什么是软件危机?

软件危机是指计算机软件的开发和维护过程中遇到的一系列严重问题。这些问题或难题并不限于那些“不能正常运行”的软件，

般来说，软件危机包括两方面的内容:如何开发软件，满足对软件日益增长的需求如何维护数量不断膨胀的已有软件

1.软件危机的表现形式

1)对软件开发成本和进度的估计常常很不准确:

2)用户对“已完成”的软件系统不满意的现象经常发生;

软件产品的质量往往是靠不住的;

软件常常是不可维护的;

软件通常没有适当的文档资料;

软件成本在计算机系统总成本中所占的比例逐年上升;

EX.12306网站

软件开发生产率提高的速度，远远跟不上计算机应用迅速普及及深入的趋势。

3.产生软件危机的原因

1)由软件本身的特点决定

逻辑产品;

不会因为使用的时间过长而被“用坏”。由软件人员本身的素质决定2)

软件的专业人员对软件的开发和维护有糊涂的观念，在实践中采用了错误的方法和技术:

软件开发人员忽略了对目标系统每个细节的完整、准确、具体的认识，急于求成:

对软供生存周期这一过程认识不全面，淮确地说开发一个软件需要经过问题定义、可行性研究、需求分析、总体设计、详细设计、编码、测试、维护八个阶段;，不重视早期修改，在软件开发的不同阶段进行修改需要付出的代价不同;

忽略了软件维护工作的重要性，对软件工程师来说，软件维护工作本身不是一项吸引人的工作，理解他人的程序也常常是困难的:

1. 由硬件设计中的长期一贯制所决定的计算机从1946年诞生至今，半个世纪以来，一直遵循冯·诺依曼的设计原则不变，即采用二进制和程序存储的原理。

4.解决软件危机的途径

1)加快新一代计算机的研制工作;

2)应该有组织，有计划，通过严格的管理手段进行软件的开发;

及时总结软件开发的成功技术和方法并加以推广:

应该选择好的开发工具进行软件的开发

软件工程是一类求解软件的工程，它应用计算机科学、数学(用于构造模型和算法)和管理科学(用于计划、资源质量和成本等的管理)等原理，借鉴传统工程(用于制定规范、设计范型、评估成本、权衡结果)的原则和方法，创建软件以达到提高质量、降低成本的目的。

软件工程是一门指导计算机软件开发和维护的工程学科。随着人们对软件系统的研制开发和生产的理解而改变，应用发展的眼光看待它。

现代软件工程的本质特性

软件工程关注于大型程序的构造；软件工程的中心课题是控制复杂性；软件经常变化

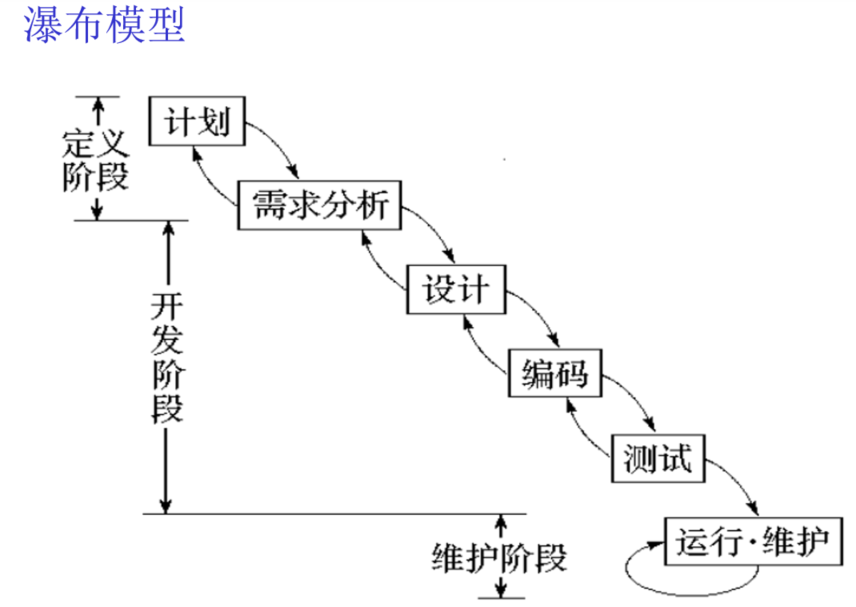
开发软件的效率非常重要；和谐地合作是开发软件的关键；软件必须有效地支持它的用户；在软件工程领域中是由一种文化背景的人替另一种文化背景的人创造产品；

现代软件工程的基本原理

用分阶段的生存周期计划严格管理；坚持进行阶段评审；严格实行产品控制；采用现代程序设计技术；结果应能清楚地审查；开发小组的人员应该少而精；承认不断改进软件工程实践的必要性；

1.3软件生存期模型(开发模型)

软件生存期模型是跨越整个生存期的系统开发、运作和维护所实施的全部过程、活动和任务的结构框架。



瀑布模型( Waterfall Model)

