**第二章**

**复习题：**

1. IEEE是怎样定义需求的？从中你可以得到什么认识？

IEEE对需求的定义：

（1）用户为了解决问题或达到某些目标所需要的条件或能力；

（2）系统或系统部件为了满足合同、标准、规范或其它正式文档所规定的要求而需要具备的条件或能力；

（3）对（1）或（2）中的一个条件或一种能力的一种文档化表述。

为了融合不同群体的看法，IEEE的定义当中同时包括了用户的观点（第一种条件和能力）和开发者的观点（第二种条件和能力），但是即便如此，不同群体的人们也很难就IEEE的定义进行一直和准确的解读，因为需求概念的内涵和外延都非常丰富。

1. 解释下列名词：问题域、解系统和共享现象，并结合它们的含义说明软件系统是如何与现实世界形成互动的？

问题域：

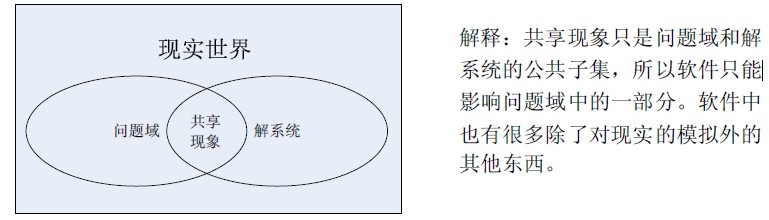
* 当现实的状况与人们期望的状况产生差距时，就产生了问题。
* 要解决问题，就需要改变现实当中某些实体的状态或改变实体状态变化的演进顺序，使其达到期望的状态或演进顺序。
* 这些实体和状态构成了问题解决的基本范围，称为该问题的问题域（Problem Domain）

解系统：

软件系统通过影响问题域，能够帮助人们解决问题，称为解系统。

共享现象：

* 软件系统能够与问题域进行交互和相互影响的原因在于，软件系统中的某些部分对问题域中的某些部分的具有模拟特性。
* 换句话说，软件系统当中含有问题域某些部分的模型（或模拟），常见的模型包括数据模型、对象模型、处理模型等。
* 问题域中的某些信息能够和模型中的信息建立映射关系
* 这些通过映射建立的共同知识，就是问题域和解系统之间的共享现象



共享现象就是问题域和解系统实现交互和互相影响的途径与接口，问题域和解系统都通过改变这些共同知识来影响对方，或者通过认同这些共同知识的改变来接受对方的影响。

1. 解释下列名词：需求、规格说明、问题与特性和约束，并结合它们的含义说明需求工程的主要任务是什么？

需求是用户对问题域当中的实体状态或事件的期望描述。

直接需求是可以通过更改共享现象被满足的需求;

间接需求是需要修改共享现象，同时连锁影响问题域才能满足的需求。

规格说明是解系统为满足用户需求而提供的解决方案，规定了解系统的行为特征。

解决方案只能通过改变共享知识，影响问题域的运行，进而满足用户的需求，所以规格说明主要包括两个部分：（1）对共享现象（模型）的描述；

（2）系统对共享现象所施加的操作的描述。

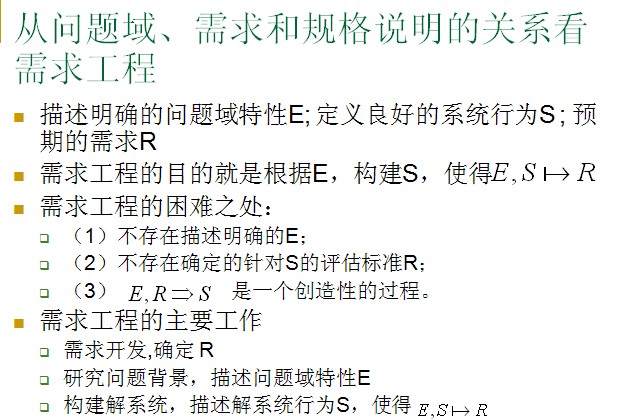
需求关注的是现实世界中的部分，软件关注的是解系统，而规格说明关注的是共享现象

问题域特性：问题域自治的规律性称为问题域特性。包括结构特性和行为特性等。

需要关注的问题域特性：

间接特性；

约束和假设：（问题域当中有些特性完全不受共享现象的影响，即完全不受解系统的影响，同时却可能很大程度上影响共享现象，影响解系统，甚至关乎解系统的成败。这些特性被认为是解系统对环境的依赖特性。当这些特性非常明确时，称之为约束；不明确时，需要限定特性的变化范围，称之为假设）



1. 需求有哪些常见的类别？功能需求和非功能需求有什么差异？

需求的分类1：

* 功能需求（Functional Requirement）：
  + 和系统主要工作相关的需求，即在不考虑物理约束的情况下，用户希望系统所能够执行的活动，这些活动可以帮助用户完成任务。功能需求主要表现为系统和环境之间的行为交互。
* 性能需求（Performance Requirement）：
  + 系统整体或系统组成部分应该拥有的性能特征，例如CPU使用率、内存使用率等。
* 质量属性（Quality Attribute）：

非功能需求

* + 系统完成工作的质量，即系统需要在一个“好的程度”上实现功能需求，例如可靠性程度、可维护性程度等。
* 对外接口（External Interface）：
  + 系统和环境中其他系统之间需要建立的接口，包括硬件接口、软件接口、数据库接口等等。
* 约束
  + 进行系统构造时需要遵守的约束，例如编程语言、硬件设施等

需求的分类2：

系统需求（System）：硬件需求（Hardware）、软件需求（Software）、其他需求

功能需求和非功能需求的差异：

除功能需求之外的其他四种类别需求又被统称为非功能需求。在非功能需求当中，质量属性对系统成败的影响极大，因此在某些情况下，非功能需求又被用来特指质量属性。

1. 描述业务需求、用户需求和系统（级）需求的区别与联系。

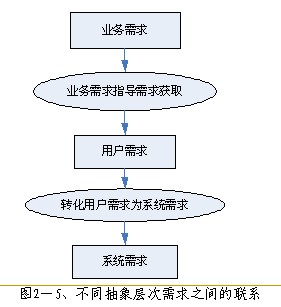
* 业务需求：

系统建立的战略出发点，表现为高层次的目标，它描述了组织为什么要开发系统

为了满足用户的业务需求，需求工程师需要描述系统高层次的解决方案，定义系统应该具备的特性（Feature）

参与各方必须要对高层次的解决方案达成一致，以建立一个共同的前景（Vision）

特性说明了系统为用户提供的各项功能，它限定了系统的范围（Scope）

* 用户需求：

执行实际工作的用户对系统所能完成的具体任务的期望

描述了系统能够助用户做些什么。

直接用户、间接用户

对所有的用户需求，都应该有充分的问题域知识作为背景支持

特性：模糊不清晰、多特性混杂、多逻辑混杂

* 系统需求：

用户对系统行为的期望，一系列的系统行为联系在一起可以帮助用户完成任务，满足业务需求；系统需求可以直接映射为系统行为，定义了系统中需要实现的功能，描述了开发人员需要实现什么。

用户需求---->系统需求的过程：

首先需要分析问题领域及其特性，从中发现问题域和计算机系统的共享知识，建立系统的知识模型；

然后将用户需求部署到系统模型当中，即定义系列的系统行为，让它们联合起来实现用户需求，每一个系统行为即为一个系统需求。该过程就是需求工程当中最为重要的需求分析活动，又称建模与分析活动。

1. 优秀的需求哪些特性？试为每一个特性都举出一个不符合的示例。

优秀的需求特性：

1. 完整性：不需要做更多的扩展就可以充分的说明用户所需要的系统功能。

每一个需求的描述都应该包含开发人员设计和实现这项功能需要的所有信息

R6（不完整）：系统应该允许被扩展

R7（完整、较R8精确）:系统的调度算法应该允许被扩展

1. 正确性：真实的反映用户的意图；必须请需求的提出者予以确认
2. 精确性：描述仅包含必要的信息；简洁、清晰

R8（不精确）：在实现之后，系统的调度算法应该允许被扩展。

1. 可行性：由开发人员进行检查

需要进行一定的分析和研究，而不是单纯的凭借经验和直觉

必要的时候要通过开发原型来加以验证

示例：保证系统核心功能可以7×24小时连续运行

1. 必要性：满足用户的业务需求所必需的
2. 无歧义：每一项需求都应该有而且只能有一种解释

定义一个可以共同理解的词汇表（Glossary）

1. 可验证：通过分析、检查、模拟或者测试等方法能够判断需求是否被满足

示例：

实现各部门的公文流转无纸化、文档一体化、业务管理的规范化、自动化和网络化；

实现工作流程合理化、高效化，决策支持科学化、准确化；

统一办公流程、规范公文格式，加强信息交流和共享，提高工作效率。

不可验证的需求往往是因为描述模糊或者过于抽象，所以在进行需求的描述时要：让需求具体化、小心形容词和副词的使用、避免程度词的使用

1. 列出需求草稿中常见问题，需求工程师如何消除这些错误？
2. 需求并没有反映用户的真实需要

原因1：用户在表达自己的需要时，可能会在潜意识下进行一定的加工

进行问题分析，发现问题背后的问题

原因2：在人际交流当中，信息会发生自然的衰减，甚至扭曲

在需求传递给开发人员之前，请需求提出者进行仔细的检查和确认

1. 模糊和歧义的需求

原因1：无意中写出模糊和歧义的需求定义往往是因为选词造句不当

为项目中重要的词汇建立一个公共的可共同理解的词汇表

原因2：有意产生的模糊和歧义的需求定义往往是为了应付对需求持有不同立场的用户

在项目前景的指导下，促进用户之间的协商解决

1. 明显的信息遗漏

原因1：明显的信息遗漏，其主要原因在于项目的范围定义不当

加强对业务需求的处理

原因2：不明显的信息遗漏，往往是因为相关信息难以发现，如系统的环境依赖信息

该类问题是最难以解决的问题，只能靠需求工程师的经验来加以避免

1. 不必要的需求

其一是用户将之作为和开发人员谈判的筹码；开发人员代表的谈判技巧

其二是用户在交流当中，用户总是倾向于表达各种各样的需要

开发人员先定义明确的业务需求，根据业务需求进行用户需求的过滤和选择

其三是需求开发人员“画蛇添足”，添加“用户肯定会喜欢”的功能

需求开发人员要保持以用户为中心

**案例题：**

1. 解答一：（1）她没有仔细认真地分析问题；

（2）她没有及时跟相关人员交流信息，没能把握住有价值信息；

（3）她没能及时跟公司员工交流，引用过时的文件结构；

（4）她没有仔细研究分析新引进的系统的性能需求是否满足；

（5）她没有仔细研究新引进的系统的功能需求是否满足；

（6）她没有仔细研究引进的系统的质量属性，对外接口是否满足。

解答二：业务需求中没有和高层管理人员沟通好；她提出的用户需求没有和用户（自己的职员）沟通好，也没有向开发人员提出可行性、质量属性（可扩展性）等。

解答三：没有获得高层支持；财政部支持；下属抵制使用；信息不流通，文件使用不一致；要求的图形报告没有；不知道是否能修改

1. 业务需求：保持财务业绩与它的发展同步；有效地追踪客户账单和收据；降低成本；实行特价时能够知道是否有利可图，是否带动去他的销售；增加回头客。

解答：业务需求如BR。

BR1：实现客户账单和收据的有效追踪；

BR2：实现产品特价时的利润和相关销售情况检查；

BR3：实现一个客户数据库。

1. 先定义明确的业务需求，获得开发系统的必要性，根据业务需求，协调涉众的立场，限定问题的范围，指导用户需求的获取过程：和涉众沟通（即向业务人员了解相关的业务知识，业务流程；再和销售人员沟通，由于他们的顾客是流动的，不确定的，只能通过销售人员间接获取来自于顾客的用户需求，了解他们的背景和习惯），最后根据业务需求对用户需求进行过滤和选择，得到充分必要用户需求。
2. UR1：使用户可以根据系统的明确操作提示做出正确的反应；

UR2：用户插入银行卡后需要输入密码，得到验证后才可进行有效的具体操作；

UR3：在用户进入系统后，可以选择使用查询金额、存取现金、转账的功能；

UR4：用户能够正确、安全地退出系统。

1. SR1：（1） 系统显示用户插入磁卡的动态图像，正确标明插卡位置；

（2）用户根据提示，正确插入磁卡；

（3）系统读取磁卡卡号，界面显示输入密码的提示；

SR2：（1）对用户输入的密码，系统自动进行字符匹配；

（2）匹配正确的话，进入具体操作界面；

（3）匹配不正确的话，警告密码不正确，并提示再次输入；

SR3：（1）若用户选择查询金额图标和查询金额币种，系统读取银行数据库中用户对应的信息，反馈在用户界面上；

（2）若用户选择取款图标和金额币种及输入金额数目，系统读取用户请求，接受金额，修改数据库中该用户对应的信息，并提示成功与否；

（3）若用户选择存款图标和金额币种，系统弹出存款框，用户放入现金，系统接收现金并辨认真伪，并反馈存入金额数目，得到用户确认后，修改数据库中该用户对应的信息，并提示成功与否；

（4）若用户选择转账图标和金额币种并输入对方账号和转账金额数目，系统读取用户请求，修改数据库中所涉及到的用户的信息，并提示成功与否；

SR4：（1）用户选择退出图标；

（2）系统提示拔卡信息。

1. 性能需求：在用户点击图标后，系统在3s内作出反应。

质量属性：易用、可靠、安全、容错、可恢复、可维护。

约束：当用户输入密码次数等于3次后就不再提示输入密码，并自动锁定银行卡。

**第三章**

**思考题：**

1. 除了需求开发的四个活动和需求管理活动之外，需求工程中还有没有需要执行的活动？如果有的话，它们是哪些活动？给出你的理由。

有。项目管理（人力、资金的管理）、过程管理

（还有其他一些活动，例如：过程管理活动和项目管理活动。过程管理活动是跟踪项目开发过程，记录项目开发过程当中所遇到的问题或者教训等等。项目管理活动是管理项目开发的一系列问题与进度，管理人员配置，以求达到最该效益。）

