系统模型

- 需求分析系统的抽象描述

目标

- 解释为什么要在需求工程过程中对系统上下文建模
- 介绍行为建模、数据建模和对象建模
- UML（统一建模语言）中的符号
- CASE工作平台是如何支持系统建模的

内容

- 上下文建模
- 行为模型
- 数据模型
- 对象模型
- CASE工作平台

系统建模

- 系统建模有助于分析者理解系统功能，模型可用来和客户沟通
- 不同的模型从不同的角度来描述系统
  - 从外部来看，是对系统上下文或系统环境建模
  - 从行为来看，是对系统行为建模
  - 从结构上看，是对系统的体系结构和系统处理的数据的结构建模

结构化方法

- 结构化方法提供了系统建模的框架
- 结构化方法定义了一系列模型，导出模型、规则、指南的过程
- 作为结构化方法的一部分，CASE工具支持系统建模

结构化方法的不足

- 不提供对非功能性系统需求的有效理解和建模
- 没有某个方法是否适合某个问题的信息
- 会产生太多的文档,需求要素隐藏在细节描述中
- 系统模型有时候太过具体、使得用户难以理解

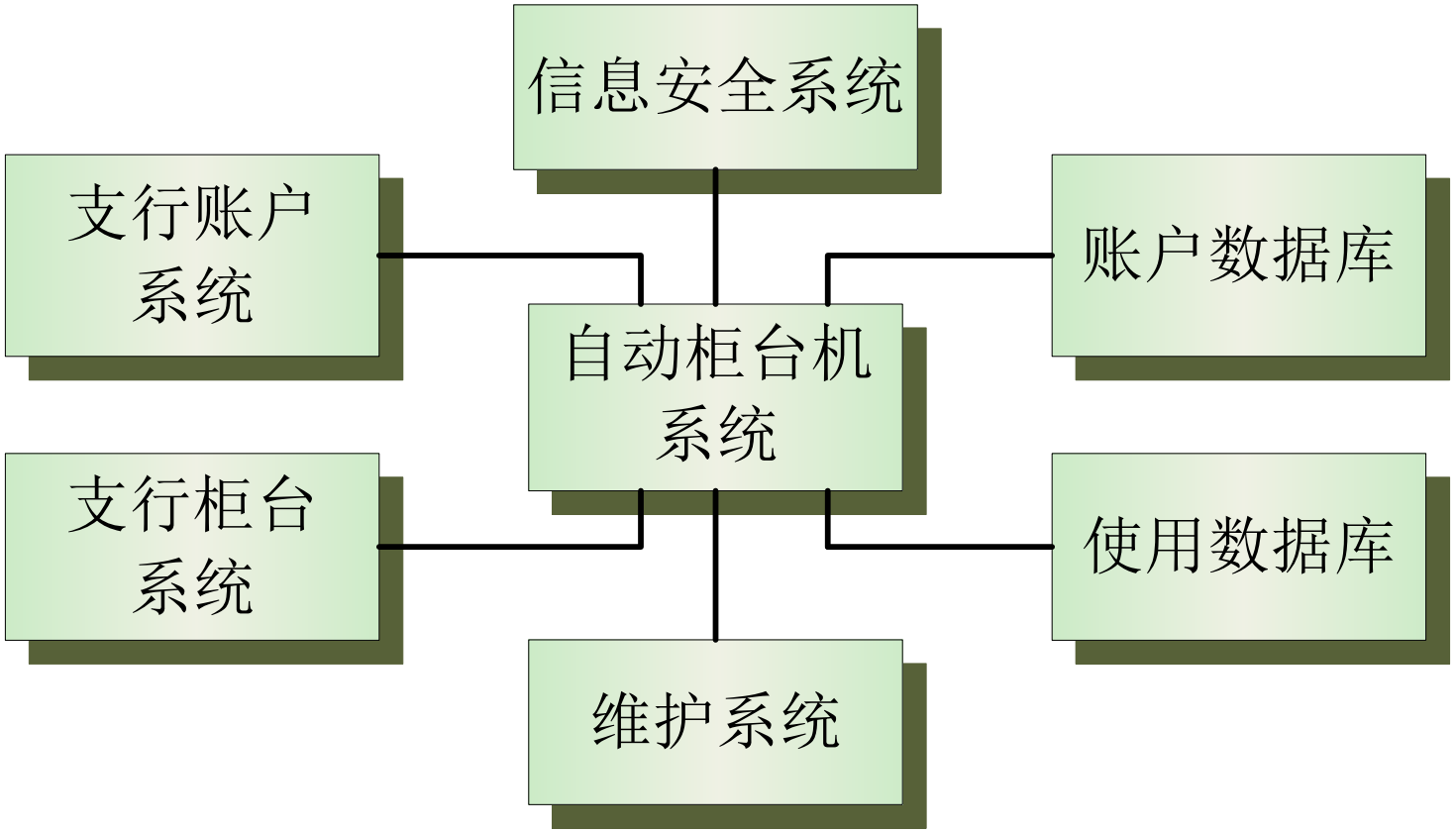
系统模型的类型

- 数据处理模型，说明数据在不同的阶段如何被处理的。（数据流图）
- 组成模型，说明系统中的实体是如何由其它实体组成的。（实体-关系图）
- 体系结构模型，说明构成整个系统的主要子系统。（体系结构图）
- 分类模型，说明实体间怎样具有共同特性。（对象类/继承关系图）
- 激励/响应模型，说明系统对事件的响应。（状态转换图）