又叫流水式过程模型，它是模仿旅游景点的阶梯瀑布，由上向下一个阶梯一个阶梯地倾泻下来，最后进入一个风平浪尽的大湖，这个大湖就是软件企业的产品库。

1. 模型的本意:根据软件生存周期是由问题定义、需求分析、策划、总体设计、详细设计、编程、测试、运行、维护等阶段所组成的，就把每个阶段当作瀑布中的一个台阶，把软件生存过程比喻成瀑布中的流水，软件生存过程在这些台阶中由上向下地奔流。开发人员按照阶段开发，管理人员按照阶段管理。
2. 模型的特点:

里程碑或基线驱动，或者说文档驱动;过程逆转性很差，或者说不可逆转。

1. 选择模型的条件:

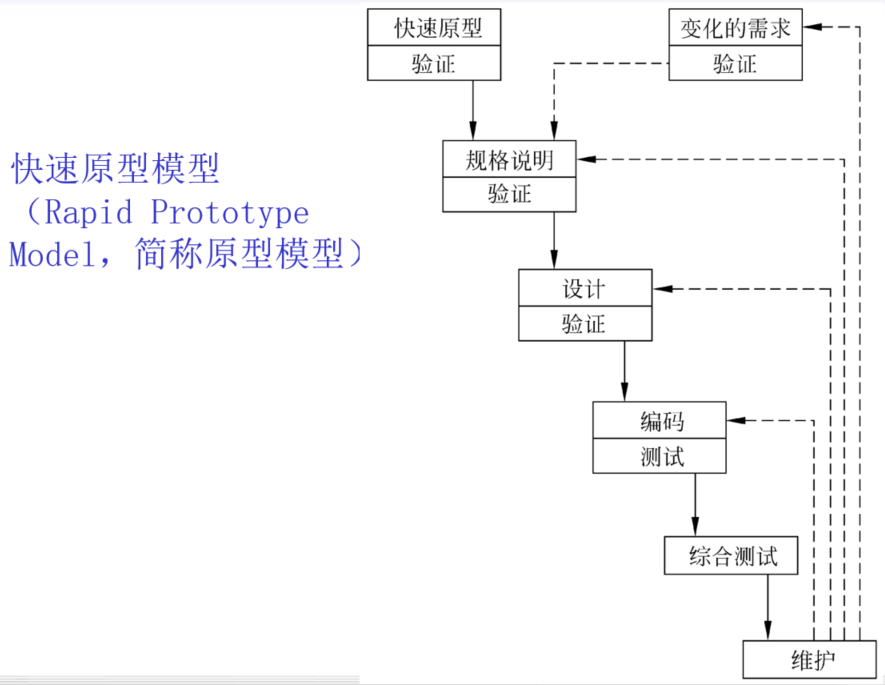
(1)在开发时间内需求没有或很少变化:(2)分析设计人员对应用领域很熟悉。

(3)低风险项目(对目标、环境很熟悉) (4)用户使用环境很稳定。

(5)用户除提出需求以外，很少参与开发。

1. 模型的缺点:可维护性差，表现在

由于逆转性很差，所以返工会造成重大损失。错误的传递，会采取发散扩大的方式。



1.模型本意:

在初步需求分析之后，马上向客户展示一个软件产品原型，对客户进行培训，让客户试用，在试用中收集意见，修改原型，再让客户试用，反复循环几次，直到客户确认为止。

2.为什么要采用快速原型方法?

用户也很难一次性把其真实的要求完全提交给软件公司。用户开始阶段提出的需求往往只是对系统的期望和比较模糊的设想。软件公司明确用户需求的最佳方式就是为用户提供原型并由用户进行评价。

3.选择模型的条件:

(1)已有产品或产品的原型，只需客户化的项目。

(2)简单而熟悉的行业或领域。

(3)有快速原型开发工具。

(4)进行产品移植或升级。

由于上述条件不太苛刻，所以凡是有软件产品的IT企业，在他们熟悉的业务领域内，当客户招标时，他们都会以原型模型作为软件开发模型，去制作投标书，去讲标投标。一旦中标，就用原型模型作为实施项目的指导方针，即对软件产品进行客户化工作，或对软件产品进行二次开发。