1. 需求开发过程具有迭代特性，但是不是所有项目的需求开发过程都必须是迭代完成的？如果不是，试给出举例和理由。

不是，在问题域很简单或者非常成熟的情况下不需要迭代完成。

（不是，一般对于业务领域不熟悉的项目，需求是具有迭代性的，需要对业务领域的认知，有一个从认识到知识重构的过程。对于某些固定需求且熟悉的项目，比如学校课程的作业软件工程实践电梯系统，就不需要迭代开发）

（需求获取——>需求分析——>需求规格说明——>需求验证。当然并不是所有项目的需求开发过程是迭代完成的，比如：当某一项目开发过程中，用户需求非常简单，开发人员已经相当明确用户需求，这时，就不需要返回到需求获取阶段以继续用户需求的获取，这样，也就不需要迭代完成。当然，这种情况非常少见。）

1. 需求开发的迭代特性与软件开发过程的迭代式开发有什么关系？它们之间会相互影响吗？如果会，那么有哪些影响？

需求开发的迭代性指的是对于开发者对知识的认知水平在某一点上，发生重构，使得知识体系复杂性下降，而继续积累知识的过程。

软件开发的迭代性指的是在软件生命周期整体开发迭代，针对变更的需求或者新增的需求一种减少风险的开发模式。

需求开发迭代不会导致软件开发过程的迭代，但有时会有影响。（需求开发的迭代特性只是软件开发过程的迭代式开发的一个子过程，软件开发过程是一个相当庞大的工程，需要在软件开发过程的各个阶段都需要进行开发工作的迭代，当然也包括需求开发中的迭代。它们之间互相影响。如果需求开发中的迭代不能很好地完成需求分析任务，就必将影响到软件开发过程的其他迭代阶段的进行。）

增量式：需求->需求->需求->。。。->需求->开发。。。 无需求迭代

演化式：需求->开发->需求->开发->需求->开发。。。 有需求迭代

1. 需求工程细节知识的实践性对不同项目的需求开发过程的差异性有没有影响？如果有，说明影响是什么，如果没有，说明是哪些因素产生了不同项目的需求开发过程的差异性。

没有。问题域的特性导致了不同项目的需求开发过程的差异。

丁老师解答：没有影响。其实是需求开发过程的差异性一定程度上导致了细节知识的实践性。现实世界问题的复杂性和差异性主要导致了需求开发过程的差异性

**第四章**

**思考题：**

结合复习题4、5、6三个题目的答案，说明需求获取的内容和需求获取的来源是怎样影响到需求获取的方法选择的？请一一举例。

4、5、6三个题目的答案：

需求获取的内容：

需求

1. 需求：需求获取的主要对象，期望解决的问题列表。通常体现为涉众的问题、期望、观点、看法和态度等。
2. 问题与描述：用来承载和解释需求的问题与特性，现实世界的业务运行状况。
3. 环境与约束：属于一种特殊的问题与特性，限定了解系统部署的环境和条件。

需求获取的来源：

问题域的知识

1. 涉众：用户；客户；领域专家；市场人员、销售人员等其他用户替代源
2. 硬数据：登记表格、单据、报表等定量文档；备忘录、日志等定性文档
3. 相关产品：原有系统；竞争产品；协作产品
4. 重要文档：原有系统的规格说明书；竞争产品的规格说明书；协作产品的规格说明书；客户需求文档
5. 相关技术标准和法规：相关法律、法规及规章制度；行业规范，行业标准；领域参考模型

需求获取的方法：

（1）传统方法：问卷调查、面谈、硬数据分析、文档检查、需求剥离等

（2）集体获取方法：头脑风暴（Brainstorming）、专题讨论会（Workshop）、JAD等

（3）原型

（4）模型驱动方法：基于场景的方法、基于目标的方法和多视图的方法

（4）认知方法：任务分析（Task Analysis）、协议分析（Protocol Analysis）等

（5）基于上下文的方法：观察、民族志（Ethnography）和话语分析（Conversation Analysis）

**第五章**

**思考题：**

系统中每一个问题解决方案的边界是如何集成建立系统边界的？集成过程中对各个问题解决方案的输入/输出和功能是如何处理的？试举例说明。

明确每一个解决方案需要具备的功能特征，根据这些功能特征，分析解决方案需要和周围环境形成的交互作用，定义解决方案的边界。

把每一个解决方案的输入/输出列出来，如果一个解决方案外部的输入来自于同一个系统当中另一个问题解决方案的输出，即系统的内部，就可隐去。如果不是，就保留输入/输出。

经过问题分析之后就可以得到更高层次解决方案及系统特性，它们清晰的定义了问题解决方案的功能和边界。将所有问题的解决方案进行综合，就可以得到整个解系统的功能和边界。

为了描述系统的功能和边界，通常会使用上下文图或系统用例图。

对于用例图描述要解决的方案，一个用例就是一个功能，将其保留；对于上下文图，将功能提取出来，功能分解。

**案例题：**

1. 不做工作陈述的风险：
2. 在获取需求时，用户往往从各自的立场出发考虑问题，提出相应的功能需求。如果没有工作陈述，用户就不会从共同的方向上考虑和理解问题，对系统的期望也就产生了较大的差距。
3. 没有工作陈述，就等于在用户之间发生需求冲突时，就没有可以用来指导并且调节协商的项目前景，冲突问题也就很难解决。

风险：1需求理解错误2不能按时完成（超期超资）3做出来的不是想要的

定义范围的必要性：1.加强用户和开发人员的理解，定义一致的理解2.降低风险

解答二：省略工作陈述的风险是不能明确项目的前景和范围。如果省略了工作陈述的话，你就不能和用户进行很好的沟通与交流，这样，项目的问题也就不能明确，即，开发人员无法与涉众对问题达成共识；无法明确问题，也就无法发现正确的业务需求，无法定义良好的解决方案及系统特性，继而无法明确项目的前景和范围，这样就会造成项目的不稳定甚至失败！

解答三：通过准确的工作陈述来定义项目范围，可以帮助涉众建立现实的期望，包括第一版范围，后续版本范围、限制与排除。第一版范围概述产品的第一个版本中实现的主要特性，描述产品的质量特性，可以为不同类别的用户提供预期利益。后续版本能够实现更多的需求和特性，并完善最初的功能。尤其要说明的是，管理范围蔓延的方法之一，是定义项目包含的需求与不包含的需求之间的界限，应该列出涉众可能希望得到，但不在产品或其某个特定版本计划之内的功能和特性。

**2.**

|  |  |  |  |
| --- | --- | --- | --- |
| 问题 | 业务目标 | 高层解决方案 | 系统特性 |
| 帐户太多，工作量太大 | 减少检查人员的工作量 | 能够快速、自动查询客户账户 | 建立一个数据库系统用来存放客户账户信息 |
| 降低工作复杂度 | 能够分析一个客户是否为问题账户 | 根据特定的判定问题账户的算法检索辨别出问题账户 |
| 需查阅账户的大量历史数据 | 能够给出一个问题账户的三年内的历史数据 | 工作人员能够检查该账户的三年内的历史数据 | 能够按账户号查询该账户三年历史数据 |
| 问题账户所占比例没有显示 | 能够计算问题账户所占比例 | 即时显示问题账户所占比例 | 根据查询结果，自动计算并显示问题账户所占比例 |

**3.**

（需要根据当时的资金，时间，以及和用户、技术人员的协商结果而定。）用户决定

**4.**

解答：她现在遇到的问题有：

|  |  |  |
| --- | --- | --- |
| 问题 | 业务目标 | 高层解决方案 |
| 不能有效地从信息部门获得工资和个人数据 | 减少从信息部门获得工资和个人数据的时间；  度量标准（Scale）：一次从信息部门获得工资和个人数据的时间；  计量方法（Meter）：检查信息部门数据库日志；  理想标准：减少50%；一般标准：减少30%；最低标准：减少20%； | 由软件从信息部门的数据库中检索出工资和个人数据，减少所需信息获取的时间 |
|
|
| 雇员数据太过分散，而且不能及时正确地更新 | 集中雇员数据，并且正确更新 | 由软件来分析雇员数据的各种特征，及早识别出数据所在位置；或由软件集中处理雇员数据，及早识别出不准确的或没有及时更新的数据，提交人工处理或自行更新 |
| 计算复杂 | 降低计算的复杂性 | 由软件来处理投资和退休假定的计算的复杂过程 |
| 雇员信息不能得到及时有效正确的更新 | 及时有效正确地更新雇员信息 | 由软件来分析个人数据的准确性，及早识别出不准确的个人信息，提交人工处理；  或定时更新数，提高数据的准确性； |
| 计算中可变条件的复杂性 | 降低计算中可变条件的复杂性 | 由软件来处理计算中可变条件的复杂性，降低出错率 |

系统特性：

* 根据信息部门提供的数据库查询工资和个人数据；
* 根据原始数据重新整理数据并更新；
* 提交查询信息；
* 创建投资和退休假定的计算过程；
* 通过公司的内联网访问系统，根据个人情况更新信息；
* 模拟计算中可变条件的变化；
* 提供最灵活的福利方案。

重要的约束有：

|  |  |
| --- | --- |
| 约束源 | 约束 |
| 操作性 | 雇员信息必须有备份 |
| 设备预算 | 有自己已有的系统上开发 |
| 技术要求 | 应用面向对象的方法 |
| 行政要求 | 需要信息部门的信息 |
| 系统 | 空间不应该超过20M字节 |
| 环境 | 安全性 |

**第六章**

**思考题：**

**1.**

用户是最终使用和操作产品的人，他们是使用软件的目的是为了更好的完成自己的任务，满足组织的目标要求。因此，一个成功的软件要能够协助用户有效的完成实际工作，用户也就自然应该是需求获取的主要信息来源。需求工程师需要了解用户实际工作的开展状况和用户希望软件系统能够给予他们的帮助。

用户参与是以用户为中心的设计方法的核心思想，它要求开发者建立和用户的直接联系，尽早地关注与用户和用户的执行过程，通过及时获得用户的反馈来调整软件设计，以完成高质量的设计。另一方面，用户参与就是反对通过和市场人员、管理者等中间媒介来了解用户。

在以用户为中心的设计方法中，用户需要参与软件开发的全过程，并且对最终软件设计和质量具有非常重要的影响，所以在该方法中参与用户的选择和普通的涉众代表采样有所不同，要吧他们区分开来。

**2.**

他们建立了良好的合作关系后，可以降低风险。

理解用户：对用户的基本特征描述（个人特征、工作特征、少数会涉及地理特征）

评估用户：优先级评估、风险评估、共赢分析

与用户协商，处理用户间对于项目期望冲突

用户的个人特征和工作特征的描述可以帮助更好的确定功能需求。

**案例题：**

**2.**

* 灌输需求知识：（用户参与的重要性）
* 涉众的重要性：涉众是指所有能够影响软件系统的实现，或者会被实现后的软件系统所影响的个人和团体。

**3.**

* 找出问题产生的根源，分析问题背后的问题
* 涉众分析，找出冲突所在，找出矛盾的焦点
* 解决信息系统部门与非信息部门之间的冲突

解答：首先，需要细分涉众类别，这里用户，需求工程师和程序员都属于涉众类别。需要分析他们各自的赢利条件，以在相互妥协中尽力实现一个共赢的结局。分析涉众的关注点和兴趣取向。 了解涉众的个人特征和工作特征，以便对软件系统的功能进行合理的调整。选择合适的代表参与项目的开发。定期举行讨论会，让用户知道项目的进展情况。优先级评估，风险评估，共赢分析…

**4.**

根本没有寻找涉众，没有涉众分析，使用的是组织级的系统，应该进行涉众分析。

**5.**

见6.3.2

例：个人特征：年龄：老年人 字大

工作特征：电脑使用程度

地理和社会特征：文化背景：中国和台湾

关注点和兴趣：反对还是赞同

目标期望：领导的目标

被影响程度：使用频率

力量程度：是否可以影响项目实施，领导

对个人特征和工作特征的描述可以帮助更好地确定功能需求；

也可以帮助形成对涉众类别的理解

**6.**

解答：（1）选择面谈对象的时候采用随机抽样，从5个阶层以及生产、会计、营销、系统、物流各选择2-3名客户参与面谈。高层管理均要参加面谈。因为在选择面谈的时候要力争均衡的收集用户的需求，因此要涉及各方面受系统影响的人。

采样的规则：控制人数（4～8），教材上册，6.3.4

（2）高层管理的人最先面谈。然后是系统层。其余层的面谈对象根据实际情况可以先后安排面谈的时间，不一定要分先后顺序。

跟高层管理人员进行面谈，采用漏斗结构，因为各个高层管理人员对各自管理的层次从大体上有准确的把握，有助于开发人员首先获取对项目的广度方面的认识，也能获取一些较为详细的信息。跟具体部门人员进行面谈，采用菱形（必要时，金字塔）结构，因为这种面谈较为具体，问题常为封闭式问题，这样有助于分析人员获得深度认识。

