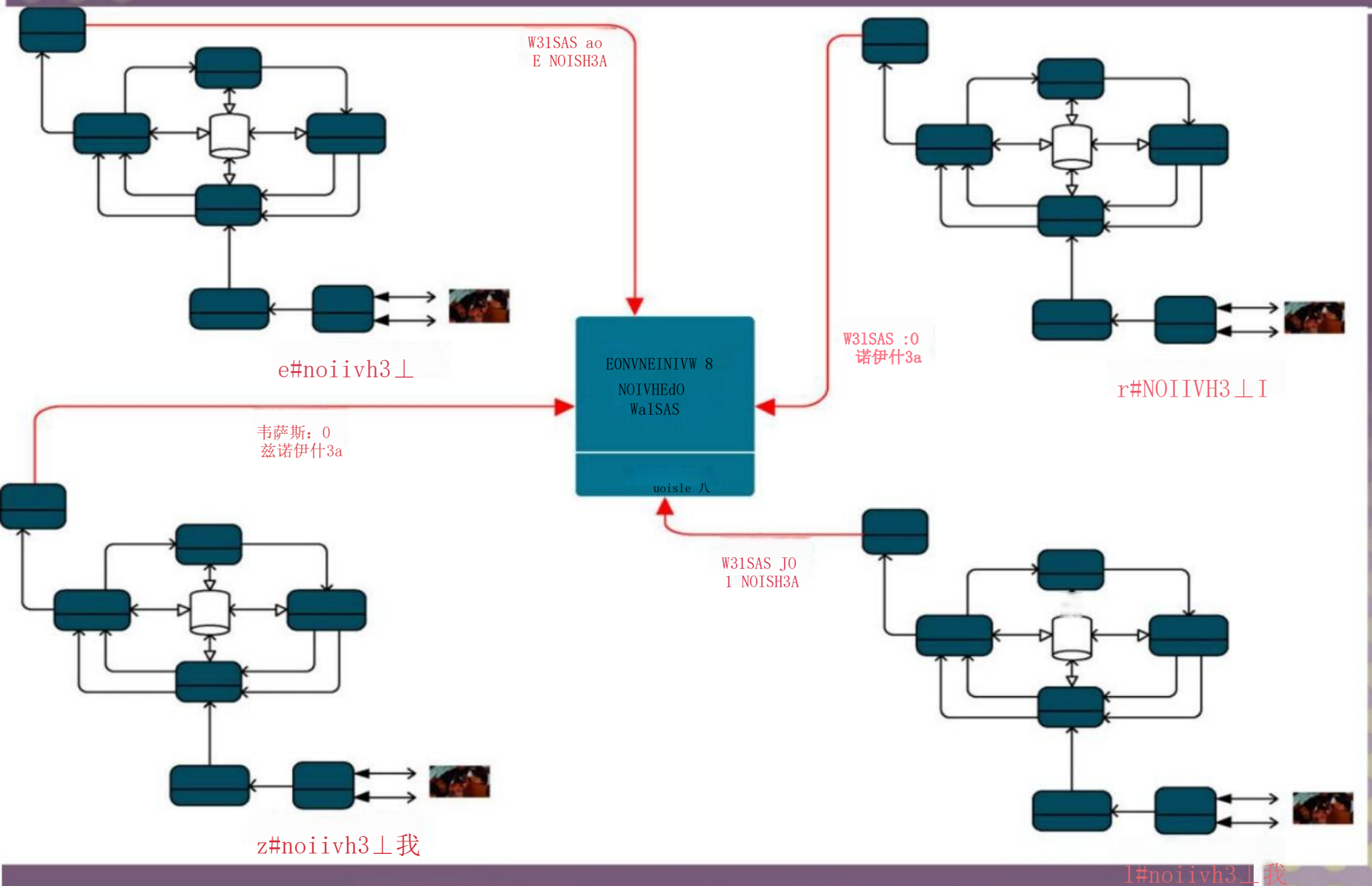
积分

9

商业应用程序包 实施策略

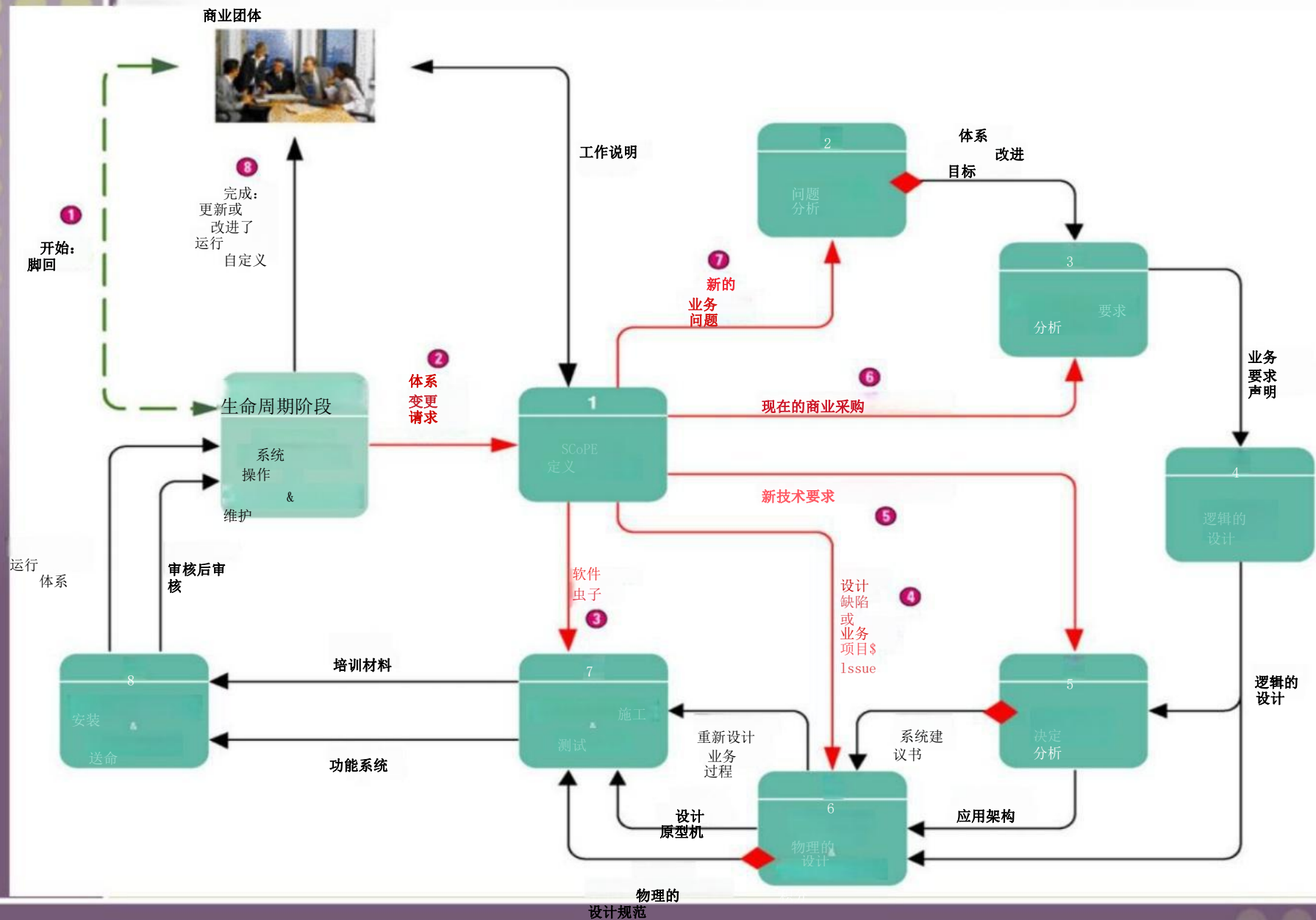
优点缺点

- 系统通常实现得更快
- 避免了开发内部解决方案所需的人员配备
- 一般来说便宜
- 供应商承担改进和改进的责任
- 修改
- 许多业务功能更多
- 相似比不同
- 在一个特定的行业中经营的企业
- 依赖于长期
 供应商的生存能力
- 很少能反映出理想的解决方案
- 经常抵制改变业务流程以适应
 软件



seibaiens puqAH tgg

3.3.5 A系统维护视角



思考题:

1. 在选择开发方法时, 不适合使用原型法的情况是 ()。

- A. 用户需求不清楚 B. 系统设计方案难以确定
C. 系统使用范围变化很大
D. 用户的数据资源缺乏组织和管理

以下情况为不适合使用原型法:

- 用户的数据资源缺乏组织和管理
- 用户的软件资源缺乏组织和管理
- 用户不参与, 不积极配合开发过程

思考题：

2. 在开发一个系统时，如果用户对系统目标不很清楚，难以定义需求，这时最好使用什么开发路线？
- A. 快速应用开发
 - B. 模型驱动

3.4 自动化工具技术

- 计算机辅助系统工程
（案例）
- 应用程序开发环境
有害药物事件
- 流程和项目经理

3.4.1 计算机辅助软件 工程（案例）

Computer-辅助系统工程（CASE） -支持绘图和

系统模型及相关规范的分析。

一些案例工具还提供了原型设计和代码
发电能力。

- 案例存储库-系统开发人员的数据库中
开发人员可以存储系统模型，详细说明
描述和规格，和其他产品
系统发展同义词：字典和
百科全书
- 正向工程案例工具能力
生成初始软件或数据库代码直接从
体系
- 逆向工程案例工具能力
从软件或
数据库代码。

3.4.2 应用程序开发环境（自学）

应用程序开发环境（ADEs）-另一个集成软件开发工具，提供所有必要的设施，以开发新的应用软件的最大速度和质量。一个常见的同义词是集成开发环境（*IDE*）

- ADE设施可能包括：
 - 编程语言或口译员
 - 接口施工工具
 - Middleware
 - 测试工具
 - 版本控制工具
 - 帮助编写工具
 - 存储库链接