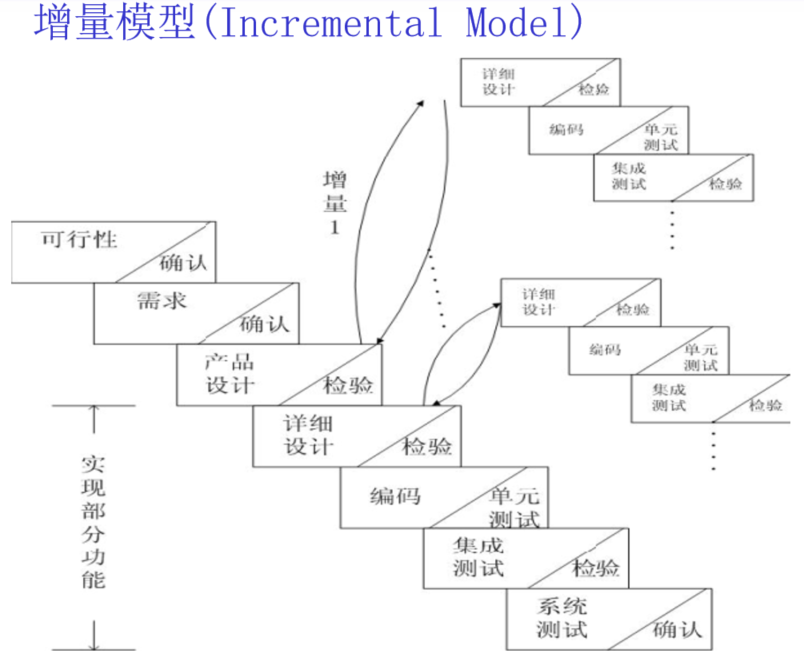
4.模型的优点:

(1)需求明确化

(2)可作为理解和确认软件需求规格说明的工具(3)该方法强调软件开发的反复性，反映了软件开发的真正本质(4)提供了一种有利的学习手段

5.模型的缺点:

因为事先有一个展示性的产品原型，所以在一定程度上，不利于开发人员的创新。



增量模型(Incremental Model)将软件产品看作一组增量构件，每次设计、实现、集成、测试和交付一块构件，直到所有构件全部实现为止。

1.模型的本意:要开发一个大的软件系统，先开发其中的一个核心模块，然后再开发其他模块，这样一个个模块地增加上去，就象搭积木一样，直至整个系统开发完毕为止。

2.模型的特点:(1)任务或功能模块驱动，可以分阶段提交产品;有多个任务单，这些多个任务单的集合，构成项目的一个总任务书(总用户需求报告)。

3.选择模型的条件:

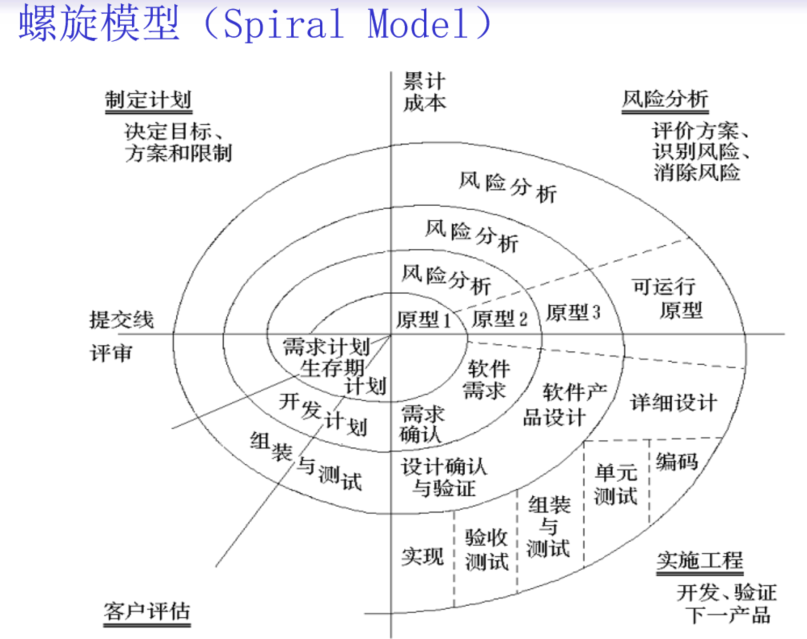
(1)在项目开发过程中，客户接受分阶段交付:开发人员对应用领域不熟悉，难以一步到位。(2)工期过紧的中等或高风险项目。(3)用户可参与到整个软件开发过程中。(4)

(5)使用面向对象语言或第四代语言。

(6)软件公司自己有较好的类库、构件库:

4.模型的缺点:

若软件系统的组装和拆卸性不强，或开发人员全局把握水平不高(没有数据库设计专家进行系统集成)，或者客户不同意分阶段提交产品，或者开发人员过剩，都不宜采用这种模型。



1.模型的本意:使用原型及其他方法尽量降低风险，

2.模型的优点:

(1)有利于已有软件的重用;

(2)有助于把软件质量作为软件开发的一个重要目标;(3)减少了过多测试或测试不足所带来的风险;