基本规则：（1）先业务需求，后用户需求，所以先领导后普通；

（2）开始漏斗，领导漏斗

（3）普通用户菱形，必要时金字塔

面谈的结构及其特点：教材上册，7.2.2

**7．**

解答：

1. 定量硬数据：

发货及收货的明细表

货物的中转表

拖拉机和仓库的使用情况表

定性硬数据：

日常业务描述文档

描述发货人、收货人和承运公司的伙伴关系文档

参考硬数据的类型：教材上册，6.5

（2）将这15年公司的情况用图表表达出来，形成对15年以来公司状况的认识，获取生产情况的时候将大致相同的年份列出来，采样时候只需要在大致相同的年份中抽取一份作为样本。

参考采样规则：教材上册，6.6

### 第7章 需求获取方法之面谈

1. **在重新浏览面谈日程的时候，你发现有几个问题看上去不合适。下面是准备问Sampson纸产品公司销售经理的原问题。这家公司想把它的一些销售信息放到Web上去，以便经理们可以交互地评论它，从而优化他们的销售方案。用更合适的方式，重新写下面的问题。**

（有错误问题：同时问两个问题；隐含和暗示；提问题时上下文相关；问的问题牵扯到了被问的对象，如最后一题的陈旧）

* 1. 你的下属告诉我，你非常渴望有一台计算机。这是真的么？

你对计算机的使用态度如何？

你认为作为一个销售经理，是不是应该拥有一台计算机？（诱导性问题）

* 1. 我是这个领域的新手，我有没有忽略什么呢？

我问的问题如何，你有什么要补充的么？

我是不是还忽略了什么？（上下文无关问题）

* 1. 你在销售计算中最常用的信息资源是什么，使用频度如何？

将两个问题分开

1、你在销售计算中最常用的信息资源是什么（双筒问题）

2、使用频度如何？

* 1. 其它销售经理认为，把一些月度销售商品放到Web上，然后做趋势分析，将会是一种主要改进，你同意他们的做法吗？

你和其他经理一样，都同意。。。，是吗？

你认为把一些月度销售商品放到Web上，然后做趋势分析会是一种改进吗？（诱导性问题）

* 1. 没有比你现在使用的陈旧的方法更好的销售方案吗？

对于现在的销售方法，你有什么更好的改进方法么？

还有比目前方法更好的销售方案吗？（上下文无关问题）

1. **作为系统分析项目的一部分，需要为生产数字钟的Chronos公司更新自动化会计功能。你将要同首席会计Harry Straiter面谈。写出4到6个涉及他所使用的信息资源、信息格式、决策频度、需求的信息性质和决策样式的面谈目标。**
   1. 说明你将如何联系Harry以安排一次面谈。

（打电话，预约：联系个人，安排一次会见，内容，选个时间，让他找个时间，安排个地点）

提前打电话或者发送电子邮件通知Harry，告知面谈内容，商定面谈时间和地点；提前通知可以给Harry时间去考虑面谈事宜。如果要进行一次深入的面谈，可以把问题通过电子邮件提前发给Harry，让他有时间仔细考虑答复。（P120）

* 1. 说明在这场面谈中你会使用哪种面谈结构？为什么？

（首席会计师**，leader，专家型**的人面谈结构同普通用户不同）

漏斗结构，适合领导专家（根据上课笔记）

* 1. Harry有3个下属也使用这个系统。你和他们面谈吗？为什么？

（涉众分析中不同涉众有不同特点，下属和他之间有没有差异，有差异则要；没差异，则为什么）

应当面谈，因为Harry和其下属对软件系统的开发和应用具有的发言权和决定权不同，属于不同的涉众类别。Harry属于领域专家，而其下属属于该系统的用户，下属和领导使用这个系统的目标不同，下属是为了更好的完成自己的任务，满足组织的目标，他们是主要的信息来源，所以应当面谈。

应当面谈，因为下属和领导应该具有不同的目标，而这些目标是领导不能提供的

考察点：涉众的分类

* 1. 写出3个开放式问题，在面谈前通过电子邮件寄给Harry。用一句话解释为什么应当由人而不是由电子邮件来指导面谈？获取许多语言文字之外的其它信息，如声音动作语气等

（三个开放式问题随便写,纸面记录和其他几种记录方式的优缺点，人的信息传达有几个方面，每个方面各占多少。只靠邮件就只剩文字了，交流中只剩文字的手段了，会产生什么缺点）

（好像很多不知怎么一句话概括）

由于面谈中可能会实现很多目标，涉及很多复杂问题，所以面谈一般应该由人而不是电子邮件来来管理（P120）。

笔录的优点有：

* + 使会见者专心和集中精力；
  + 帮助回忆重要的问题；
  + 表现会见者对面谈的兴趣；
  + 表明会见者是有准备的。

虽然笔录有一些好的优点，但也有一些缺点：

* + 丢失很多被会见者在谈话中表现出来的语调、停顿等语音信息；
  + 做笔记时，会让被会见者说话犹豫；
  + 造成对事实注意过多，而对感觉及观点注意过少。

录音和摄像的优点有：

* + 记录了更多的信息；
  + 会见者能轻松地倾听并更快速地做出响应；
  + 可以完整的重现面谈过程。

录音和摄像也有很多的缺点：

* + 被会见者可能会紧张，回答不自在；
  + 数据采集的代价较高；
  + 事后进行信息寻找时难以定位。

1. **对第6章的案例题6，说明Phil应该怎样开展他的面谈工作？包括：面谈对象选择的先后顺序，每次的面谈结构。说明原因。**

（列了需求的计划

打算安排几轮面谈，每次的参与人员

每次面谈结构，可能的话可以安排第三个轮次的面谈,分析原因

第一个轮次获得前景和范围

第二个轮次详细

第三个轮次验证需求）

进行三轮面谈，具体安排如下：

第一轮面谈：

面谈对象：高层管理员

面谈结构：漏斗式结构

因为第一轮面谈主要是为了获得项目的前景和范围，通过探讨一些高层次的问题来和项目目标推导出业务需求，并根据问题帮助确定系统高层次的解决方案和系统特性，从而到了项目的前景和范围文档。而这种问题的讨论需要高层的管理员和对整个业务了解的人，所以第一轮的面谈对象是高层管理员。

根据面谈结构的特性，漏斗式的面谈结构适合于领导和专家这样的被会见对象，所以选择漏斗式结构。

第二轮面谈：

面谈对象：管理层以下的员工

面谈结构：菱形式结构？

第二轮面谈的目标是为了获取详细的需求。详细的需求涉及系统的各个层次，而各个层的工作目标和工作特性各不相同，所以需要要各个层次的工作人员进行面谈。

第三轮面谈：

面谈对象：各个层次的职员

面谈结构：长序列的封闭式问题？

第三轮面谈的目标是为了验证已获取的需求。分析汇总了获取的需求后，将获得的需求分类罗列后，根据具体不同的需求需要向各个层面的涉众确认验证已经获取的需求，保证需求的正确性，完整性，一致性。

由于需求已经基本确定，所以采取封闭式问题。

1. **从你进门到现在，面谈对象Max Hugo一直在翻阅文件、看手表、点燃和掐灭香烟。根据你看到的有关面谈对象的情况，可以猜出Max很紧张，因为它需要做其它事情。用一段话描述，为了使面谈能在Max全神贯注下完成，你将如何处理这种情况。（Max不能在另外一天重新安排面谈。）**

（轻松气氛，了解其兴趣点，使用开放式问题；如何缓解紧张气氛，开放式问题可以使人放松；转移注意力，漏斗结构)

可以采用一些非常一般的，轻松地，开放式的问题。这种方式可以打开一个面谈的局面，建立一个和MaxHugo面谈的轻松气氛。通过仔细听取前期的回答，可以帮助我们适应后面的问题。早期开放式的问题也能够帮助解释被会见者的态度，价值观和信仰，这些能够帮助了解被会见着如何使用信息及其对组织中其他人有何看法。

如果开放式问题未能解决MaxHugo的紧张，我们可以转移话题，谈一些其感兴趣的话题，转移其注意力，建立与MaxHugo之间的信任和信赖，再将话题转移回面谈内容。

使用漏斗型面谈结构。

1. **下面是系统分析团队的一名成员提出的第一份面谈报告：“在我看来，面谈进行的很好。我和他就这个问题聊了一个半小时。他告诉我有关公司的所有历史，很有意思。他也提到，自他来到该公司的16年间，公司没有任何变化。我们不久将再次举行会面，以及结束这次面谈，因为我们还没有深入研究我准备的问题。”**
   1. 试评论这个面谈报告。假设你要团队成员使用图1提供的报表，那么他漏了什么主要信息？

（打算干嘛面谈目标，实际有没有）

面谈时间稍长，而且控制不佳。

遗漏了关于“最新建议的系统的观点”

* 1. 什么信息对面谈报告来说是无关紧要的？

（面谈目标和内容无关）

有关公司所有的历史。

* 1. 如果真的发生了报告中提及的情况，则必须向队友提出哪3个建议，以帮助他更好地举行下一次面谈。

（三个建议的重点是那些是帮助控制面谈主题的）

1．控制面谈的过程。面谈开始的时候可以通过例如谈公司历史来酝酿一下交流的气氛，但是不能偏离主题。如果长时间的谈论不相关的信息的时候，需求分析人员就可以委婉的提醒面谈对象，并重新切回正题。

2．注意保持面谈的主题。针对每个面谈的目标，要在面谈的过程中安排合适的提示，逐一引导面谈对象对各个主题的叙述。

3．总结面谈的要点，注意此次面谈过程的成功和失误，明确下次的目标，以便为下次面谈做充分的准备。

|  |  |
| --- | --- |
| 面谈对象：SalDomask 日期：3月3日  会见者：S.Cabbot 主题：计算机使用  面谈的目标：找出关于计算机使用的态度；  获得用户的使用估计；  看最新建议的系统的观点是否满足目标吗？  下次面谈的目标：  找出Sal怎样看待系统支持部门。  找出下一个面谈对象的观点。 | |
| 面谈的要点：  Sal说道：“计算机是我的朋友。”  “一直”都在用计算机。  迫不及待地要熟悉新系统。 | 会见者的观点：  对了解更对有关系统如何促进工作感兴趣。  如果不使用计算机进行工作，会感到枯燥。  将成为新系统的热情支持者/促进者。 |

**6**.**假设现在由你来负责所在学校选课系统的需求工作，现在需要你来安排一次群体面谈，你打算怎么做？**

（群体面谈的准备阶段）

计划面谈

1.确定参与人员（涉众、主持人、负责人、分析人员、记录人员、观察员 ）

2.安排会谈时间 （全职的2－4天参与会议 ，拟定一份议程 ）

3.选择会谈地点 （充足的空间，道具支持，良好的餐饮服务 ）

4.准备会谈内容 （面谈的主题和范围，会议的议程，需求的预期和会谈的目标，各种材料）

**7.Cab Wheeler 是小组新解雇的需求工程师。Cab一直觉得问卷调查没有用。现在你要为Mega Trunks公司做一个系统项目，Mega Trunks是一家在130个城市有分公司和职员的国际运输公司。你想使用问卷调查表引出一些对当前系统和建议的系统的看法。**

（1）根据你对Mega Trunks的了解，给出3条有说服力的理由，说明为什么应该在这个研究中使用调查问卷表。

（三个理由，根据项目的具体情况 地理分散（涉众众多，地理分布不均匀），调查问卷的好处）

1. 系统的涉众在地理上是分布不均的，分布在130个城市
2. 系统的涉众数量众多有130个城市的所有分公司的所有职员，而了解所有涉众的统计倾向是非常重要的，所以只有问卷调查才能做到统计
3. 想要了解一些涉众对当前系统和建议的系统的看法，因为了解当前的总体状况，对进行一项探索性的研究，在确定具体方向有很好的参考价值。

面谈在这三个方面都有局限性，无法执行面谈，而调查问卷正好可以处理这些面谈的局限性情况。

问卷调查的优点（Google）

1）问卷调查法的最大优点是，它能突破时空限制，在广阔范围内，对众多调查对象同时进行调查。

2）便于对调查结果进行定量研究。

3）匿名性。

4）节省人力、时间和经费 。

（2）Cab在你的劝说下同意使用问卷调查表，但是极力主张所有问题都采用开放式问题，免得约束回答者。用一段劝服劝服Cab，封闭式问题是有用。一定要指出每种问题类型间的折中考虑。

（封闭式也是有用的，开放式和封闭式的优缺点就行）

各有优缺点（如下）：开放式 优点：

让被会见者感到自在；

会见者可以收集被会见者使用的词汇，这能反应他的教育、价值标准、态度和信念；

提供丰富的细节；

对没采用的进一步的提问有启迪作用；

让被会见者更感兴趣；

容许更多的自发性；

会见者可以在没有太多准备的情况下进行面谈。

缺点：

提此类问题可能会产生太多不相干的细节；

面谈可能失控；

开放式的回答会花费大量的时间才能获得有用的信息量；

可能会使会见者看上去没有准备。

封闭式 优点：

节省时间；

切中要点；

保持对面谈的控制；

快速探讨大范围问题；

得到贴切的数据

缺点：

使得被会见者厌烦；

得不到丰富的细节；

出于上述原因，失去主要思想；

不能建立和面谈者的友好关系。

开放式和封闭式问题各有优缺点,需要根据实际情况，考虑它们的优缺点，进行两种基本问题的结合使用：

事实和问题的范围比较确定，只希望被会见者进行选择和确认时，可采用封闭式问题；在被调查者对事实和问题的掌握比较有限，希望被调查者提供丰富信息时，使用开放式问题。

|  |  |  |
| --- | --- | --- |
| 开放式问题 |  | 分封闭式问题 |
| 低 | 数据的可靠性 | 高 |
| 低 | 适应时间的效率 | 高 |
| 低 | 数据的精度 | 高 |
| 广 | 广度和深度 | 窄 |
| 多 | 需要的技巧 | 少 |
| 难 | 分析的难易度 | 易 |

