

《软件工程》 实验指导书

计算机学院

2017年2月

软件工程实验指导

前 言

软件工程实验是为计算机相关专业本科《软件工程》课程配套设置的，是《软件工程》课程讲授中一个重要的、不可或缺的实践环节。其目的是使学生能够针对具体软件工程项目，全面掌握软件工程管理、软件需求分析、软件初步设计、软件详细设计、软件测试等阶段的方法和技术，通过该课程设计使学生进一步理解和掌握软件开发模型、软件生命周期、软件过程等理论在软件项目开发过程中的意义和作用，培养学生按照软件工程的原理、方法、技术、标准和规范，进行软件开发的能力，培养学生的合作意识和团队精神，培养学生对技术文档的编写能力，从而使学生提高软件工程的综合能力，提高软件项目的管理能力。

按该课程的特点，实验内容包括软件开发的两大方法学的专题训练，即结构化（生命周期学）的方法学和面向对象的方法学，通过对一个简单项目，要求学生利用结构化软件开发技术或面向对象的软件开发技术完成对该项目的开发。因此设置五个实验项目，从项目发的准备工作，系统分析过程，系统设计过程，软件测试到系统实施，覆盖软件开发的整个过程，此外又引入我国国家《计算机开发规范》，以规范技术文档的书写标准，提高实验教学质量。

通过实验训练，达到如下目的：

使学生进一步了解和掌握软件工程原理，提高对实际项目的分析和设计能力，通过实验课程，熟悉和基本掌握软件工程方法学、软件开发的过程，文档资料的编写格式及规范，全面领会和贯通所学习的理论知识，从而培养学生综合运用所学课程知识，分析解决问题的能力，培养学生理论联系实际作风，实事求是，严肃认真的科学态度和良好的工作作风，为今后从事科学研究工作打下基础。

实验要求

软件工程实验具体要求如下：

每个项目小组必须按照《软件工程实验指导书》附录中给定的文档规范标准提供项目文档；

题目自定或采用附录二中的题目；

软件开发的方法自定（结构化或面向对象的方法学）。

实验一 用 Visio 进行功能分析和建模

1. 实验目的

掌握结构化分析的方法。

掌握使用 Visio2003 软件绘制数据流图、状态转换图的一般方法和技巧。

2. 实验环境

软件平台:Microsoft Windows XP, 软件工具: Microsoft Visio 2003。

3. 实验原理

结构化分析方法以数据字典为核心,采用实体关系图、数据流图和状态转换图等图形来表达需求,直观明了且易于理解和掌握。

数据流图作为功能建模的基础,描述数据怎样转换以及转换的功能,状态转换图作为行为建模的基础,表示系统的各种行为状态以及状态间的转换方式。

4. 实验内容与要求

绘制学生成绩管理系统(案例如下)的数据流图及状态转换图。

5. 撰写实验报告

案例 1

某校准备开发一个学生成绩管理系统。在该系统中,教务人员录入学生信息、课程信息和成绩信息,学生可以随时查询自己所选课程的成绩。由于学生成绩属于敏感信息,系统必须提供必要的安全措施以防非法存取。