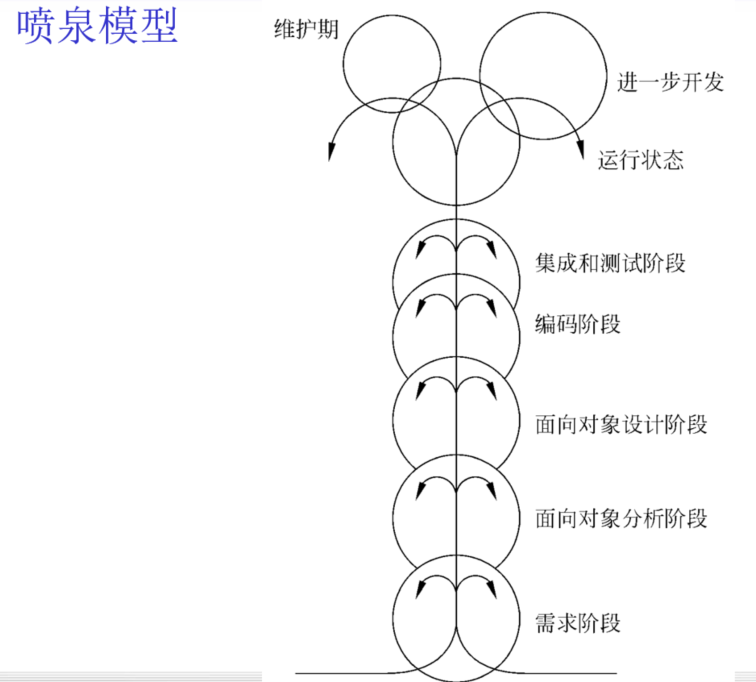
软件维护与软件开发没有本质区别。4)

3.模型的局限:

使用螺旋模型开发软件，要求软件开发人员具有丰富的风险评估知识和经验。

4.模型的适用范围:

主要适用于内部开发的大型软件项目



1.模型的本意:

以用户需求为动力、以对象为驱动，采用面向对象的软件开发过程。

2.模型的优点:可以同步开发，节省开发时间。

3.模型的缺点:

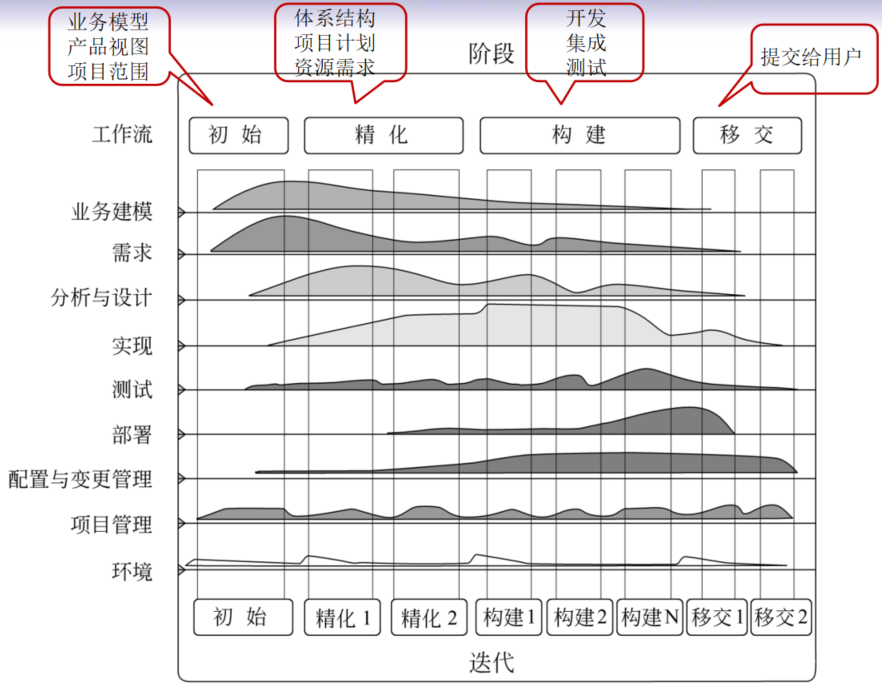
需要大量的开发人员，不利于项目的管理:要求严格管理文档，增大审核难度。

1. 模型的适用范围:面向对象的软件开发。

智能模型

智能模型也称为基于知识的软件开发模型，它综合了上述若干模型，并把专家系统结合在一起。该模型应用基于规则的系统，采用归纳和推理机制，帮助软件人员完成开发工作。为此，建立了知识库，将模型、软件工程知识与特定领域的知识分别存入数据库。