上下文模型

- 上下文模型用来说明系统的边界
- 社会和机构的因素会影响系统的边界
- 体系结构模型描述一个系统及其与其它系统之间的关系

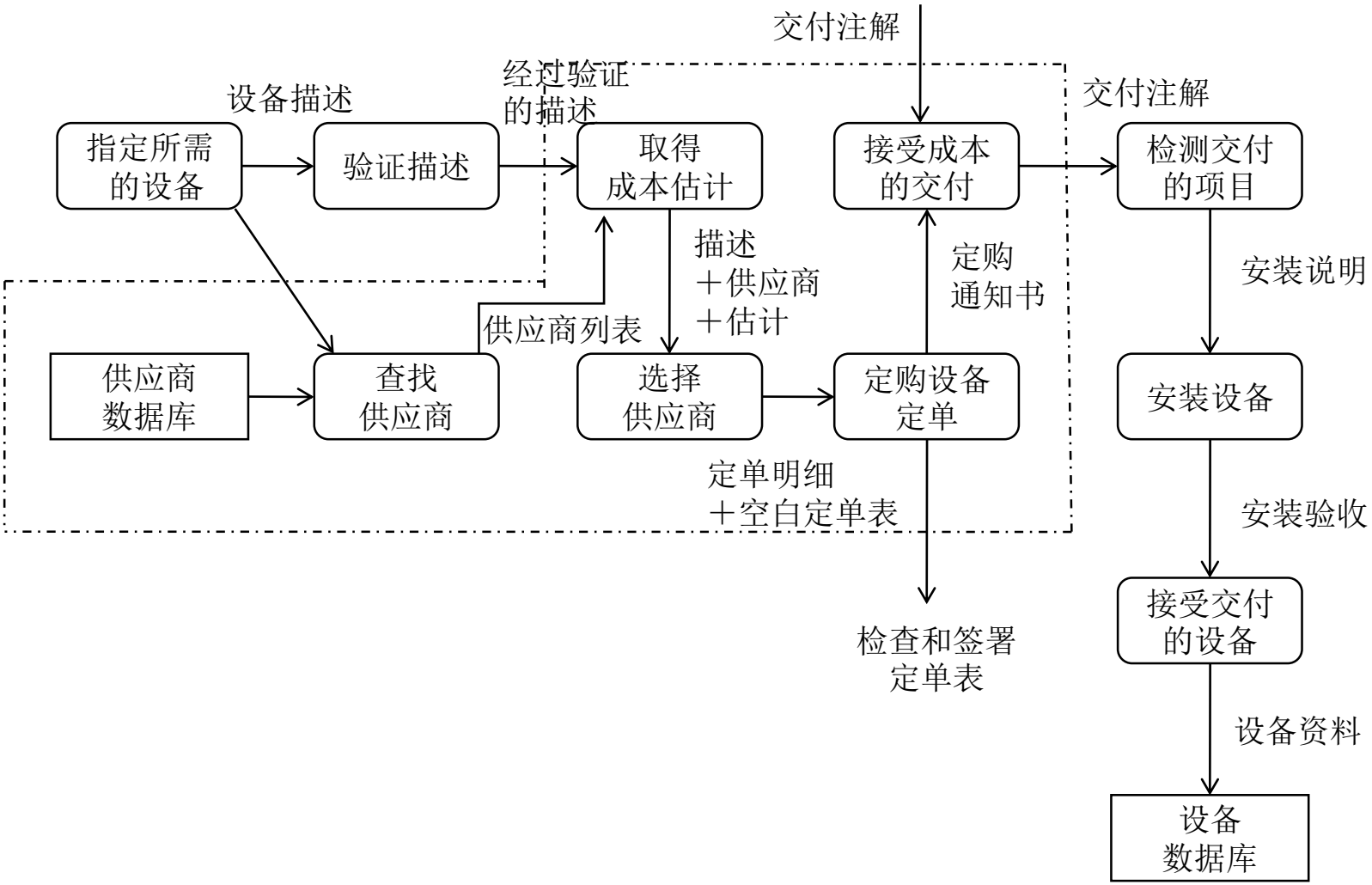
一个ATM系统的上下文



过程模型和数据流模型

- 简单的体系结构模型一般要辅以其它模型共同描述系统
- 过程模型给出系统中支持的各种活动过程
- 数据流模型给出了信息流如何在系统中流动

设备采购过程模型


交付注解
经过验证
的描述
设备描述
交付注解
指定所需
的设备
验证描述
取得
成本估计
接受成本
的交付
检测交付
的项目
描述
＋供应商
＋估计
定购
通知书
安装说明
供应商列表
供应商
数据库
查找
供应商
选择
供应商
定购设备
定单
安装设备
定单明细
＋空白定单表
安装验收
接受交付
的设备
检查和签署
定单表
设备资料
设备
数据库