**第八章 需求获取方法之原型**

**案例题(第一道题目,第三道题目,第五道题目第二问 不会做额)**

1. **“每当我认为已经获取用户的信息需求时，他们却已经发生了变化。这就像试图射中一个运动目标。在半数时间里，我认为甚至用户自己也不知道需要什么。”Flo Chart说。他是2Good 2 Be True公司的需求工程师，该公司负责为几家制造公司的营销部门调查产品的用途。**
   1. **用一段话向Flo chart解释，原型化方法怎样才能帮他更好地定义用户的信息需求。**
   2. **用一段话评论Flo Chart的观察：“在半数时间里，我认为甚至用户自己也不知道需要什么。”一定要解释原型化方法怎样才能真正地帮助用户更好地理解和阐明他们自己的信息需求。**
   3. **用一段话向Flo Chart建议：一个具备原型特征的交互式Web站点缘何能解决Flo关于捕获用户信息需求的问题。**

解答：（1）答案主题：为什么要原型以及原型的作用

（1）根据需要确定原型类型；（2）进行原型开发；（3）获得用户反馈；（4）定义所得需求

（2）答案要以“隐含知识”和“用户表述时的主观加工”为主题

（3）原型化方法利用直观化的界面来最快程度的得到用户的反馈，通过用户的反馈

来获知其实际的需求

1. **“我有一个绝妙的主意！”Bea Kwicke宣布，他是系统团队的一位新来的需求工程师，“让我们跳过所有的SDLC垃圾，直接为一切设计原型。我们的项目会进展的更快，还可以节省时间和金钱，并且所有的用户会感到我们似乎很在意他们，而不是连续几个月不与他们交谈。”**
   1. **列出你（作为与Bea同一个团队的成员）用来劝阻她不要试图放弃SDLC，而直接为所有项目设计原型的原因。**
   2. **Bea对你所说的话很失望。为了鼓励她，用一段话向她说明，你认为适用于原型化方法的情形。**

解答：（1）主要原因：原型仅仅是开发当中使用的一种手段，它利用得当可以加速开发的进程，但不能代替软件开发中的所有工作。

原型开发最大的缺点就是：成本太高，高的让人难以接受。所以原型方法只在必要的时候使用原型方法。通常来说，如果用户需求出现了模糊，不清晰，不完整等具有一定不确定性的特征，就可以考虑使用原型方法。

原型方法的复杂性使得它会给项目引入了新的风险。

（2）情形见下表，尤其是其中红色的部分

|  |  |  |
| --- | --- | --- |
|  | 废弃型 | 演化型 |
| 水平型 | 阐明并细化用例和功能性需求  识别遗漏功能  研究用户界面方法 | 实现核心用例  根据优先级实现其他用例  使得系统适应快速变化的需要 |
| 垂直型 | 演示系统可行性 | 实现并扩充核心功能  实现并扩充核心算法  测试并调整性能 |

用户需求出现了模糊，不清晰，不完整等一定不确定性的特征，就可以使用原型。

如果开始是以缺陷需求为起始点，需要不断调整的情况，就可以使用探索式原型开发

如果开始拥有清晰地用户需求，但是开发者对这些需求的实现方法，实现效果和可行性没有太大的把握，则可以使用实验式原型的方法

如果开始有清晰的需求也有项目积累下来的原型资产，这样的情况可以使用演化式原型开发

1. **说明下列典型的应用情境适合使用哪种类型的原型？**
   1. **人机交互界面——信息表格或者报表。**
   2. **人机交互界面——具体功能的设置。**
   3. **人机交互界面——任务的执行过程。**
   4. **功能探索——任务的功能内容。**
   5. **功能探索——任务的执行过程。**
   6. **功能探索——任务的执行效果。**

解答：

1. **Itall多年来一直担任Tun-L-Vision公司的系统分析员。在你加入该系统分析团队以后，建议在目前项目中把原型化方法作为SDLC的一部分，Itall说：“当然可以，但是你不能太在意用户所说的话。他们也不知道自己需要什么。我会做原型化工作，但是我不会‘观察’任何用户。”** 
   1. **在不明确否决Itall的前提下，尽可能巧妙地说明原型化过程中观察用户反应、用户建议和用户创新的重要性的原因。**
   2. **用一段话描述，如果系统的某部分已经被原型化，并且在后续系统中没有考虑用户的反馈信息，可能会出现什么情况？**

解答：（1）通过观察用户的反应会得到比较多的信息，比如说观察到用户总是出错则说明设计有问题，用户在某个界面停留很久这就说明软件的导航有问题，通过观察发现用户老是从一个位置移到另一个位置，说明界面中按钮放置的有问题，有的时候用户使用的方式超出了我们的想象（用户创新），像这些都要通过观察得到。在评估中，用户会对原型系统的人机教会和功能设置提出建议，这些建议可以帮助开发者们改进，改变或调整原型，从来可以是原型更接近于它的目标实现。对于用户的创新则是用户潜在的需求，这些可以通过观察还有用户的反馈中得到，做到以上，我们可以获得很多信息，使我们的原型更加完善。

（2）如果系统的某部分已经被原型化，但是在后续系统中没有考虑用户的反馈信息，这个原形都不能算是一个符合要求的原型。这样会导致开出来的原型根本就不符合用户需求，开发出来以后用户不满意可能会受到用户的抵制。可能在后期才被发现，开发方需要做很大的调整修改，导致项目延期，严重者可能会导致项目的失败。

1. **Nordic Designs 是一家专营Scandinavia 当代家具的连锁企业，它已经发布了一则夸耀其配送信息系统原型的公司简讯。简讯报道声称：“我们的配送信息系统原型一发布就投入使用了。绝对没有任何修改的必要，经理们说它是追踪家具配送的最佳解决方案。不久就可以你们商店中接触原型了。”**
   1. **这则报道的作者对原型化方法概念明显存在什么样的误解？用一段话解释它。**
   2. **如果用户期望原型“绝对没有任何修改的必要”的话，列出原型设计者可能会面临的问题。**

解答：（1）这则报道中提到“我们的配送信息系统原型一发布就投入使用了”可以看出作者误解了一点：开发出的原型不是最终的软件，原型不能直接发布使用，我们使用原型的目的是获取需求的内容，而不是获取原型的代码，原型代码最终应该是会被抛弃的。作者还说“绝对没有修改的必要”这句话显然有问题，原型开发的过程中腰不断地根据评估者反馈的不足进行原型的修改，调整完后还要准备再次原型评估，如果不能通过，则在根据反馈，观察进行原型修正，所以不能说“绝对没有任何修改的必要”。

（2）首先原型是本来就是用来获取需求的，最终代码一定要被抛弃，不然开发出来的软件质量会很差。 如果用户期望原型“绝对没有修改的必要”的话，也就是说一次就获取完需求，显然这样的方法是不可行的，不能获取到完整明确的需求，这样会导致配送系统漏洞多，不能满足用户的需求，不受用户的欢迎甚至抵制，严重的可能影响到业务 花费大力气在原型上，时间花费过大

1. **下面这段话是在Fence公司的经理与系统分析团队的会议上听到的：“你们告诉我们原型可以在3个星期以前完成。但现在我们还在等。”**
   1. **用一段话来评价快速提交原型的重要性。**
   2. **原型化中可能有哪些难以管理的因素？试列举它们。**
   3. **有哪些方法可以帮助控制原型开发的过程和速度？**

解答：（1）快速提交原型可以避免对原型开发工作投入太多的工作，这样开发团队就不会消耗过多的时间和过大的成本，避免了最后被迫只能匆匆忙忙实现一个产品，甚至只是交付一个原型的情况出现。

（2）创新性内容，例如创新性思想 都是些在使用原型方法之前没有想到的内容，有的还可能是用户的潜在需求，要通过不断地用户和原型交互，观察用户才可能发现，很难管理

探索式原型时候，如果探索的内容太多就很难管理

时间成本：使用原型方法的时候如果选择方法不正确，起初设计不够健壮等问题可能会导致原型开发占用时间长，成本高，风险大，这里需要很好的原型管理

尝试东西的可行性，新的想法

* 1. 如果是探索内容太多，尽量分散在不同的项目里，不要集中在一个项目中。

根据需要，花最小的代价来创建原型，如果花的最小代价，显然就能很容易的控制管理（比如说：如果是界面原型，可以用纸质界面可以节约代价；如果要验证某个可行性，功能一般都是比较固定的可以选择合适的，再比如如果选择垂直原型一定是功能范围比较少的，否则很难控制）

# 第九章 需求获取方法之观察与文档审查

## 思考题：

1.观察用户工作总是困难的。它通过使你和用户都感动不舒服。为了确保由于你的访问而不至于使用户的行为发生改变，你应该怎么办？为了使观察看起来更自然一些，你应该怎么做？

**二玉哥哥语：**书上没有，属于日常经验，让别人感觉到你的访问不发生困难的手段

**参考答案：**

1. 每次去用户的工作场所帮助用户干活，使得用户感觉自然
2. 在特定的观察之前就经常去用户的工作场所，等到进行观察时，用户对需求工程师的到来已经习以为常。
3. 在去观察之前做足准备功夫，熟悉客户的性格特点、个人嗜好、生活习惯和工作风格等，观察中可以适时与客户交流，以提高客户对需求工程师的好感度和熟悉度来降低客户的不自然
4. 观察客户的时候不要太明显，尽量不使客户察觉到。

2.在需求获取阶段，需求工程师收集了大量的硬数据样本，解释这些样本的类型以及它们适用于哪种文档审查方法。

**二玉哥哥语：**书上就有，都可做成复习题，难度不够-\_-^^^^

**参考答案：**

|  |  |  |
| --- | --- | --- |
| **文档类型** | **文档审查方法** | **描述** |
| 相关产品的需求规格说明 | 需求重用 | 分析相关产品的规格说明，发现可以移植到新产品中的需求信息，进行需求的重用：   * 问题与信息 * 用户界面特征 * 业务需求、组织策略、政策法规 |
| 硬数据 | 文档分析 | 阅读研究得到的硬数据，从中发现需求信息   * 问题域信息 * 工作流程 * 业务细节 |
| 客户的需求文档 | 需求剥离 | 抽取客户需求文档中的需求描述   * 粗粒度需求 |

## 案例题：

1. Ceci Awill说：“我想我能记得他所做过的大部分事情。”Ceci准备与OK　Ｃorral公司战略规划副总裁Biff Weblldon进行面谈。OK　Ｃorral是一家拥有130间牛排连锁店的公司。“我的意思是说，我有好的记性。我认为听他说什么比看他做什么更重要。”

作为需求工程团队的一员,Ceci Awll向你诉说了他要写下在面谈中对Biff的办公司和Biff的活动进行观察的愿望。

(1) 用一段话来说服Ceci，在面谈时仅仅倾听是不够的，观察和记录所观察的内容同样是很重要的。

**二玉哥哥语：**仅仅倾听时不够的，很多东西是用户没有办法直接告诉需求工程师的，需要观察才能得到

**参考答案**：答题点见上。具体话语依据个人组织

(2) Ceci似乎接受了你认为观察时很重要的观点，但是不知道该观察什么。列出需要观察的项目和行为，在每一项行为的旁边用一句话指名Ceci通过观察应该得到的信息。

**二玉哥哥语**：稍微有一点儿超出书上的范围，要知道有几个部分的内容需要处理：暂时性、局部性和模糊性。

**参考答案**：

* + 观察客户的个人信息：年龄、着装、语言、身体语言等——得到用户的性格、喜好偏向和态度等信息
  + 观察客户的周边环境：办公室的位置及摆放、桌上物件的数量和摆放、书橱的书和报刊等——得到用户工作习惯和生活风格等信息
  + 观察客户的日常行为：——查看客户是否有异常行为；有无重要的事件和工作过程等
  + 观察用户的实际工作：——查看规章制度与之是否相符

2. “我知道你有很多材料。那些材料里到底有什么？”Betty Kant问道，她是MIS特别工作组的负责人。MIS特别工作组是你的系统团队联络Sawder家具公司的桥梁。你拖了一大堆材料，正准备离开这栋楼。

“哦，是过去6个月的一些财政决算、生产报表，还有Sharon给我的一些业绩报表，业绩报表涵盖了过去6个月的目标和工作业绩。”你在回答时，有些纸掉到了地上，“你为什么问这个问题呢？”

Betty为你拾起纸并把它放到最近的桌子上，回答道：“因为你根本不需要这些垃圾。你来这里要做一件事情，就是和我们这些用户谈话。从这些材料中得不到任何有益的信息。”

1. 只有告诉Betty你从每份文档中找到的东西才能使她相信每份文档都是重要的。用一段文字解释文档为需求工程师提供了什么帮助？

**二玉哥哥语**：看看书上 文档采样 那部分就知道了

**参考答案**：

不同的文档为需求工程师提供了不同的帮助，譬如：

* 需求规格说明书：帮助需求工程师发现需求信息，从而进行需求的重用
* 硬数据：通过研究和阅读也可以发现需求的相关信息
* 客户的续修文档：可以得到粗粒度的需求

通过分析这些文档，可以获取组织业务的问题域信息、业务工作流程的业务细节中存在的问题等。一个有经验的需求工程师会从现有的文档中获取事实，理解问题域。

1. 在你和Betty谈话的时候，意识到实际上也需要其他的定量文档。列出你缺少的东西。

**二玉哥哥语**：看看定量的和定性的就知道了

**参考答案**：

* 定量硬数据缺少：

**数据收集表格**：反应组织的信息域，收拾正在使用的每张表格，连同填写和分发说明一起，与填好的表格进行对比

* 定性硬数据缺少：
* 整个组织的**描述文档**：Sawder公司的组织结构图
* **业务指导文档**：Sawder公司的工作指南和规章手册
* **业务备忘**：