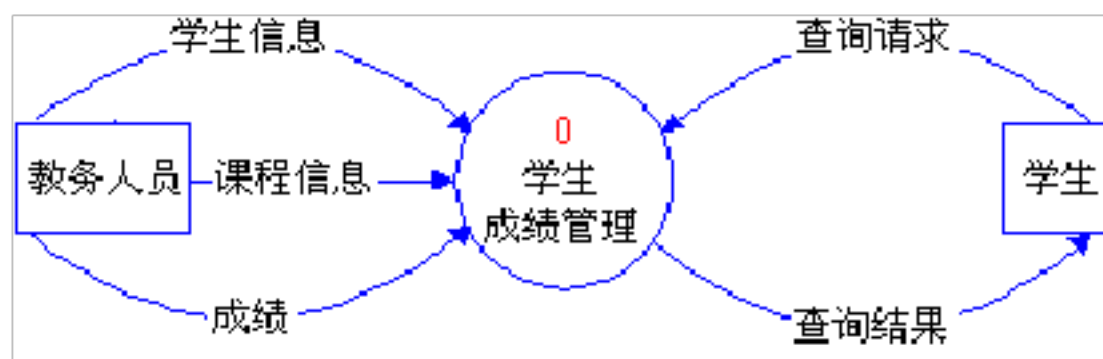
用 Visio 操作

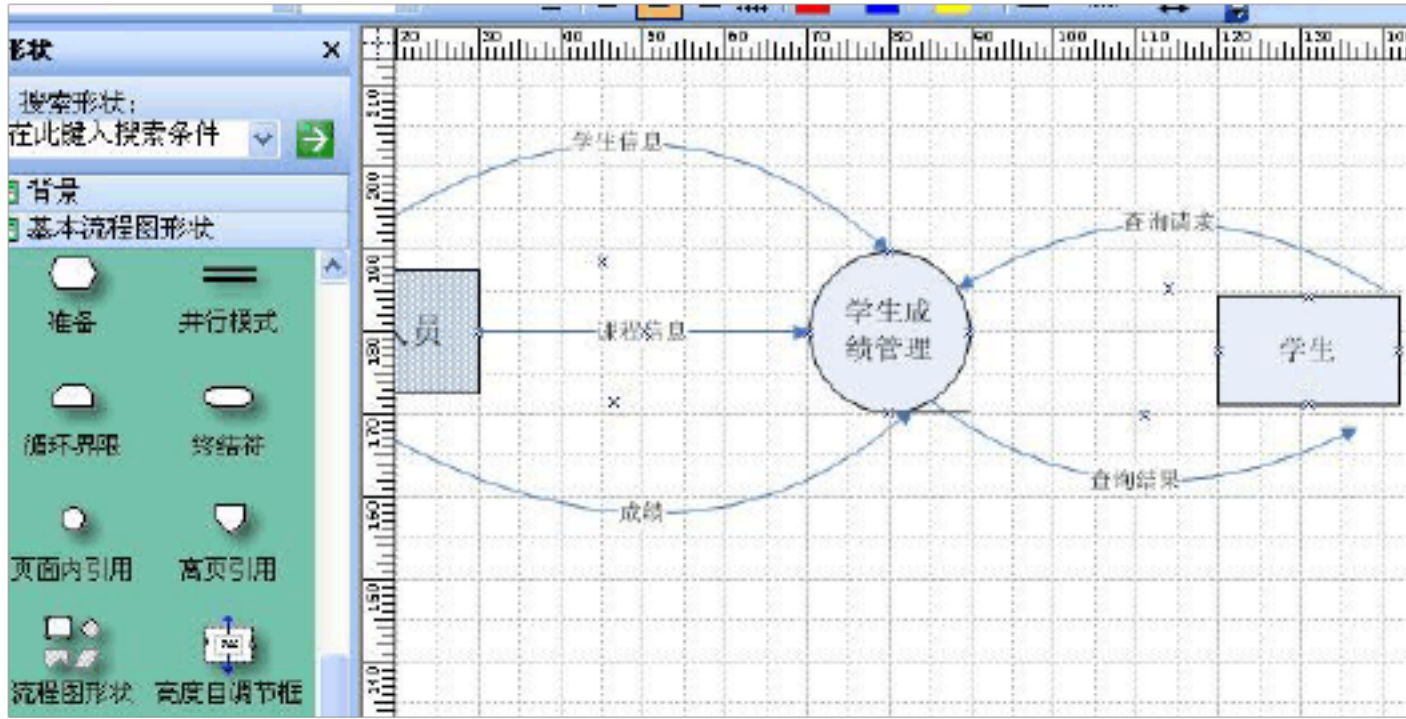
实验步骤及相关详细讲解:

* 第 0 层 DFD 图

教务人员维护学生信息和课程信息,并登录学生的选课成绩;