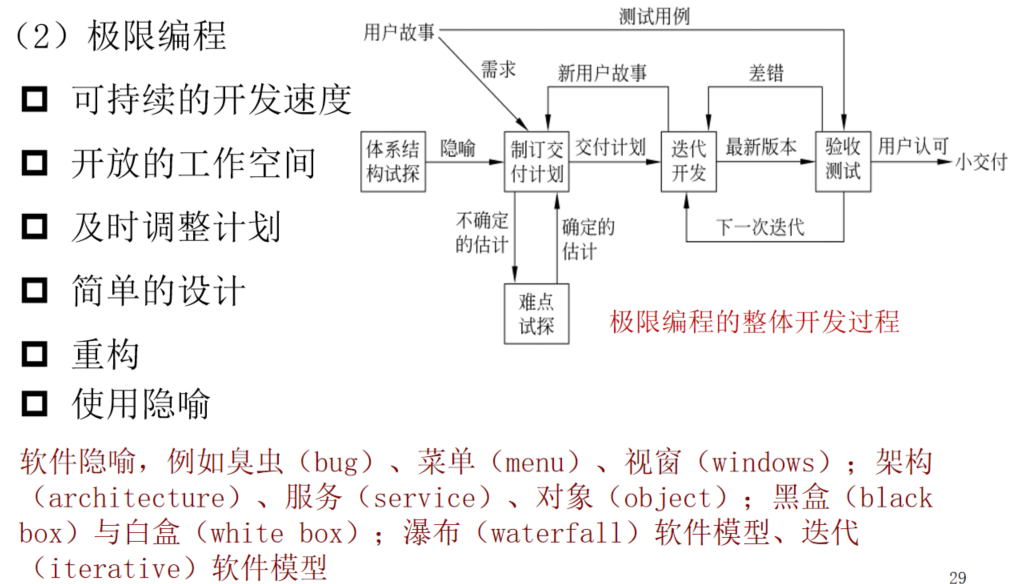
2.敏捷过程与极限编程

(1)敏捷过程(小规模团队敏捷开发）

个体和交互胜过过程和工具；可以工作的软件胜过面面俱到的文档；客户合作胜过合同谈判；相应变化胜过遵循计划；通过尽早的、不断地提交有价值的软件来使客户满意；

(2)极限编程(Extreme programming,XP是一种轻量、高效、低风险、柔性、可预测、科学而充满乐趣的软件开发方法)

客户作为开发团队成员；使用用户素材；短交付周期；验收测试；结对编程；测试驱动的开发；集体所有；持续集成；



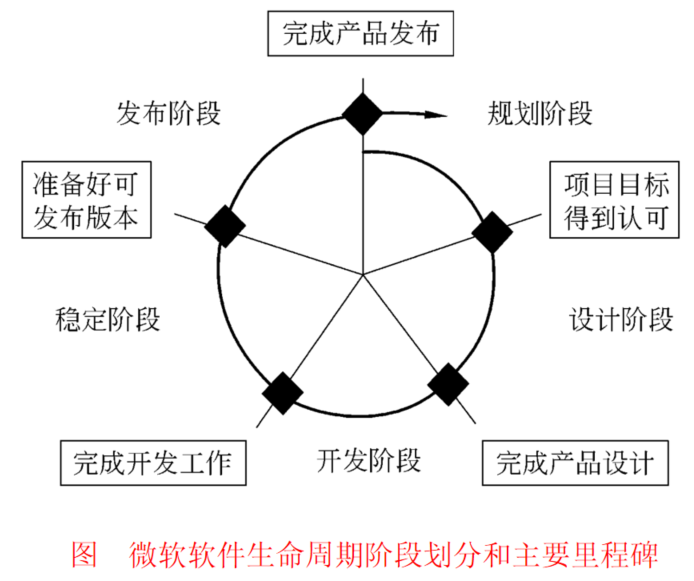
3.微软过程

(1)微软过程准则

项目计划应该兼顾未来的不确定因素；有用效的风险管理来减少不确定因素的影响/；经常生成并快速地测试软件的过渡版本，以提高软件的稳定性和可预测性；

采用快速循环、递进的开发过程；项目进度表应该具有较高稳定性和权威性；使用小型项目组并发地完成开发工作；使用原型验证概念，对项目进行早期论证；

把零缺陷作为追求的目标；里程碑评审会的目的是改进工作，切记相互指责；



AI增强的敏捷开发:新一代软件工程方法论

随着大语言模型(LLM)的兴起，AI已经从辅助工具逐渐转变为开发流程中的核心驱动力。软件工程进入了一个人机协作，从传统的“手工开发”转向“AI辅助开发”的新阶段。