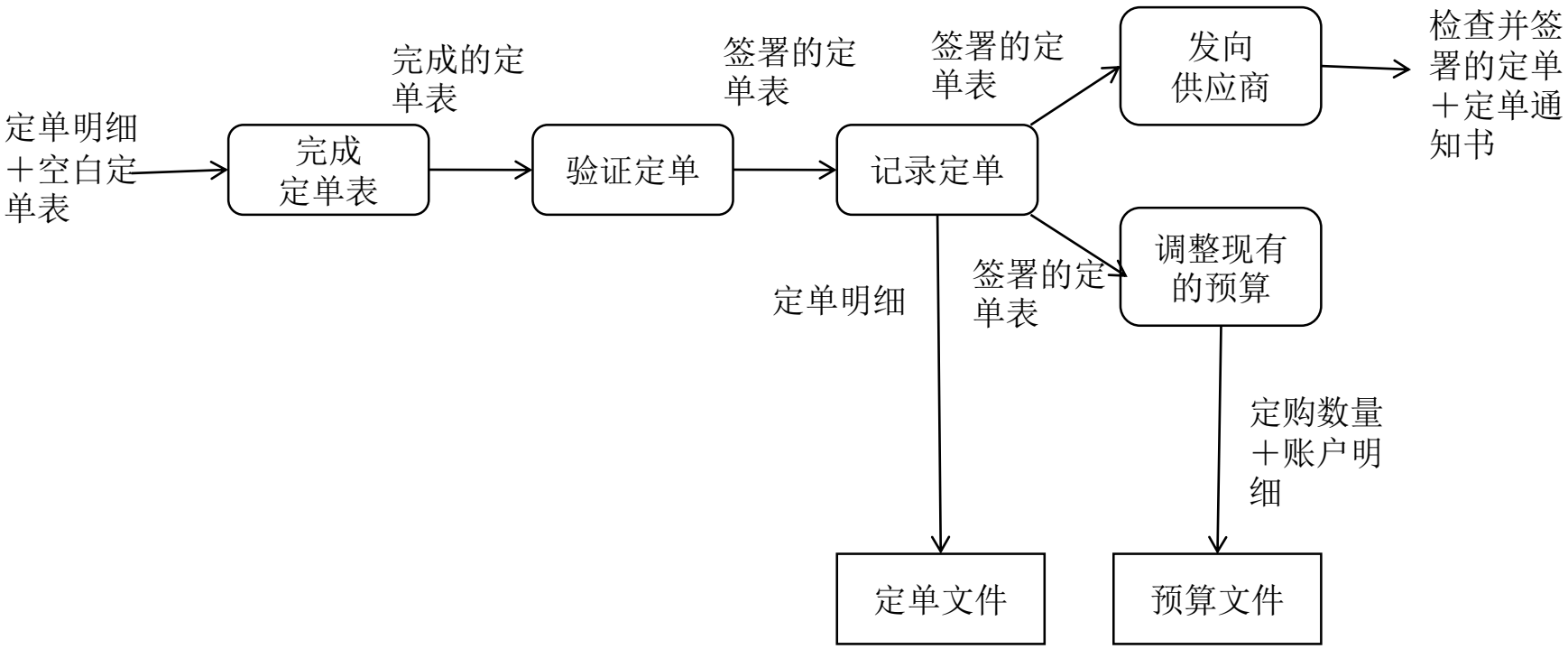
行为模型

- 行为模型是用来描述系统的所有行为
- 两种行为模型：
  - 数据流模型：数据如何在系统中流动
  - 状态机模型：系统对事件的响应
- 两种模型都要求对系统行为作出描述

数据处理模型

- 用数据流图对系统的数据处理过程建模
- 数据的处理步骤就是数据在系统中的流动
- 包含在许多分析方法中
- 客户能够理解的简单直观的符号
- 端到端的数据处理过程，不涉及步骤的内部