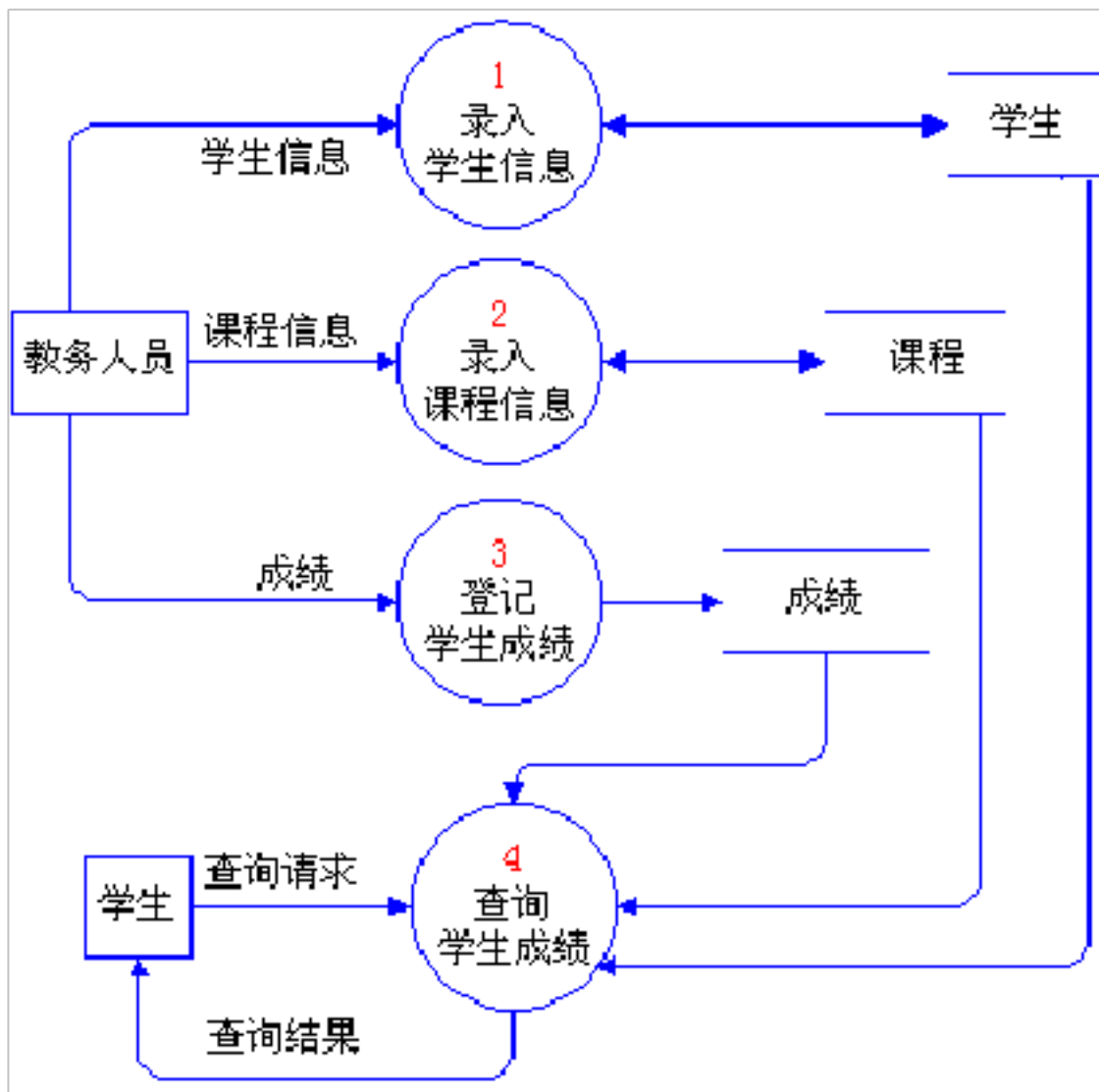
学生查询自己的成绩单。