# 第十一章 需求分析概述

## 思考题：

1. 分析“结构化分析”和“面向对象分析”的过程，说明它们为什么都开始于系统的边界定义？

**二玉哥哥语**：和第二章相关：第二章云“软件要完成用户的任务需要和外界协调互动“开始于边界是因为边界是它们互动的地方。系统分析给自己做个定位的话，是先分析互动的反应，然后分析系统内部的反应，所以框架中有一些叫系统外部行为等，在需求早期阶段都是外部分析，到了后期阶段才会进入内部分析

**参考答案**：

软件要完成用户的任务需要和外界协调互动，经过问题分析之后一般可以得到高层次的解决方案及系统特性。而一个系统通常会有很多高层次问题，虽然问题分析之后可以得到解系统为了解决某一问题而需要具备的知识片段，却无法将这些片段自动连接为整个系统的概要全图，所以很有必要将各个问题的分析结果进行综合与处理，已确定整个解系统的功能，建立系统的边界。

之所以把系统边界作为需求分析的起点，是因为边界是软件和外界互动的地方。解系统为自己做定位，首先要分析互动的反应，然后分析系统内部的反应，所以，框架中有一些系统的外部行为等。

一般情况下，在需求分析的早期阶段做的都是外部分析，从系统的边界图开始，逐一分析和细化系统和外界的交互，以保证最终产品的行为能够和环境形成互动，以满足用户的需求；然后在需求分析的后期阶段，才会逐渐进入内部分析。

3. 列举结构化分析的各种技术，说明它们的数学基础是什么？

**二玉哥哥语**：有些有基础，而有些是没有基础的，结构化、数据流图、ERD、状态图（结构化的一种）都是有基础的，有些数据流图的细节方法是没有基础的

**参考答案**：

**结构化分析技术**：数据流图、实体联系图、状态转移图、功能实体矩阵、实体生命历史和事件实体矩阵。

以数据流动为中心，以DFD为核心技术，以λ演算为数学基础。

4列举面向对象分析的各种技术，说明它们是对结构化分析技术的继承和借鉴吗？如果是，那么说明它们借鉴了哪些结构化分析技术，如果不是，那么说明它们的数据基础是什么？

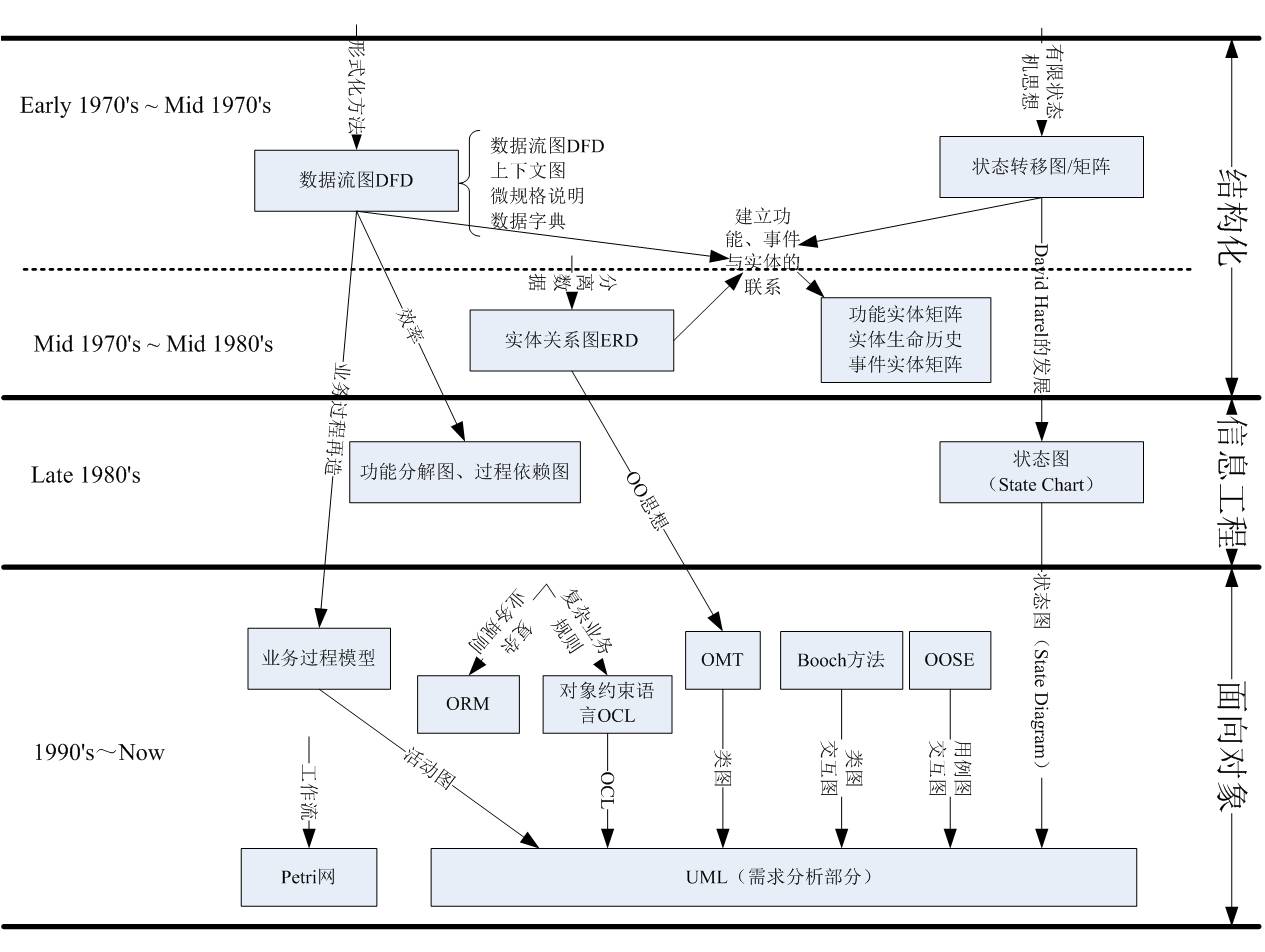
**二玉哥哥语**：很明显是，怎么是的看看发展历史的那张表

**参考答案**：

是

**面向对象技术：**用例图、类图、交互图、活动图、对象约束语言、状态图和工作流

**借鉴的结构化分析技术：**实体关系图、数据流图、状态转移图



5 Wieringa框架和Zachman框架给了你什么启示？

**二玉哥哥语**：看懂两个框架就明白了-\_-^^^^

**参考答案**：

* **Wieringa框架：**仅仅关注需求分析的建模技术，将需求分析技术分成7个类别：外部功能、外部通信、外部行为、概念组元、组元功能、组元通信和组元行为。
* **Zachman框架：**关注的是软件生产中的所有建模技术，提出了一个能够概括和分类系统开发中的信息结构。矩阵的每个单元都是从相应行的视图出发，对相应列的信息内容进行描述。行的内容包括：目标/范围、企业模型、系统模型、技术模型、组件模型和实际运行的系统；列的内容包括：数据、功能、位置、人、时间和动机
* **启示：**需求分析技术需要综合的运用，不同的分析方法在对同一个方面进行分析时会运用不同的技术。一方面，每一种需求分析技术都有自己的特点，有适用的情况。另一方面，复杂的应用需要要多视角的建模处理。为了更好的实现需求分析技术的综合运用，可以从以下的方面对需求分析技术的特点和联系进行深入的分析：
* 需求分析方法
* 技术的发展历程
* 技术的应用特征

这两个框架就是对技术的应用特征的诠释。

6“事件”和“事物”一直是进行需求分析的一个重要思路，你对此如何评价？

**二玉哥哥语**：有作用却也有局限性。事物和事件正好对应着结构化方法的两条路径，一个是DFD，一个是ERD，可参考书上结构化分析有两条路径这块内容，在结构化分析方法那章。但在面向对象里就不太好用了，因为面向对象的核心是多对象协同，多对象协同既不是事件，也不是事物，所以面向对象开始使用场景。事件就是行为，就是DFD；事物就是数据，就是ERD。

**参考答案**：

* **事件：** 可以描述、值得记录的在某一特定时间和地点发生的事情。

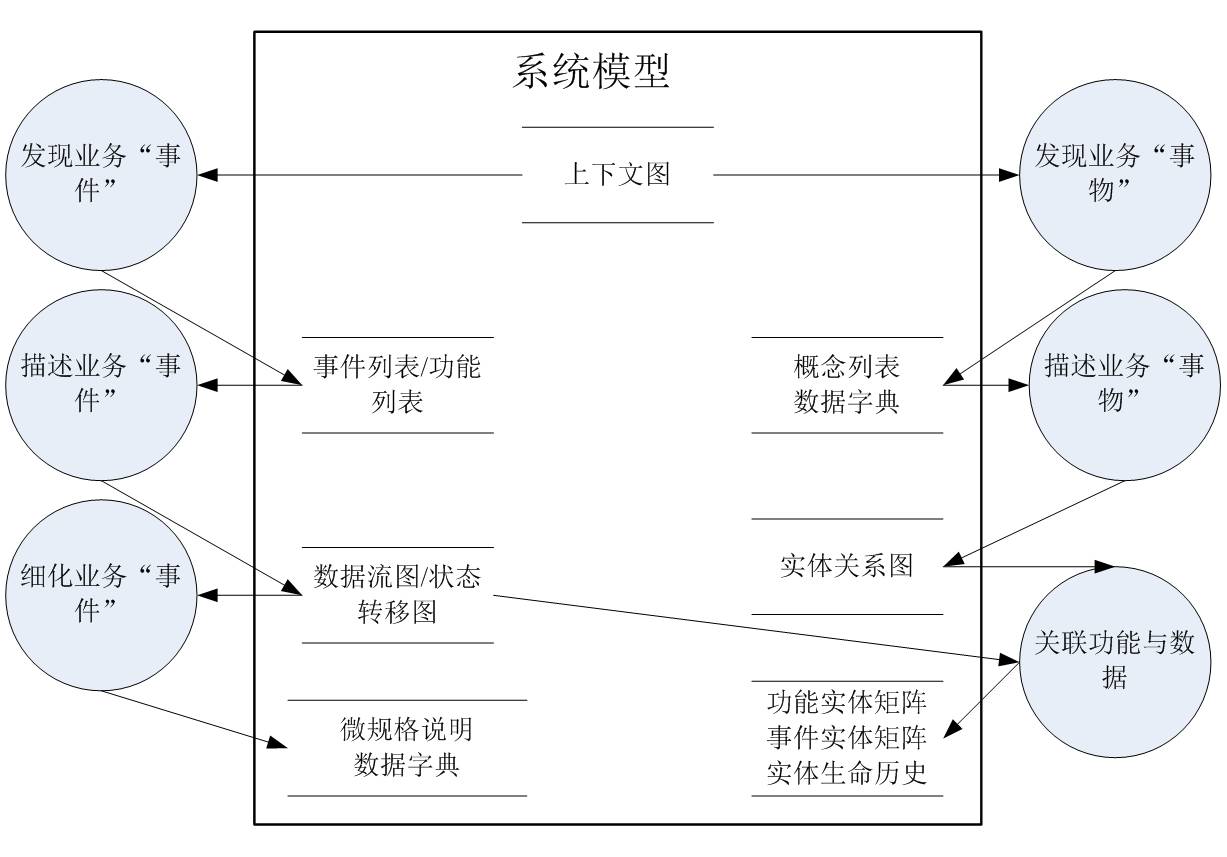
通过对事件的分析可以将复杂的系统需求分解成易处理并能更好理解的小单元。

事件可分为以下几类：外部事件、临时事件、状态事件。

* **事物**： 在面向对象的编程中，这些事物是在系统中相互交互的对象。

事物的类型：实物；角色；组织部门；设备；突发事件、事件或交互；地点/位置

* 结构化分析：事物和事件正好对应着结构化方法的两条路径，一个是DFD，一个是ERD



* 面向对象分析：核心是多对象的协同，而多对象既不是事件，也不是事物，而是基于场景的。所以，在面向对象分析中，事件就是行为，指的就是DFD；事物就是数据，指的就是ERD。

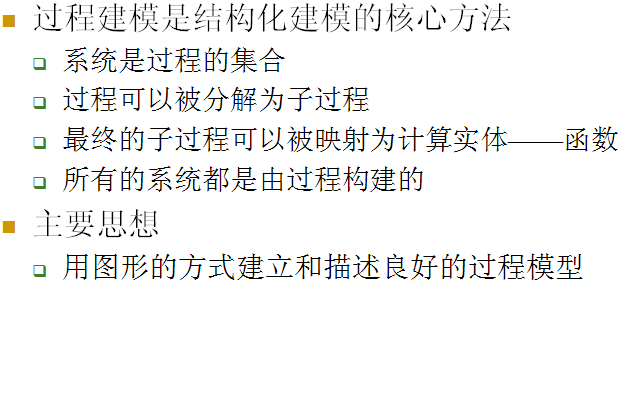
### 第12章 过程建模

**思考题**

1. **什么是系统思想？过程模型如何反映系统思想？**

系统思想（systems thinking）为一种整体分析方法，它的重点为系统组成部分相互关联的方式，提倡用整体、全局的、联系的观点看问题。系统思想使用计算机模拟和多种图表来建模、解释和预测系统行为。

过程模型没有把一个复杂的系统看成一个单个过程来对待，而是逐步功能分解，直到分解成易于理解和能计算机化的小功能，小模块



1. **第5章提出将系统中每一个问题解决方案的边界集成起来，就可以建立系统边界。你认为这种想法对上下文图的建立有什么启示？这种想法与基于DFD片段建立0层图的方法有何异同？**