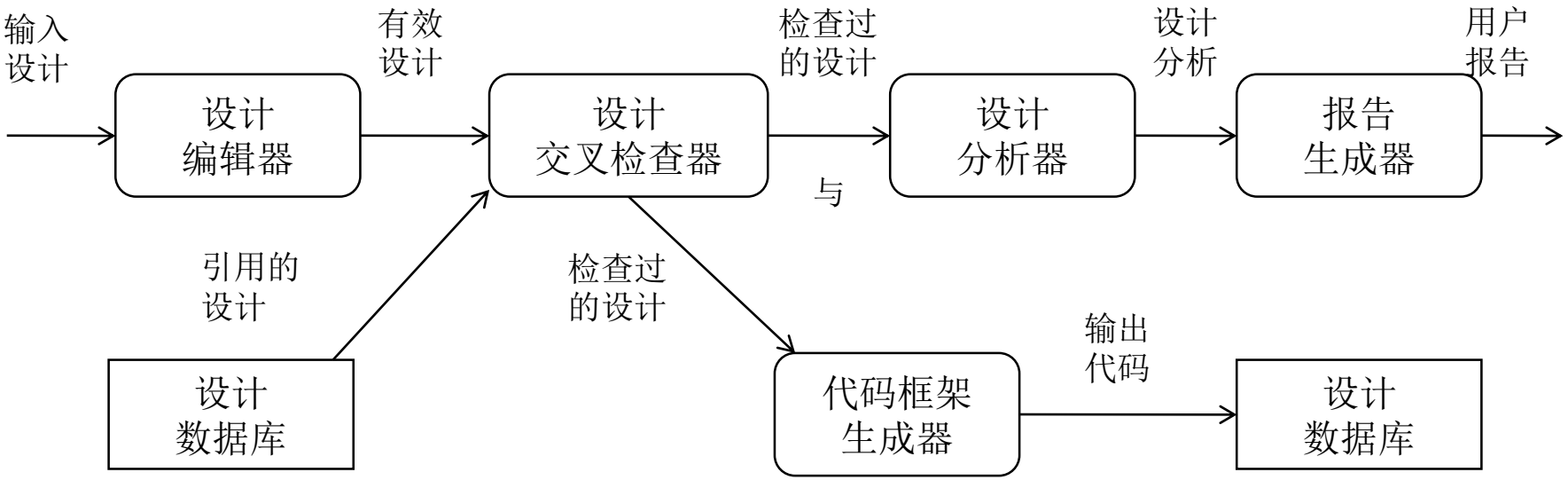
订单处理的数据流图


定单明细
＋空白定
单表
完成
定单表
完成的定
单表
验证定单
签署的定
单表
记录定单
签署的定
单表
发向
供应商
检查并签
署的定单
＋定单通
知书
定单明细
签署的定
单表
调整现有
的预算
定购数量
＋账户明
细
定单文件
预算文件