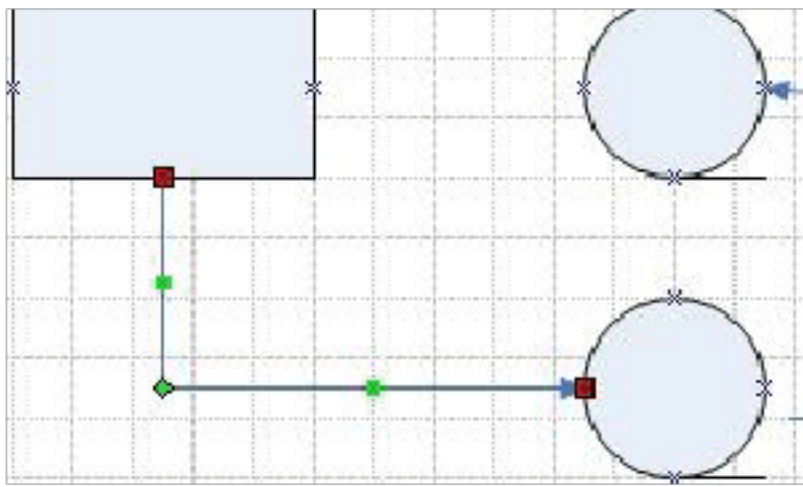
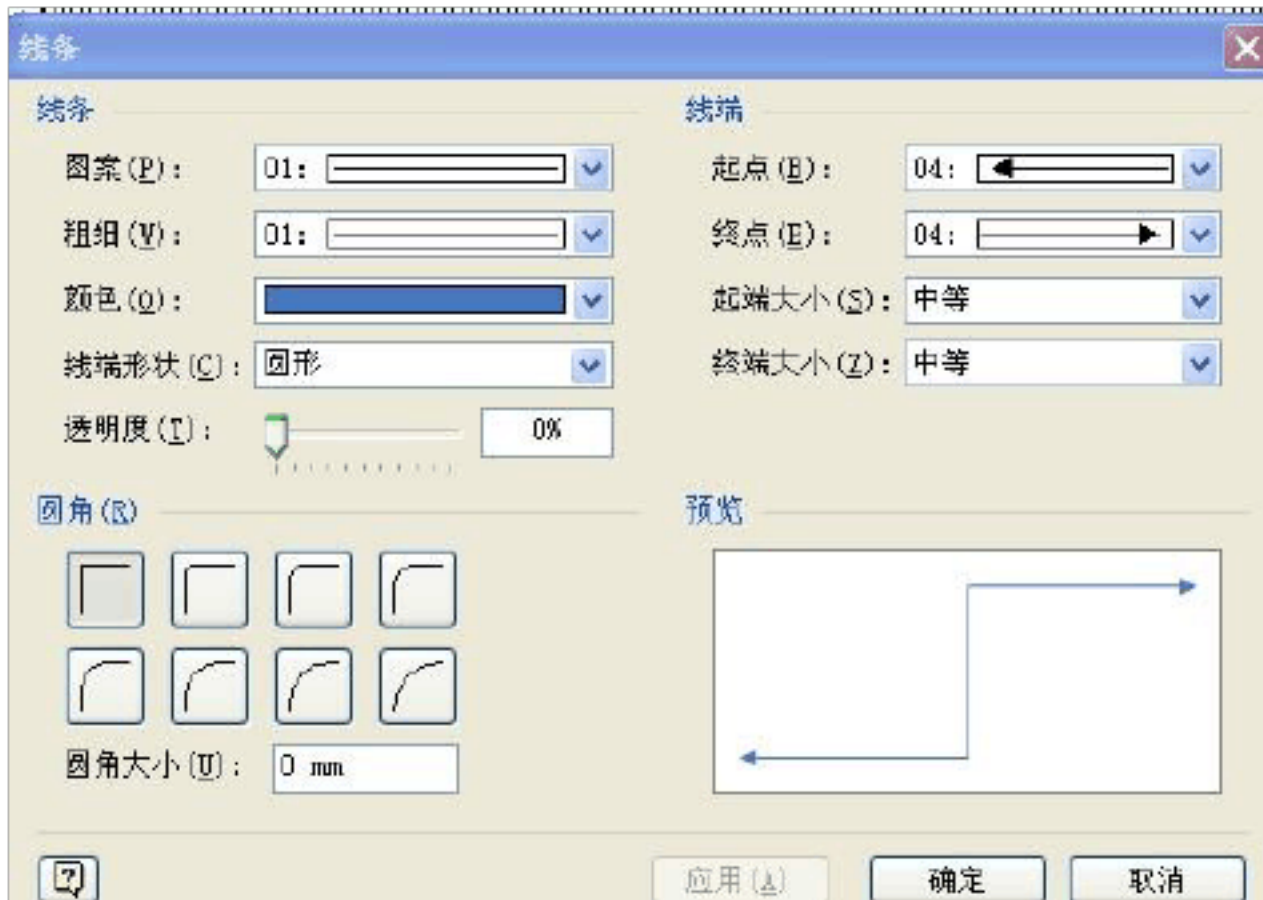