1）要建立上下文图，首先要清楚系统的功能范围。系统的功能范围能够帮助界定系统的边界，进而发现系统应用的上下文环境。

2）共同点是都是集成组合而成，不同点：需要多次调整DFD片段的整合结果才能得出0层图；系统边界的建立是将各个问题的分析结果进行综合与处理，以确定整个解系统的功能，建立系统的边界。

1. 在需求获取阶段，需求工程师收集了大量的样本，包括文档、表格和报告，解释这些样本对过程建模有哪些用处。

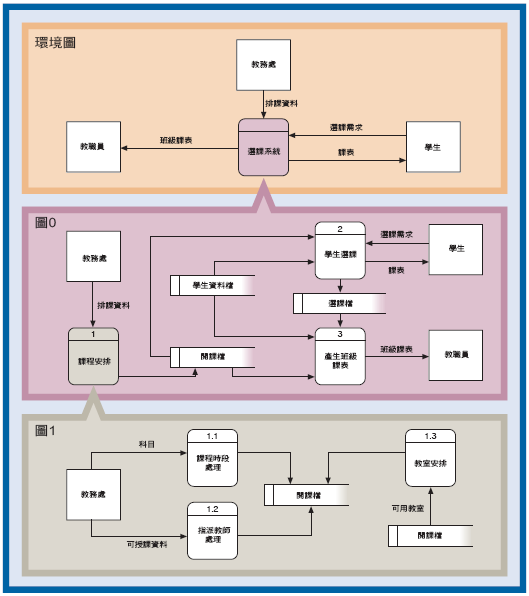
这些文档资料被称为硬数据，格式化的定量硬数据有助于理解数据实体，非格式化的定性硬数据反映业务流程和工作流程，这实际上帮助确定DFD的活动过程。

**案例题**

1. 分析你所在学校使用的选课系统，给出它的DFD描述。

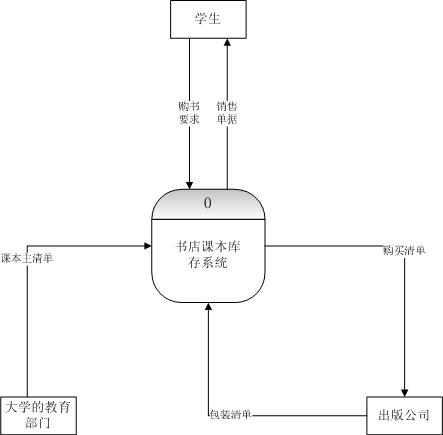
（网上教程中的一张图，我觉得挺对的，就不改了，大家自己改改

要说明的是，图中环境图指的是上下文图）



1. **根据下列叙述性描述，为描述的内容绘制一个上下文DFD。**

校园书店“课本库存系统”的目的是向学生提供本地大学课程的课本。大学的教学部门通过一个“课本主清单”向书店提交初始数据，包括课程、教师、课本和预计注册人数。书店生成一个“购买订单”，“购买订单”被送到供应课本的出版公司。图书订单随着一个“包装清单”到达书店，它被接收的部门检查和验证。学生填写包含课程信息的“购书要求”，当他们付了书款之后就得到一个“销售单据”。



1. **建立一个决策表，正确反映下面的课程评分策略**

一个学生可以得到一个期末课程成绩A、B、C、D、F。为了给出学生的期末课程成绩，老师首先确定一个学生的初始期末成绩，具体按照以下的方式确定：

头三次作业和测验中总成绩不低于90分，并且第4次作业成绩不低于70分的学生，这门课将得到成绩A。头三次作业和测验总成绩低于90但不低于80，并且第4次作业成绩不低于70的学生，这门课将得到成绩B。头三次作业和测验总成绩低于80但不低于70，并且第4次作业成绩不低于70的学生，这门课将得到成绩C。头三次作业和测验总成绩低于70但不低于60，并且第4次作业成绩不低于70的学生，这门课将得到成绩D。头三次作业和测验总成绩低于60，或者第4次作业成绩低于70的学生，这门课将得到成绩F。一旦老师确定了学生的初始成绩，他将决定最后的课程成绩。如果学期期间旷课不多于3堂课，这个学生的学生课程成绩将同他的初始成绩一样。否则，学生的学期课程成绩将比他的初始课程成绩低一级。

存在某些条件使得老师无法采取行动吗？如果有，你将如何改正错误？你的决策表可以通过消除不可能的规则或合并规则进行简化吗？

解答：1）存在使老师无法采取行动的条件，当初始期末成绩为F，并且旷课多于3堂时。

2）改正错误：添加条件，当初始期末成绩为F，旷课多于3堂时，不做降级处理

3）这个看自己的图说，

下面给出最终图

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 条件和行动 | 规则 | | | | | | | | | |
| 头三次作业和测试总成绩 | >=90 | >=90 | >=80并<90 | >=80并<90 | >=70并<80 | >=70并<80 | >=60并<70 | >=60并<70 | <60 | >=0并<=100 |
| 第四次作业成绩 | >=70 | >=70 | >=70 | >=70 | >=70 | >=70 | >=70 | >=70 | >=0并<=100 | <70 |
| 旷课堂数 | <=3 | >3 | <=3 | >3 | <=3 | >3 | <=3 | >3 | 任意 | 任意 |
| 成绩A | X |  |  |  |  |  |  |  |  |  |
| 成绩B |  | X | X |  |  |  |  |  |  |  |
| 成绩C |  |  |  | X | X |  |  |  |  |  |
| 成绩D |  |  |  |  |  | X | X |  |  |  |
| 成绩E |  |  |  |  |  |  |  | X |  |  |
| 成绩F |  |  |  |  |  |  |  |  | X | X |

1. **如果基本数据类型是单字符char，有效域为{‘a~z’, ‘0~9’, ‘A~Z’}，那么请以此为基础定义其他的数据类型：String，Integer（32位），Date（1900-01-01之后，包括1900-01-01）。**

**（此题不确定，自行斟酌）**

解答：String=1{[‘a’-‘z’| ‘0’-‘9’| ‘A’-‘Z’]}n (n正无穷)

Integer=[1|2|3|4|5|6|7|8|9]+31{‘0’-‘9’}31

Date=(闰年？？？)

### 第13章 数据建模

**思考题**

1. **在需求获取阶段，需求工程师收集了大量的样本，包括文档、表格和报告，解释这些样本对数据建模有哪些用处。**

解答：如果样本数据表述信息充分，那么可以通过从信息的描述中辨识和描述数据模型元素，

进而创建数据模型。在ERD的创建活动中，如果信息描述充分，那么，其创建活动如下：1）从描述信息中辨识实体2）确定实体的标识符3）从描述信息中辨识并最终建立实体间的关系4）进一步从描述中挖掘信息，为数据模型添加详细的描述信息

除了文本描述信息外，硬数据表单也是建立数据模型的理想资料。而且利用硬数据表单建立数据模型的情况在实践中非常常见。在ERD的创建活动中，面对复杂的硬数据表单，可以如下步骤创建：1）分析表单内容，确定表单主题，然后将每个主题描述为一个独立的数据实体2）建立主题间的关系，从而确定实体间的关系3）围绕主题组织表单的项目，将这些项目分派到各自的主题，并将其作为属性围绕主题组织起来。4）补充ERD的详细信息，挖掘表单中的信息对ERD进行补充

1. **比较过程模型和数据模型，每个模型显示了什么？应该在两种建模策略之间做出选择吗？为什么？**

解答：1）过程建模时分析需求获取活动中获取的信息，发现系统的功能和其与外界的交互，建立能够实现系统功能的过程分解结构，形成系统的过程模型，并用图形的方式将过程模型描述出来。过程建模以数据在系统中产生和使用为重点，以进行数据转换的过程为核心，建立层次结构的过程模型来描述系统，它同时描述了系统的行为和数据。不过在数据说明方面，过程模型更多的是侧重数据产生于使用的时间、地点和方式，而没有描述数据的定义、结构和关系等特性。数据建模技术能够弥补过程模型在数据说明方面的缺陷，它描述数据的定义、结构和关系等特性。数据建模建立的模型称为数据模型，是问题域和解系统共享的知识集合，通常能够反映企业业务的核心知识。数据模型说明了问题域和解系统共享的事物、对共享事物的描述和共享事物之间的关系。 （概括提炼一下）

2）有选择。

（二玉说：）如果过程模型没有数据存储这两种就没有冲突，但是如果有数据存储，就要考虑数据存储和数据模型的选择

数据模型是在过程模型上发展出来的新模型，要点以数据为主，通常涉及数据时应该使用数据模型

1. **有些需求工程师认为数据建模是业务需求建模中最重要的方面，你如何评价这种看法？**

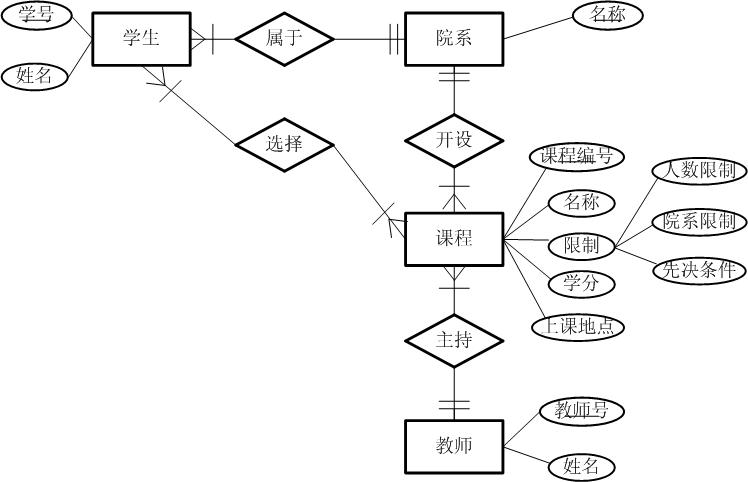
解答：从方法学的角度来看待这个问题

在结构化分析阶段：1）处于传统结构化分析 (late 1960’s)和现代结构化分析 (late 1970’s)时期时，是以数据流动为中心，以DFD为核心技术，辅助ERD,STD…，这个时候过程建模的地位要高于数据建模2）历经发展，在信息工程 (late 1980’s) 时期，以数据知识结构为基础，ERD为核心技术，辅助DFD，STD, FDD, PD…，这时数据建模被认为最重要。

在面向对象分析阶段：以对象为中心，以UML（类图）为核心技术；以全面思想革新为理想，以承继结构化技术为现实。这个时候，数据建模仍然重要，因为，现阶段毕竟还是数据库为主。但是，考虑到其局限性，数据为中心在面向对象中的所起的作用不如信息系统工程中所起作用明显。

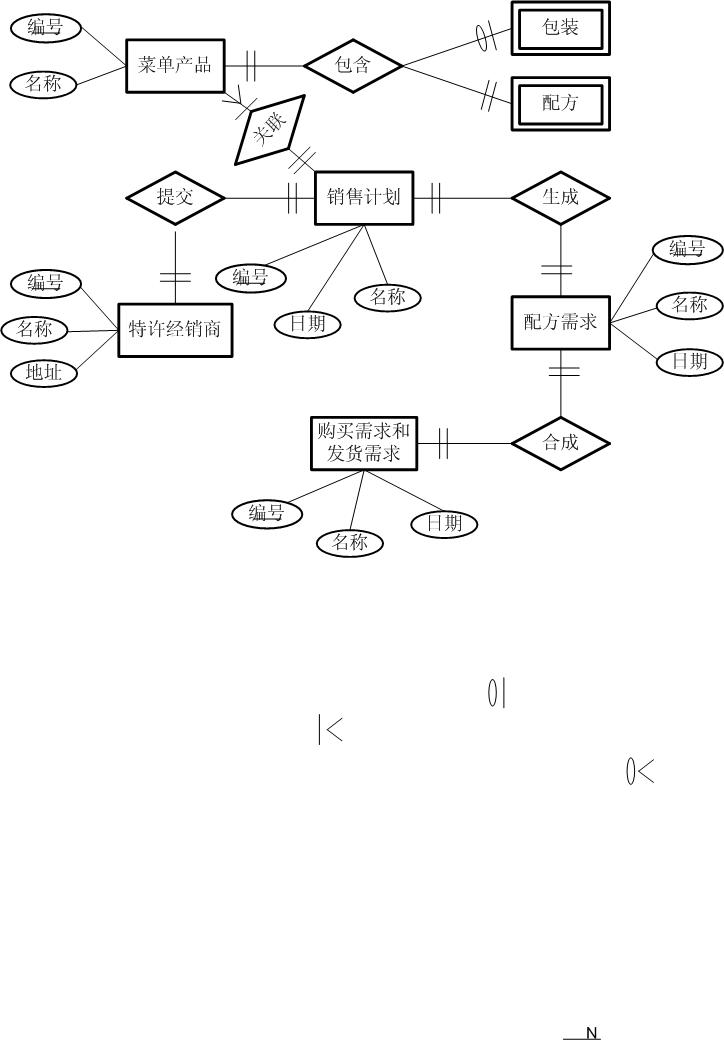
**案例题**

1. **分析你所在学校使用的选课系统，给出它的数据模型描述。**

解答：

1. **为下列描述建立ERD**

Burger World分销中心为45家Burger World特许经销商提供供应服务。你参与了为分销中心构造一个数据库系统的项目。每个特许经销商对下一个月其Burger World的菜单产品提交一份当天的销售计划。所有的菜单产品需要有配方和（/或）包装。基于商店销售计划，系统必须每天生成一个当天的配方需求，然后，将那些需求合成为每周一次的购买需求和发货需求。