数据流图

- 数据流图从功能的角度对系统建模
- 追踪数据的处理有助于全面地理解系统
- 数据流图也可用于描述系统和外部系统之间的数据交换

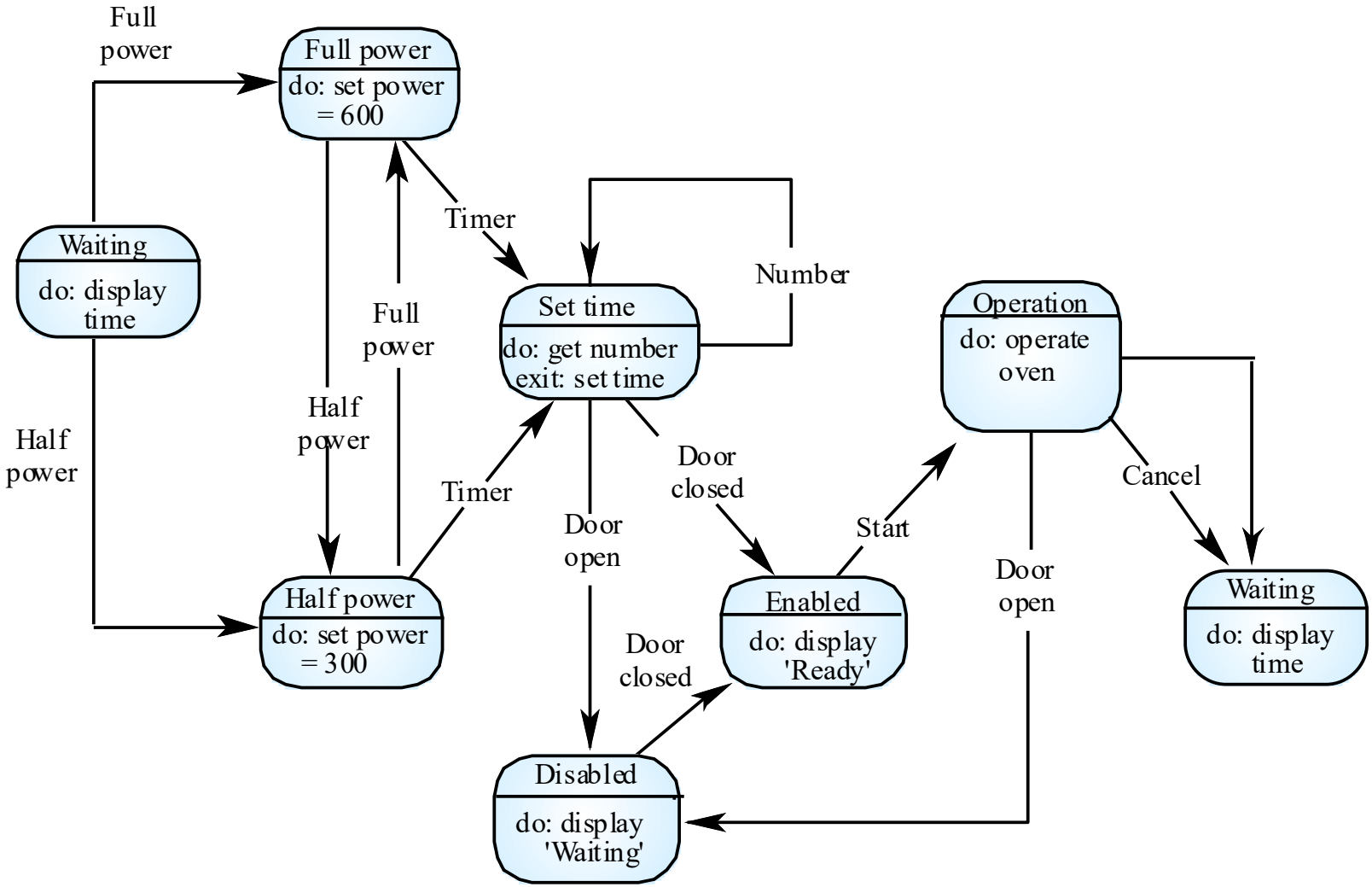
数据流图—CASE工具


输入
设计
设计
编辑器
有效
设计
设计
交叉检查器
检查过
的设计
设计
分析器
设计
分析
报告
生成器
用户
报告
与
引用的
设计
检查过
的设计
输出
代码
设计
数据库
代码框架
生成器
设计
数据库

状态机模型

- 对外部和内部事件的系统响应行为进行建模
- 他们表示系统对激励的响应，经常用来对实时系统建模
- 状态机模型用节点表示系统状态，用节点间的连线表示事件。当一个事件发生时，系统从一个状态转移到另一个状态
- 状态表是UML的一部分

微波炉的模型


Full power
Full power
do: set power = 600
Waiting
do: display time
Timer
Number
Set time
do: get number
exit: set time
Operation
do: operate oven
Full power
Half power
Half power
Timer
Door closed
Door open
Start
Cancel
Door open
Half power
do: set power = 300
Enabled
do: display 'Ready'
Waiting
do: display time
Door closed
Disabled
do: display 'Waiting'

微波炉状态描述

状态	描述
Waiting	等待输入，显示当前时间
Half power	功率设定为300w，显示Half power
Full power	功率设定为600w，显示Full power
Set time	烹饪时间设定为用户输入值，显示选择的烹饪时间，当设定时间时显示刷新
Disable	为安全起见，操作被禁用，炉内灯亮，显示Not ready
Enable	操作被启用，炉内灯灭，小时Ready to cook
Operation	处于操作状态，炉内灯亮，显示倒计数，当烹饪完成时蜂鸣器响5秒，外部灯亮，显示Cooking complete