* 第 1 层 DFD 图

对第 0 层 DFD 图中的一个加工 学生成绩管理 进行展开。



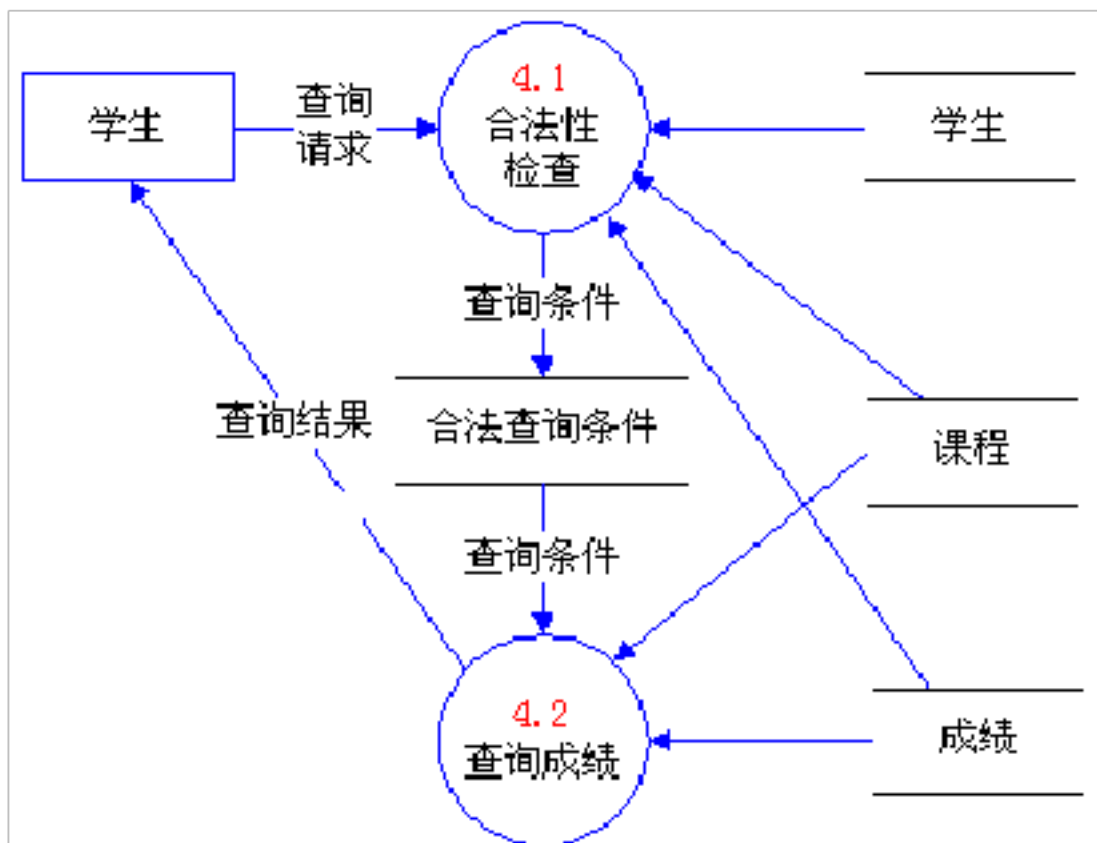
双箭头：直线——右键格式——线条，线端的起点终点



直线用动态连接线

* 第2层 DFD 图

对第1层 DFD 图中的一个加工 查询学生成绩 进行展开。



绘制第 0 层 DFD 的时候，将整个系统看成一个加工，然后找出作用于该加工的外部实体，以及相应的数据输入和输出。对于 学生成绩管理系统 而言，整个系统就是一个加工 学生成绩管理 。从用户的需求描述可知， 教务人员 是数据的源点， 学生 是数据的终点。另外，教务人员需要录入学生信息、课程信息和成绩，说明 学生信息 、 课程信息 和 成绩 是数据流；同样， 查询请求 和 查询结果 也是数据流。根据上述分析，得到如图所示的第 0 层 DFD。

绘制下一层数据流图时，细化第 0 层的加工 学生成绩管理 ，从而描述系统的主要功能。从第 0 层 DFD 得知， 学生信息 是教务人员需要录入的一个信息，因此加入一个加工“录入学生信息”，同样得到 录入课程信息 、 登记成绩 两个加工。另外，数据流 查询请求 和 查询结果 应该由加工 查询成绩 来完成。这样，我们用 录入学生信息 、 录入课程信息 、 登记学生成绩 和 查询学生成绩 四个加工代替第 0 层的 学生成绩管理 ，同时增加这些数据流对应的数据存储，即 学生 、 课程 和 成绩 ，最后得到如图所示的第 1 层 DFD。

为了继续进行分解，我们分析第 1 层 DFD 中的加工 查询学生成绩 。学生查询成绩时需要提供合法性检查，因此， 查询学生成绩 可以分解为 合法性检查 和 查询成绩 两个处理步骤，从而形成如图所示的第 2 层 DFD。

根据以上实例和经验，绘制数据流图应当遵循以下原则：

- (1) 分层时，子图的输入、输出数据流必须和父图中相应加工的输入、输出数据流一致；
- (2) 加工的编号应该唯一且具有层次性；
- (3) 加工不应该只有输入或只有输出，通常既有输入又有输出；
- (4) 数据流图不应反映处理的顺序；
- (5) 加工之间应通过数据存储进行通信，避免从一个加工直接流到另一个加工；
- (6) 数据应通过加工进行流动，避免从一个数据存储直接流到另一个数据存储；
- (7) 数据流图中所有元素的命名应当对客户有意义，且与业务相关；
- (8) 不要在一个图中绘制 7 个以上的加工，否则难于绘制和理解。

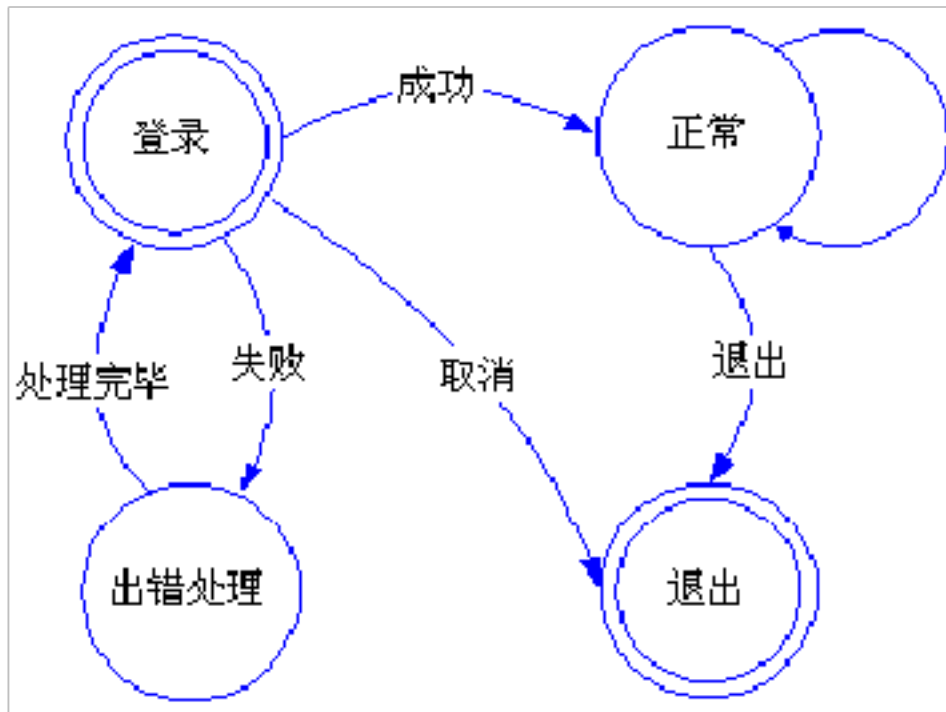
通常来说，行为建模用于实时系统。实时系统中可能存在许多脚本，很多实体需要进行状态划分和描述状态转换图，有时为了描述系统的并发行为，还需要使用其他一些工具进行描述，如 Petri 网。在事务系统中，系统行为相对简单，只有某些行为较复杂的实体才需要建立其状态转换图。