解答：

1. 我们企业的MIS 部门想构造一个数据库来跟踪所有的硬件和软件。我们拥有工作站、网络服务器和外设，而且MIS 部门想跟踪软件包以及这些软件包的许可证。有些软件许可证是针对单机的，我们可以把这个软件安装在网络服务器上，但只能允许与许可证授权的用户数同样多的网络用户使用该软件。我们还拥有网络许可证，单个网络许可证授权了一定数量的用户。非网络许可证可以安装在工作站或服务器上。我们想跟踪软件许可证安装在哪里。某些许可证可以在某个时间未被安装在任何地方。我们还必须能够证明安装软件的合法性。每个许可证必须被跟踪到一个购买订单、赠品或者一次租借。我们也可以订购一些软件。我们订购软件包，同时收到许可证。请通过集体讨论构造数据模型和属性。

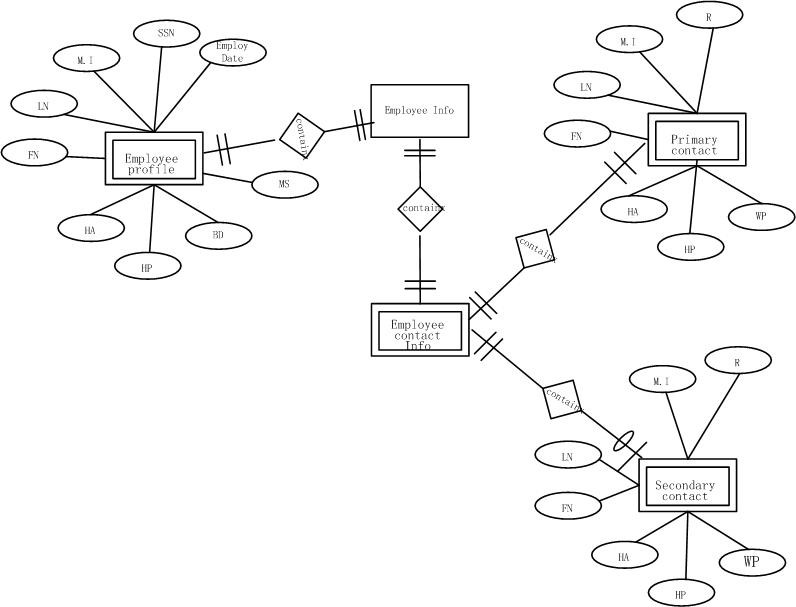
解答：

1. Sunset Valley Distributors公司最近完成了一个大的转换项目。几个月前，公司决定进入数据库时代。公司的计算机文件有很多以及不可靠了，难以维护，并且对于实现许多最终用户的报告和查询请求来说太不灵活了。DBMS看来是一个显然的解决方法。两个需求工程师主要负责这个转换项目，这花了他们几个月的时间。需求工程师已经决定简单地将每个计算机文件实现成关系数据库中的一个独立的表。一旦转换完成，文件系统中存在的问题又会出现在数据库系统中。报告包括了不正确的数据，报告和查询请求不容易实现，数据维护仍然很困难。公司雇佣了一个顾问来研究这个问题。顾问认为许多问题是因为分析员没有成功进行数据建模造成的。请解释设计数据库时进行数据建模的重要性。

解答：（二玉没讲过，书上PPT上都没有~~）

1. **结合下图的表格内容，给出相应的ERD描述**



解答：

**第14章 面向对象建模**

**思考题**

**在需求获取阶段，需求工程师收集了大量的样本，包括文档、表格和报告，解释这些样本对面向对象建模有哪些用处。**

帮助人们识别类、对象、活动顺序时序与事物状态

评价下面一句话：面向对象是个好东西，但是只有专家开发者才能用好它。

不看

**比较包括CRC策略在内的各种对象与类的发现方法，说明各种的优缺点和适用场景。**

答：

1在拥有充分需求信息描述的情况下，发现对现象和类的主要方法有三种：使用概念类分列表、名词分析和行为分析。使用概念类分类列表和名词分析又被称为经典方法，因为他们主要来源于经典分类理论，产生的概念类具有数据驱动的特征。这两种方法易于使用，而且他们产生的数据驱动的领域模型也特别适合于对信息系统的问题域进行描述。相比之下，行为分析就是以概念聚类为基础的职责驱动的方法，它适合描述复杂希特同系统和控制系统。

概念类分类列表：事先给出一个概念类的分类列表，然后又分析人员在需求信息中寻找相应类别的候选对象，最后对候选对象进行确定和归纳，形成概念类。

名词分析师一种运用语言分析的实用方法，名词分析从文本描述中识别出有关的名词和名词短语，然后将它们作为候选的对象，最后对候选对象进行确定和归纳，形成概念类。

行为分析，和名词分析不同的是，行为分析师从需求描述中搜寻动词，识别出系统行为，然后找出系统的主动对象和被动对象作为候选对象。找出候选对象后，就按照对象的含义进行对象的确定，针对确定后的对象，以其发起行为的组合作为特征描述，并以特征的相似性进行归纳分类，产生概念类。

前面的面向对象建模方法都是适用于简单情况下的建模方法，在需求信息比较明确时，他们能够发挥很好的作用。但是在复杂情况下，需求的获取和分析是交织前进的，也就是说在进行分析和建模时没有非常明确和固定的需求信息可以利用，此时前面描述的建模方法很难取得好的效果。

基于CRC卡的职责驱动方法是一种用来处理复杂情况的面向对象建模方法，比前面描述的面向对象建模方法复杂很多，需要更多的实践经验和技巧。CRC是Candidates Responsibilities和Callaborators的缩写，基于CRC可以建立一种索引卡片，CRC卡。每个卡片代表了一个被发现的候选对象。卡的背面是关于候选对象的非正式描述，卡的正面记录了对象的职责（所维护状态和可以执行状态的行为）和协作者。CRC卡简洁方便可以随时被移动、修改或者丢弃，所以它特别适合在复杂的系统中充当进行对象的发现和设计思想的挖掘，即进行复杂情况下的面向对象的分析和设计。

概念类分列表 简单方式

名次分析：源于ERD 倾向于数据驱动 数据库为主，作用

行为分析：行为复杂时 倾向于行为驱动 控制行为复杂

CRC：面对复杂情况 数据行为综合考虑，需要限定和规则

**一直以来，开发者认为面向对象方法在两个方面有着自己的优势：（1）对象的思想符合人们认识现实世界的思路；（2）顺利的实现了从分析向设计的平滑过渡。请你对此进行评价。**

1面向对象分析——现实世界的复杂模型：

复杂总是简单部分的组合

简单部分又是更简单部分的组合

简单组成复杂的过程存在层次性

每个最小简单部分独立负责完成一系列相关任务

相比较而言，每个组合内部各部分的关系比其内部与外部的关系都更紧密

各个部分通过一致的接口进行组合，即一个部分对其它部分的所知仅仅是接口

2面向对象分析——映射现实模型的面向对象思想

任何系统都是能够完成一系列相关目标和任务的对象

对象完成一个任务时会请求一系列其他对象帮助其完成一些子目标

其他对象为了完成其任务又会请求将子目标更细分为子子目标，并请求其他对象帮助完成

子目标的分解和责任分担一直进行直到最后产生的子部分可以映射到计算实体

计算实体：对象

层次关系：聚合（组合）、继承、关联

组合接口：一个对象暴露的接口

面向对象建模：一种用于辨识系统环境中的对象及这些对象之间关系的技术。面向对象建模是面向对象方法学在需求分析中的应用，也称面向对象分析。它采用面向对象方法学的世界观，将系统看做是一系列对象的集合。每个对象有独立的职责，完成独立的任务，对象之间通过消息机制相互协作，共同实现系统的目标。

对象的概念是对象模型的基础，从理解现实的角度来看，它是对现实世界事物的抽象，就是说对象代表了现实世界的事物。在软件设计中可能会出现一些在现实世界中没有对应事务的对象，不过在需求分析中，对象都应该代表相应的现实世界事务，这也是面向对象分析和面向对象设计的一个不同点，

2面向对象的方法首先从需求的源头（主要是用户）进行需求的获取，组织需求信息的用户描述，建立用例模型。

在得到用例模型后，面向对象的方法一方面从用例模型中寻找对象和类，建立领域模型，领域模型描述了业务工作中的概念类和类的重要属性;另一方面，面向对象方法依据用例模型建立行为模型，行为模型是用力模型的实现，体现了用例描述中的系统行为。这个阶段是需求的分析阶段，它的主要目的是理解用户的需求，所以它建立的对象模型和行为模型都是比较粗略的模型，还没有涉及软件实现相关的技术细节。

在得到用户需求的完整、准确理解后，面向对象方法就开始考虑软件的实现机制，进行软件设计。设计阶段以软件的高质量实现为第一目的，所以设计阶段需要在领域模型和粗略行为模型中加入软件实现的细节信息，进行模型的细化，建立细化的对象模型和行为模型。最后结合细化后的对象模型和行为模型，为类进行系统行为的分析，完成最终的完全对象模型。

在得到完全的对象模型后，编程人员就可以选择一种面向对象的语言，完成程序的编写，使得软件变成实际的存在形式：程序代码。

在整个面向对象方法的过程中，分析和设计的分界线比较模糊，因为：

分析和设计都是用相同的模型，只是模型内包含的信息有所不同而已，分析则侧重于需求的理解，设计侧重于实现的细节

关于需求理解信息（WHat）和实现细节信息（How）的区分比较模糊；

模型粗略和细化仅仅是程度上的不同，无法定义一个非常准确的区分标准，所以无法准确界定应该何时终止对粗略模型的加工以及设计应该何时开始对模型进行细化。

**第5章提出将系统中每一个问题解决方案的边界集成起来，就可以建立系统边界。你认为这种想法对系统用例图的建立有什么启示？**

系统边界是指系统所包含的系统成分与系统外事物的分界线，用例模型使用一个框来标示系统边界，以显示系统的上下文环境。一个系统边界建立，所有的用例都是子系统的内部功能，介于系统边界之内，所有的参与者都是系统外的交互对象，介于系统边界之外。用例在需求的获取和组织上都有着很好的特性，因此面向对象对象方法主张使用女管理作为需求获取和组织的主要手段，用例模型就是以用例为基本单位建立的一个系统功能展示模型，他是系统所有用例的集合，以统一、图形化方式展示系统的功能和行为特性。

基于以下描述开发一个领域模型。

这个例子是一个简化了的大学图书馆系统。当然，图书馆系统必须跟踪书的情况，同时还要维护关于书的标题及副本的信息。书的标题维护信息是关于名称、作者、出版商和目录号等信息。每个副本维护副本号、版本、印刷日期、ISBN、本书状态和归还日期等信息。

同时图书馆系统也要跟踪图书馆借书人的情况。由于它是一个大学图书馆，所以有几种类型的借书人，他们有各自不同的特权。这里包括教职工借书人、研究生借书人和本科生借书人等。借书人的基本信息包括姓名、地址和电话号码等。对于教职工借书人，还要包括诸如办公室地址和电话等信息。对于研究生借书人，还要包括研究项目和导师信息等。对于本科生借书人，还要包括项目和所有学分信息等。

图书馆系统也要跟踪借出书本信息。当一个借书人捧着一堆书去借书台办理借书手续时，借出这个事件就发生了。随着时间的过去，一个借书人可以多次从图书馆中借书。一次可以借出多本图书。

如果借书人想要的书已被借出，他可以预约。每个预约只针对一个借书人和一个标题。预约日期、优先权和完成日期等信息需要维护。当借书完成，系统会将这本书与借出联系起来。

1发现对象和类

候选对象 ： 人：借书人 教职工借书人、研究生借书人和本科生借书人 图书馆管理员

事物： 书 副本

组织： 图书馆 借书手续 预约手续

概念： 预约日期 标题 借书终端

事件： 借书 预约

摒弃对象： 标题：只有状态没有行为

预约日期： 无状态无行为 （黑体为剩下的）

2建立类之间的关联

需要协作的概念类：

借书人 借书终端

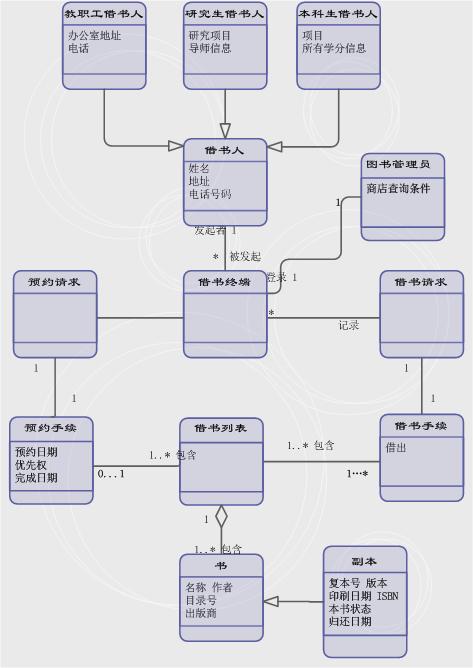
图书管理员 借书终端

借书手续 书

预约手续 书

问题域内的联系：

3 添加类的重要属性



**案例题**

**结合题1和下面的描述，给出大学图书馆系统的用例模型。**

借书人根据图书馆的信息来检索书名，同时检索这本书是否可以被借出。如果一本书的所有副本都被借出了，那么借书人可以根据书名预订这本书。当借书人把书拿到借书台的时候，管理员可以为这些书办理归还手续。管理员要跟踪新书到达的情况。

图书馆的管理者有属于自己的活动。他们要分类打出关于书的标题的表格，还要在线检查所有过期未还的书，也标出来。而且，图书馆系统还可以从另外一个大学的数据库中访问和下载借书人的信息