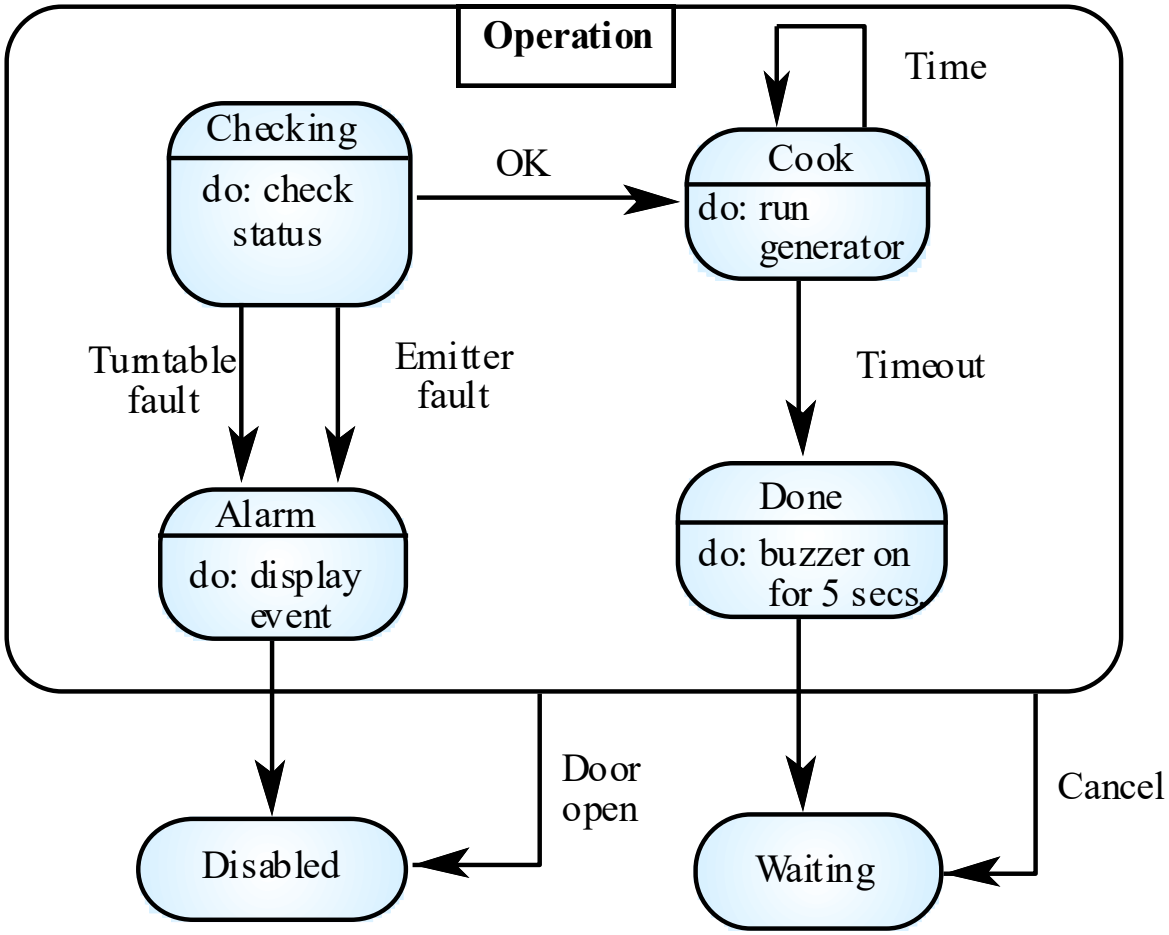
微波炉的激励

激励	描述
Half power	用户按下了半功率按钮
Full power	用户按下了全功率按钮
Timer	用户按下了定时器按钮
Number	用户按下了数字键
Door open	微波炉的门开关处于开门状态
Door closed	微波炉的门开关处于关门状态
Start	用户按下启动按钮
Cancel	用户按下取消按钮