(1) 分析外部事件，所谓外部事件是指外部实体与系统的一次交互。

(2) 分析事件的响应者，该响应者为了响应该事件要进行怎样的活动，这种活动又会激发哪些事件等，这样构成了系统行为的脚本。

(3) 根据事件和活动划分实体的状态，也可根据其他知识划分实体状态，考虑发生怎样的事件使该实体进入这个状态，怎样的事件使该实体从这个状态转换到另一状态等。

举例分析：



(在数据流程图中) 或 UML 图中

在 学生成绩管理 系统中，学生成绩信息需要采取安全措施，我们可以采取登录方法避免非法使用系统。这样，该系统存在 登录、正常 和 出错 等状态的转换。

学生启动系统之后，系统处于 登录 状态。在这种状态下，学生可以进行登录或取消登录。如果取消登录，系统直接退出；如果登录失败，系统进入 出错处理 状态，在显示错误信息后，又重新回到 登录 状态；如果登录成功，系统进入 正常 状态，即显示操作界面，等待学生查询，学生可以多次查询不同课程的成绩，直到学生选择退出为止。

实验二用例模型设计

1. 实验目的

学会 IBM Rational Rose Enterprise Edition 的基本操作。

掌握使用 Rose 进行用例建模。

2. 实验环境

软件平台:Microsoft Windows XP, 软件工具: IBM Rational Rose Enterprise Edition。

3. 实验原理

使用用例方法来描述系统功能需求的过程,就是用例建模,它是实现功能模型建模的主要手段之一。用例模型主要包括以下两部分内容。

(1)用例图(Use Case Diagram)

确定系统中所包含的参与者、用例和两者之间或其自身的关系,用例图是基于系统要实现的功能的一个可视化描述。

① 参与者(Actor)

② 用例(Use Case)

用例是用来描述参与者使用系统,以达到某个目标时所涉及到的一系列的场景的集合。一个用例的核心并不是上述的图标,而是一个规格化的叙述型文档,它描述了参与者要实现某项功能的事件流程,展示和体现了其所描述的过程中的需求情况。用例名称一般以“做什么”即“动宾词组”形式来命名。

③ 用例和参与者及自身的关系

泛化关系(generalization)

包含关系(include)

扩展关系(extend)

(2)用例规约(Use Case Specification)

所谓规约,就是业务规则的规格说明。针对每一个用例,都应该有一个用例规约文档与之相对应,以描述该用户的细节内容。每一个用例的用例规约,都应该包含以下内容:

①用例名称(Use Case Name):用例的名称一般由动词+名词构成,简单说明做什么。

② 简要说明(Brief Description): 简要介绍该用例的作用和目的。

③ 前置条件(Previous Condition): 系统在执行该用例前必须处在的状态。

④ 事件流(Flow of Event)

⑤ 用例场景(Use Case Scenario): 包括成功场景和失败场景,场景主要由基本流和备选流组合而成。

⑥ 特殊需求(Special Requirement): 描述与该用例相关的非功能性需求(性能、可靠性、可用性和可扩展性等)以及涉及约束(所使用的操作系统、开发工具等)。

⑦ 后置条件(Post Condition): 系统在执行完该用例之后应该处在的状态。

4. 实验步骤

(1) 找出系统边界以外的角色(actor),角色是与系统进行交互的外部实体,可以是与系统交互的人员、与系统相连并交换信息的设备和其他系统;

(2) 从这些角色如何与系统进行交互的角度,使用用例(use case)来描述角色怎样使用系统以及系统向角色提供什么功能,用例所表示的是从外部用户角度观察的系统功能;