先找用户再找目标。



**下面是一段用例的描述，针对一个汽车保险系统中“将一辆新车加入一个已有保单中”的用例。请你为其设计系统顺序图和活动图。**

**客户打电话给保险公司，并提供他的保单号，办事员输入这个信息，系统显示基本的保单。然后办事员检查信息，以确保保险费通用及保单有效。**

**客户给出要添加的汽车的牌子、模型、年份和车辆识别代号（VIN），办事员输入这些信息系统验证这些数据是否有效。然后客户选择期望的保额类型，以及每种类型的数量，办事员输入这些信息，系统会逐一记录并根据保单限制验证所请求的数量。输入所有的保额后，系统验证保额总和，包括保单上的其他汽车。**

**最后，客户必须要确认所有的驾驶员，以及他们驾驶汽车的时间比例。如果有一个新驾驶员加入，则调用另一个用例“增加新驾驶员”。**

**整个过程最后，系统更新保单，计算新的保险费，打印新的保单说明，邮寄给保单所有人**

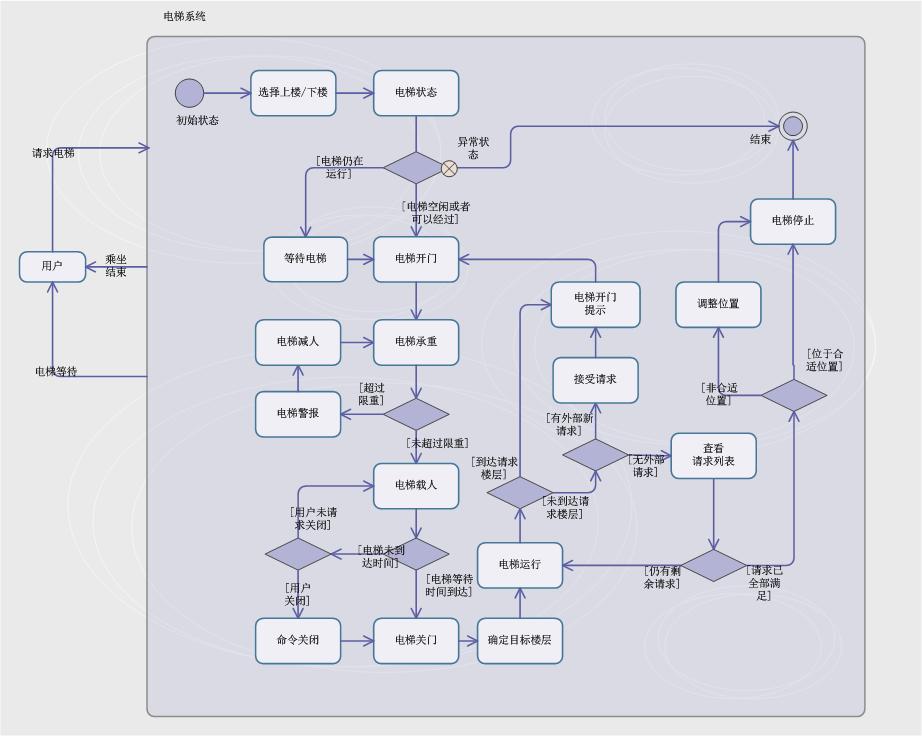
几个活动 数据流 参与人角色 甬道

顺序图 活动图



**请你给电梯调度系统开发一个状态图描述。**

建立状态转移矩阵



**第15章 需求规格说明**

**思考题:**

**什么时候建立术语表？**

项目初期有，需求整个阶段不停维护，

术语表是对重要术语清晰、一致的说明，用于准确描述术语的含义。通过术语表，文档的写作者和读者可以就术语所表示的意义达成共同理解。定义术语表时可以避免下列常见问题：

术语不一致：在人们计划和编写文档时，通常会使用一些术语，然后无数次地修改这些术语，特别是当他们对属于有新的想法，以及对问题域的理解有更深认识的时候，这种做法带来的负面影响是，在同一个文档中累计了针对同一个事物的不同的、旧的和新的术语定义，这会给读者的正确理解带来困难。如果能够建立并维护一个术语表，那么人们可以很容易的再文档中进行术语的查询和替换，这样就可以避免文档中出现的术语不一致。

方言问题：建立术语表时可以逐一解释术语的“方言”形式，这样就能妥善解决方言问题。

建立属于问题域的术语表可以避免术语问题，在术语表的基础上展开文档的内容描述可以避免冗余术语问题。

**在需求获取和需求分析当中采用哪些手段可以保证最终需求集的完备性、一致性和正确性？**

完备性，

需求规格说明文档时完备的，当且仅当1描述了用户所有有意义的需求，包括功能、性能、约束、质量属性和对外接口；2定义了软件对所有情况的所有实际输入（无论有效输入还是无效输入）的响应；3为文档中的所有插图、图、表和术语、度量单位的定义提供了完整的引用和标记。需求的完备性要求不能遗漏任何需求或者必要的信息，为避免需求遗漏，需求工程师要做好业务需求的分析，建立并控制正确的项目规范，建立业务需求、用户需求和系统需求的跟踪关系也用于发现需求的遗漏现象。文档内所有TBD（待解决问题）被全部解决之前，需求规格说明文档都是不完备的。

一致性，1细节的需求不能同高层次的需求相冲突，2同一层次的不同需求之间也不能互相冲突。也就是说，软件需求规格说明文档既要在所含内容上保持一致，也要和更高层次的文档（例如系统需求规格说明书）所含的内容保持一致。为保证需求规格说明文档的一致性，由开发人员和非开发人员对于其进行手工评审时非常必要的。

正确性，保证文档中每个单一需求都是优秀的需求。单一需求的优秀特性可以使整份文档满足正确性，无歧义和可验证。正确性是文档内的所有需求都有正确性，无歧义指的问胆囊管选中的指令，可验证文档内所有的所有需求都是可验证的。

**关于文档化的三种手段——非形式化、半形式化和形式化，一致以来存在着较多的争论，对此你的看法是怎样的？**

形式化不利于理解，但很符合数学推理，数学语言

非形式化易于理解，但存在歧义与模糊性，自然语言

所以半形式化比较好

非形式化语言，即自然语言：自然语言具有复杂的规则和多样化的表达方式，所以它的表达能力最为强大。而且自然语言术语普通人的语言，每个人都熟知其规则、表达方式和特点，所以非常利于用户的理解。但是同时自然语言也具有松散、模糊、歧义、凌乱等不好的特性，这使得它无法被机器所理解，它所描述的信息内容也无法准确的映射为机器行为。

形式化语言：基于教学方法的语言，具有数学的表示法特性。使用形式化语言描述的信息内容是可以进行逻辑一致性推到和证明的，所以它能够保证信息的正确性。但是形式化描述的信息描述能够被机器所理解，它所描述的信息内容可以准确的被映射为机器行为。但是形式化描述的信息要求读者具有谓词演算方面的知识，这对普通用户而言显然要求过高，以至于大多数客户无法读懂这个形式化方法描述的信息。形式化方法所能描述的内容也很有限的，具体的有限性因行使化方法的不同而各异。

半形式化语言室介于自然语言和形式化语言之间的描述语言。一方面，半形式化语言具有严格的语法，定义方式比自然语言更加严格，这使得它可以避免自然语言模糊、松散、歧义、凌乱等不好的特性。另一方面，半形式化语言具有丰富的予以，使用规则比形式化语言更复杂和多样，使得它具有比形式化方法更强的表达能力。但是，丰富的语义使得半形式化语言的语法无法严格到可以等价于数学方法的程度，所以它描述的信息还需要进行额外的处理才能够被机器所理解或者准确的映射到机器行为。同时，严格的语法限制也使得半形式化语言的表达能力无法达到自然语言程度。而是因为具有独特的语法和语义，所以半形式化语言对普通用户而言无异于一门全新的语言，它所描述的信息很难被用户所理解。

为了实现复杂的规则、多样的表达方式和强大的表达能力，自然语言采用了以文本为主的描述方式。形式化语言也是使用以文本为主的描述方式，但是它所使用的文本都是经过严格选择和限定的，代表着特定的数学符号。和他们不同的是，半形式化语言采用了以图形为主的描述方式。

因为：

1 半形式化语言的语法限制使得它用于信息描述的基本元素师有限的，这个有限性使得它以限定文本或限定图形符号为描述方式成为可能；

2 半形式化语言追求表达语义的丰富性，而在这一点上图形符号式胜过限定文本的，所以人们倾向于选择使用图形符号的表达方式。

因为三类语言的特性区别，所以在进行需求规格说明文档的编写时，用户倾向与使用自然语言，因为其他两种类别的语言难以理解。而开发人员倾向于使用半形式化语言和形式化语言，因为自然语言的表达不够严格和准确。形式化语言在实践中应用很少，因为需求规格说明书对语言的予以核表达能力有着较高的要求，而这恰恰是形式化语言有所欠缺的。（可靠性要求高的，如医学）

为了让需求规格说明文档的内容能同时满足用户和开发人员的需要，需求工程师在实践中更多时候会综合使用自然语言、半形式化语言和形式化语言。如，为半形式化语言和形式化语言添加自然语言的注释，或者分别使用自然语言和半形式化语言（或形式化语言）重复描述同样的信息，或者使用半形式语言和形式化语言描述概要与抽象信息，然后再用自然语言进行详细信息的描述

非形式化的自然语言一种语法规则非常复杂的模型语言，他具有很强但并不复杂的语法规则，能准确描述一些有限的特殊情景。分析模型采用的半形式化语言则是介于自然语言和形式化语言之间的语言形式。

为同时满足开发者的严格逻辑性要求和读者的一度性要求，实践中更多的是同时使用半形式语言和自然语言，及时同样的内容进行两次不同方式的描述。文档中的图形符号要进行额外的解释，自然语言是首要选择。

**第16章 需求验证**

**1 用于需求获取的原型与用于需求验证的原型有何异同？**

原型是在软件开发中被广泛使用的一种工具，在软件开发过程中的各个阶段包括需求开发，都会使用不同类型的原型达到不同的目的

当有些需求涉及复杂的动态行为时，可能就需要使用模型来加以验证。面对不确定的需求的时候，需求获取的时候就会使用原型的方法。相同点就是都使用了原型的方法。 区别：原型验证是开始于确定完整的需求，都是比较明确的需求，对于修改的要求不高，但是对性能捕获上要求比较高，由于开始于清晰地需求，所以原型验证更倾向于实验式。而需求获取的原型，一般都是开始于不确定的需求，修改性比较大。并且一般通过原型进行获取的需求，一般不用原型检验

**2多种需求验证的方法应该如何结合运用？**

需求验证的方法：需求评审（静态分析，需求验证的一种主要方法）， 原型与模拟，开发测试用例，用户手册编制，利用跟踪关系，自动化分析

每个需求都需要经过评审，对于动态行为评审不能完成的就要通过原型和模拟的方法来验证。在正常的工作当中，可以顺便用上用户手册，测试用例，跟踪等方法在一些错误之处或者一些需求上进行验证，也是比较有效的。总而言之，大多数情况下，需求都是在静态的方式下被加以验证的（评审的方法），也可以说几乎说的需求都要经过评审的方法进行验证，个别动态复杂的需求需要用原型与模拟的方法进行验证，工作之间产生的衔接可以用上开发测试用例，用户手册等方法，这样可以实现高效的综合运用。

### 17章需求管理

**思考题**

**如何有效的处理需求的变化时很多现代化软件的开发技术的主题，对此现象你有什么看法？你将怎样做可以控制一个需求多变的项目？**

（1）（需求的变化正当不可避免的原因，还涉及需求的变更控制）

需求开发是一个获取、明确定义需求的过程。但需求并不是在需求开发结束之后就会恒定不变的。在产品开发和实现中或者产品递交之后，用户也常常提出需求的变化，这会给系统的开发工作带来额外的烦恼，增加工作量。尤其在软件应用日益复杂的情况下，需求变更带来的影响越发明显。

需求的变化是正当和不可避免的，情况有：

1问题发生了改变

2环境发生了改变

3需求基线存在缺陷

（实践中下列因素也会导致需求变化）

4用户变动

5用户对软件的认识变化

6相关产品的出现

（2）（怎么样做变更控制：变更控制的人、过程、处理；尤其是我们讲的注意事项）

需求的变化时正当不可避免的，在需求开发之后冻结需求是不恰当的做法。但是需求的变更又可能会给项目带来很大的负面影响，随意的需求变更也是不恰当的做法。正确的做法是在形成需求变更基线之后，进行需求的变更控制。

需求的变更控制就是以可控、一致的方式进行需求基线中需求的变更处理，包括对变化的评估、协调、批准或拒绝、实现和验证 。

变更控制过程：提请求需求变更；接受变更请求；变更评估；变更决策（拒绝or决定）；执行变更；验证变更（具体见P390）

在需求变更控制中有以下注意事项：（自定回答的精度）

1认识到变更的必要性，并为之制定计划

* + 定义明确的变更控制过程，建立变更控制的有效渠道
  + 所有提交的需求变更请求都要进行仔细的评估
  + 是否进行变更的决定应该由变更控制委员会统一做出
  + 必须对变更的实现结果进行验证
  + 需求的变化情况要及时的通知到所有会受到影响的项目涉众

2维护需求基线，审计变更记录

3管理范围蔓延

* + 根据业务目标、产品前景和项目范围，评估每一项提议的新增需求和特性

4灵活应对变更请求

* + 推迟产品的交付时间。
  + 要求增派人手：在有限的情况下有效
  + 要求员工加班工作：只能适度的使用。
  + 推迟或者去除尚未实现的优先级较低的需求
  + 容许产品质量的降低：尽量不使用

5使用辅助工具

* + 工具应该具有以下几个特性，以支持需求变更过程：
    - 可用定义变更请求中的数据项。
    - 可用辅助项目涉众完成变更控制过程中的协作。
    - 可以帮助维护需求基线，审计变更记录。
    - 能够将变更情况及时的通知到相关人员。
    - 可以生成标准的和定制的报告和图表。