状态表

- 允许将一个模型分解成几个子模型
- 在每个状态的操作内容中列出动作的简要描述
- 可以用表格的形式描述状态和激励

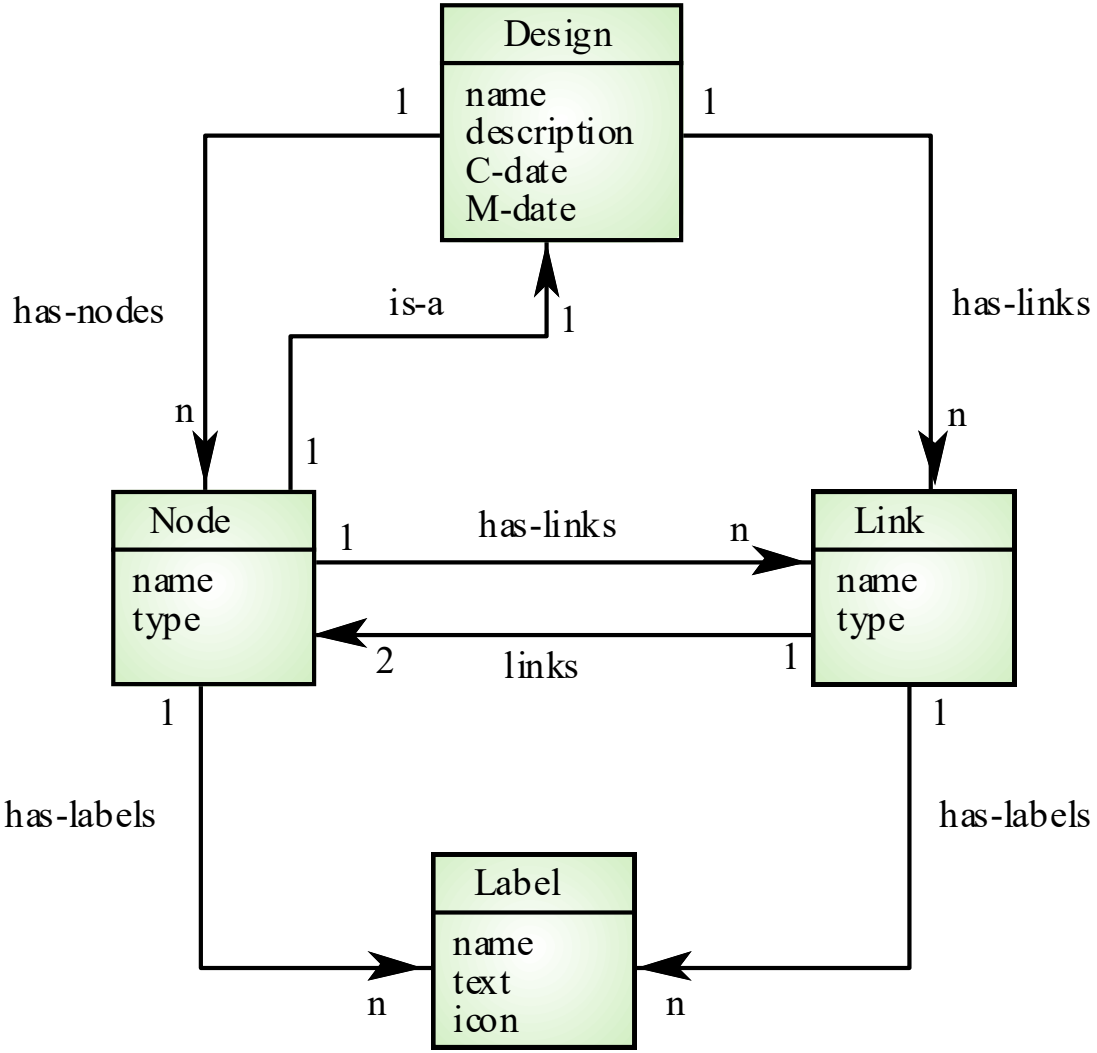
微波炉的操作


Operation
Time
Checking
do: check status
OK
Cook
do: run generator
Turntable fault
Emitter fault
Timeout
Alarm
do: display event
Done
do: buzzer on for 5 secs.
Door open
Cancel
Disabled
Waiting

语义数据模型

- 用来描述系统数据加工的逻辑结构
- 实体-关系-属性模型列出系统中的实体、这些实体及其属性之间的关系
- 广泛运用于数据库设计。用关系数据库比较容易实现。
- 在UML中没有提供特定的符号，可以使用对象和连接

软件设计的语义模型


Design
name
description
C-date
M-date
1
1
has-nodes
is-a
1
has-links
n
1
n
Node
name
type
1
has-links
n
Link
name
type
2
links
1
1
1
has-labels
has-labels
Label
name