·人类工程师:负责系统设计、逻辑验证和质量把控;·AI:负责代码生成、优化、文档撰写等重复性任务。

开发流程更加实时、并行和自动化。强调Prompt设计与交互，开发过程资产化。

如何画数据流图

在数据流图的外围画系统的输入和输出

集中精力找出数据流，并与外围的输入输出数据流连接起来

给每个数据流命名

为加工命名

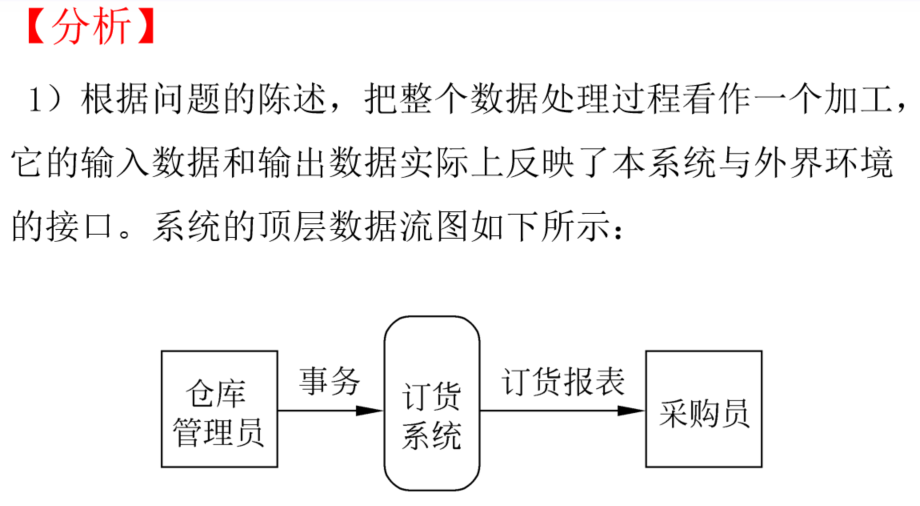
注意:

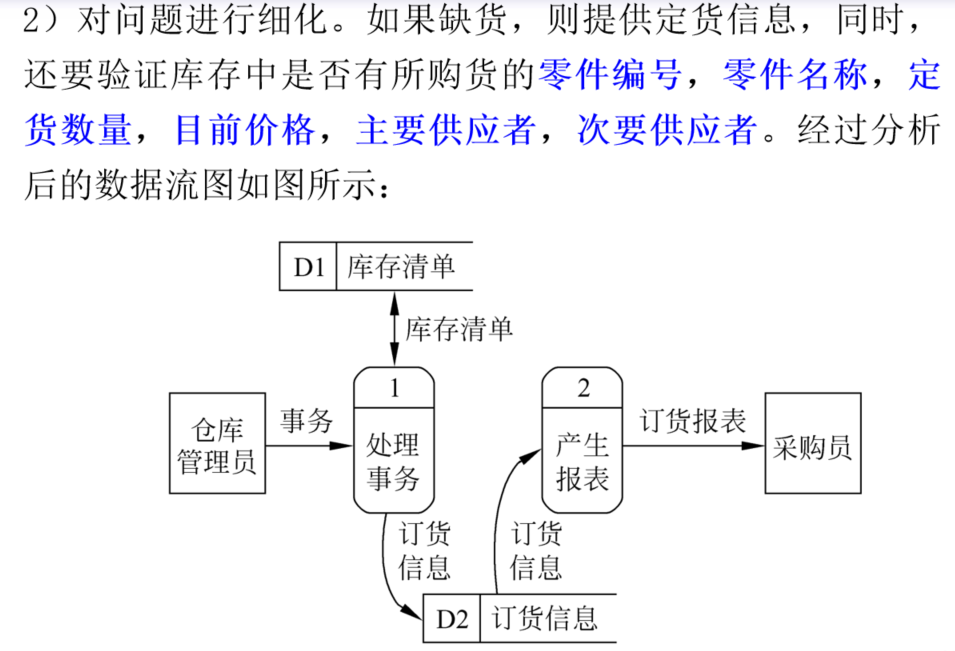
在画数据流图时，应忽视琐碎细节，对一些异常处理，先不要考虑。同时，人类的思维是一种迭代的过程，所以，画数据流图时应不断反复，逐步完善，直到满意为止。

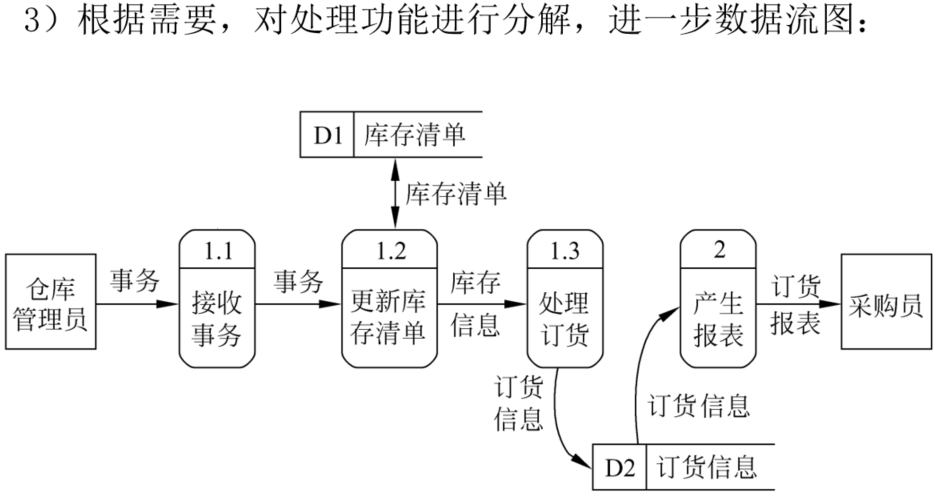
例子1：

某家工厂的采购部每天需要一张定货报表，报表按零件编号排序，表中列出所有需要再次定货的零件。对于每个需要再次定货的零件应该列出下述数据:零件编号，零件名称，定货数量，目前价格，主要供应者，次要供应者。通过CRT终端把事务报告给定货系统。某种零件的库存量少于库存量临界值时就要再次定货。