(3) 绘制用例图,并编写详细的用例描述。用例图只能宏观地描述系统的功能,但却不能提供用例模型所必需的所有信息,每个功能的含义和具体实现步骤则以文本方式描述。

5. 实验内容与要求

绘制用例图,详见教材 P95(4.7)。

6. 撰写实验报告

用例规约及活动图

一、实验目的

- 1.熟悉活动图的基本功能和使用方法。
- 2.掌握用例规约的撰写。
- 3.掌握如何使用建模工具绘制活动图方法。

二、实验器材

- 1.计算机一台。
- 2.Rational Rose 工具软件。

三、实验原理

1. 用例规约描述用例

单纯使用用例图不能提供用例所具有的全部信息，因此，需要使用文字描述那些不能反映在图形上的信息。用例描述实际上是关于角色与系统如何交互的规格说明，要求清晰明确，没有二义性。描述用例时，应该只注重外部能力，不涉及内部细节。

每一个用例的用例规约，都应该包含以下内容：

- ①用例名称：用例的名称一般由 动词+名词 构成，简单说明 做什么 。
- ② 简要说明：简要介绍该用例的作用和目的。
- ③ 前置条件：系统在执行该用例前必须处在的状态。
- ④ 事件流：基本流和备选流。
- ⑤特殊需求(Special Requirement)：描述与该用例相关的非功能性需求(性能、可靠性、可用性和可扩展性等)以及涉及约束(所使用的操作系统、开发工具等)。
- ⑥后置条件(Post Condition)：系统在执行完该用例之后应该处在的状态 。

2. 活动图描述用例

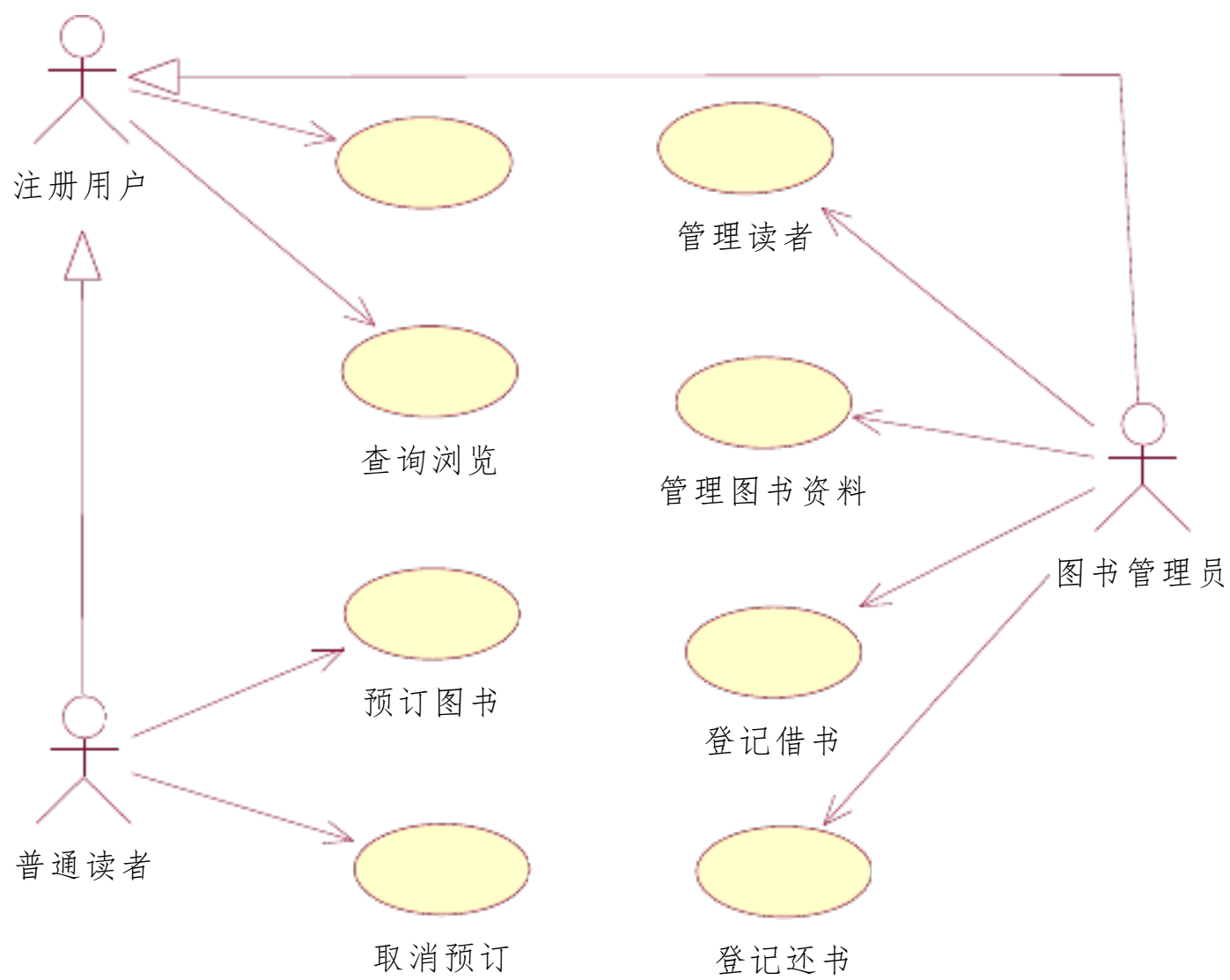
在 UML 中，活动图类似于流程图，它描述了执行某个功能的活动。使用活动图来描述用例，比用例规约更直观。

组成活动图的元素：

- ①活动的起点—实心圆
- ②活动的终点—半实心圆
- ③状态—带圆端的方框
- ④转移—带箭头的直线
- ⑤分支—菱形
- ⑥泳道—将活动图的活动状态分组

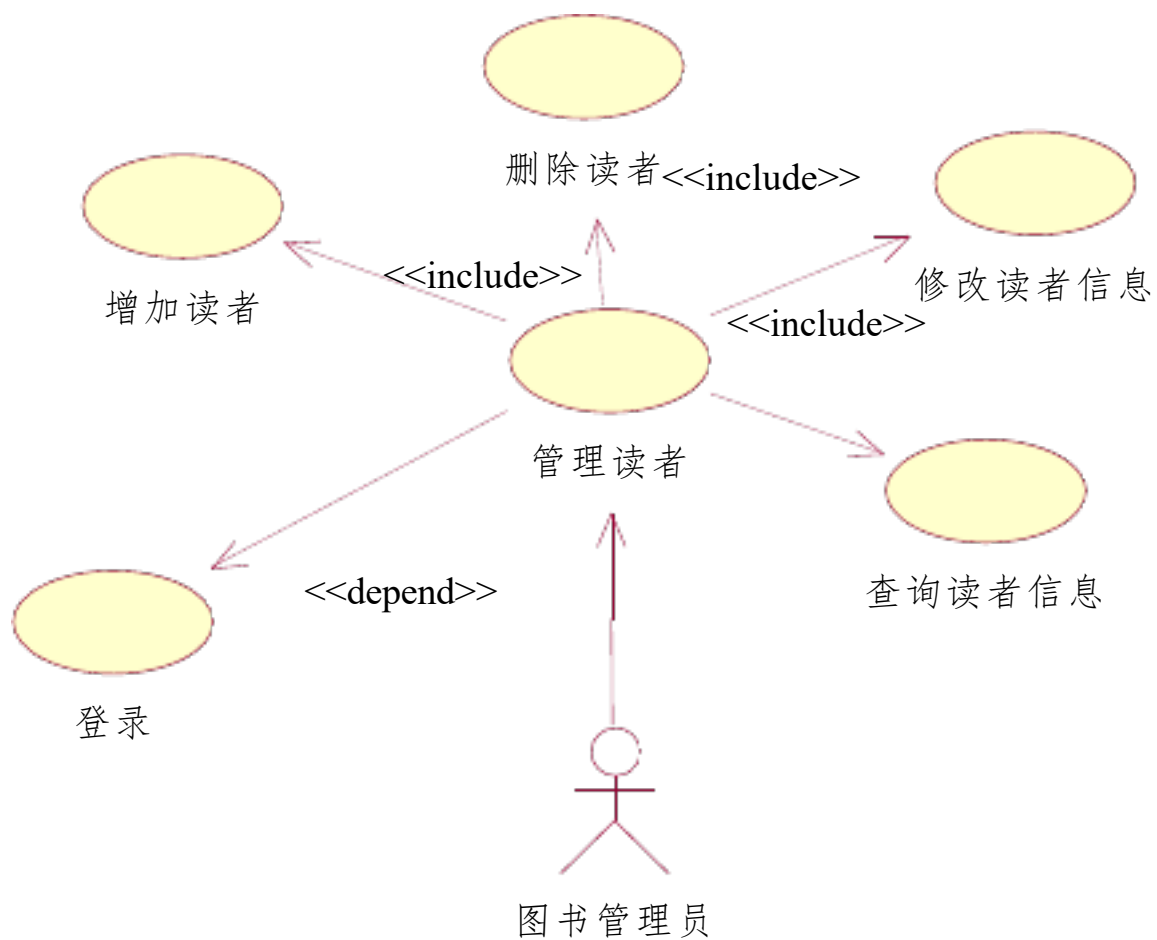
四、实验内容

图书管理系统的用例图如下：



图书管理系统用例图

可进一步添加或细化。其中图书管理员的用例可细化如下(部分):



图书管理员用例图 (部分)

其中删除读者信息一般按照以下步骤进行:

- (1)管理员在录入界面, 输入待删除的读者的信息;

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/247005026015006120>