请给出该系统定货情况的数据流图。

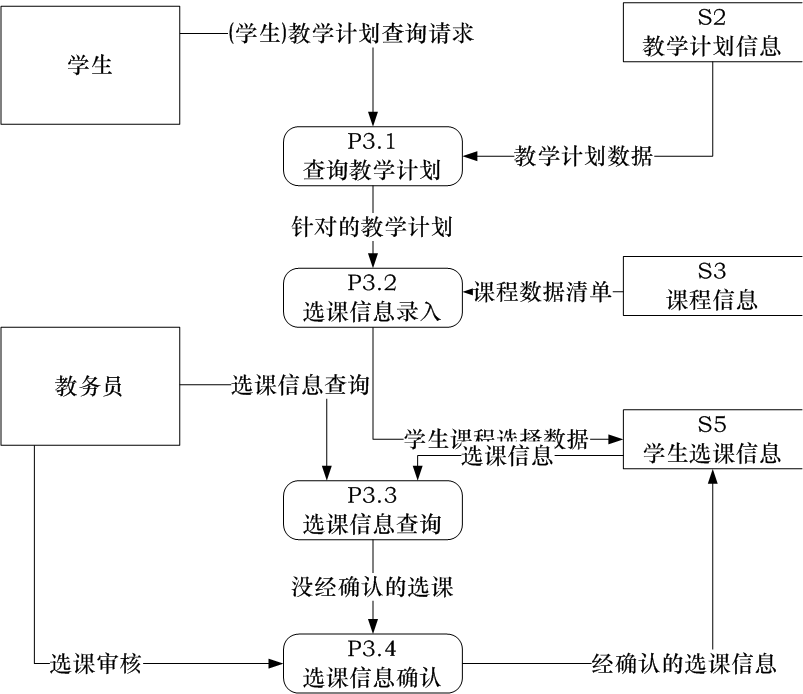






例子1：

选课管理中，学生根据学校对其专业制定的教学计划，录入本学期所选课程，教务员对学生选课记录进行审核，经审核得到的选课就为本学期的选课。

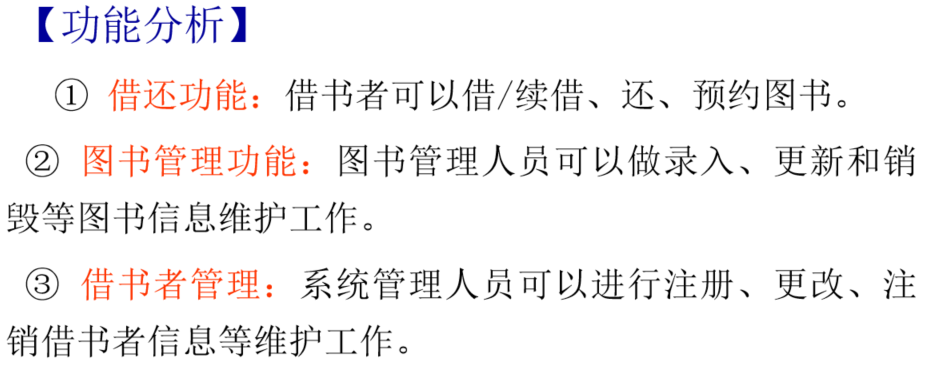
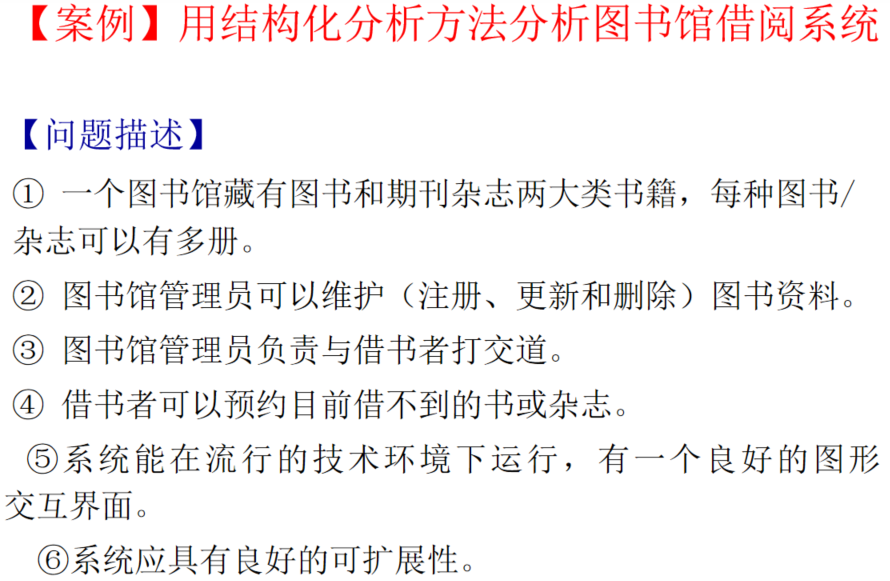


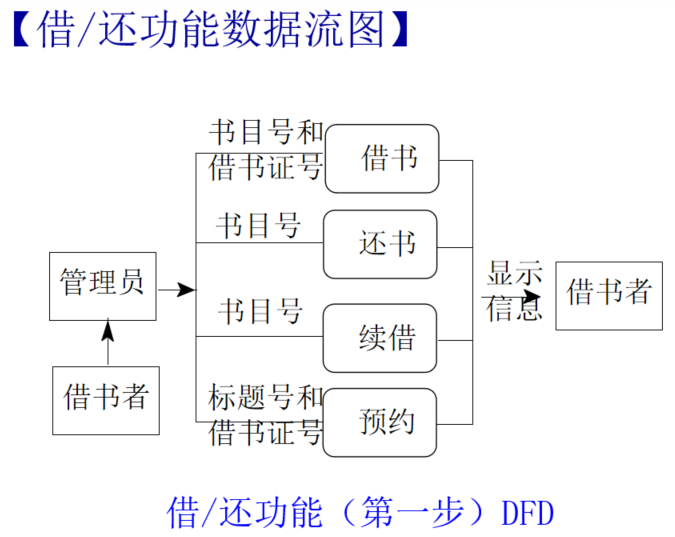
5.优点

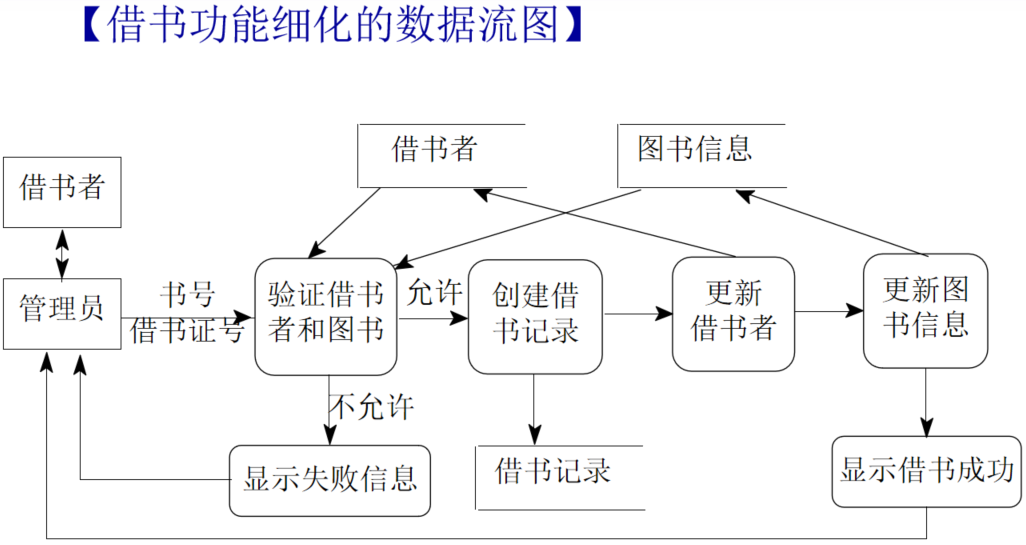
DFD用一种直接而又直观的方法来描述数据的流动和信息系统中的操作。首先，它使用客观图像描述了用户的需求不带有任何个人或组织的说法或观点。其次，数据流图形象真实，便于用户理解和接受，同时也便于系统分析员之间交流信息。最后，数据流图采用自顶向下分解的多层次图形具有一定的抽象性，而且数据流图不强调控制流，突出数据流，便于找到主要矛盾，省略次要细节，从而减少系统的复杂性。

6.缺点

DFD图形符号也存在着一定的缺点。首先，数据流图对时间、界面等方面的内容无法表达。其次，DFD只能做出粗略的模型，而如果需要更精确、更详细的说明时，DFD无法做到。







【建立实体-关系图】

图书馆系统有“图书”和 “借书者”两个实体。在数据流图的分析中发现“图书”实体，没有真正区别借书和预约的关系。借书者是借一本具体的书，即“书目”，而预约图书指的是此类书，而不是指县体的某册书目。因此，要用“标题”实体区分二者，即预约的是书的“标题”，而不是“书目”

外部实体除了“借书者”，还有“系统管理员因此，必须考虑有严格授权进入的权限。