text
icon
n
n

数据字典

- 数据字典是系统模型中用到的所有名称的列表。包括实体、关系和属性的描述
- 优点
  - 支持名字管理、避免重复
  - 机构知识的储备和分析、设计、实现连接起来
- 许多CASE工作平台支持数据字典

数据字典实体

名字	描述	类型	日期
Has-labels	在节点或关联实体和类型标签实体间的1：N关系	关系	5.10.1998
Label	存放节点或关联的结构化或非结构化信息。标签由一个图标和相关的文本表示	实体	8.12.1998
Link	表示设计实体的节点间的1：1关系，关联具有类型和名字	关系	8.12.1998
name(label)	每个标签具有一个说明类型的名字。该名字在设计中的标签类型必须唯一	属性	8.12.1998
name(node)	每个节点名字在整个设计中必须唯一，名字可以长达64个字符	属性	9.11.1998

对象模型

- 对象模型描述系统的对象类
- 一个对象类是一系列具有共同属性和服务的对象的抽象描述
- 不同的对象模型
  - 继承模型
  - 对象聚合模型
  - 对象交互模型

对象模型

- 比较自然地反映了系统所处理的真实世界中的实体
- 越是抽象的实体，用这种方法建模越是困难
- 对象类识别是一个困难的过程，需要对应用领域有一个深入的理解
- 反映领域实体的对象类，在系统间是可复用的。

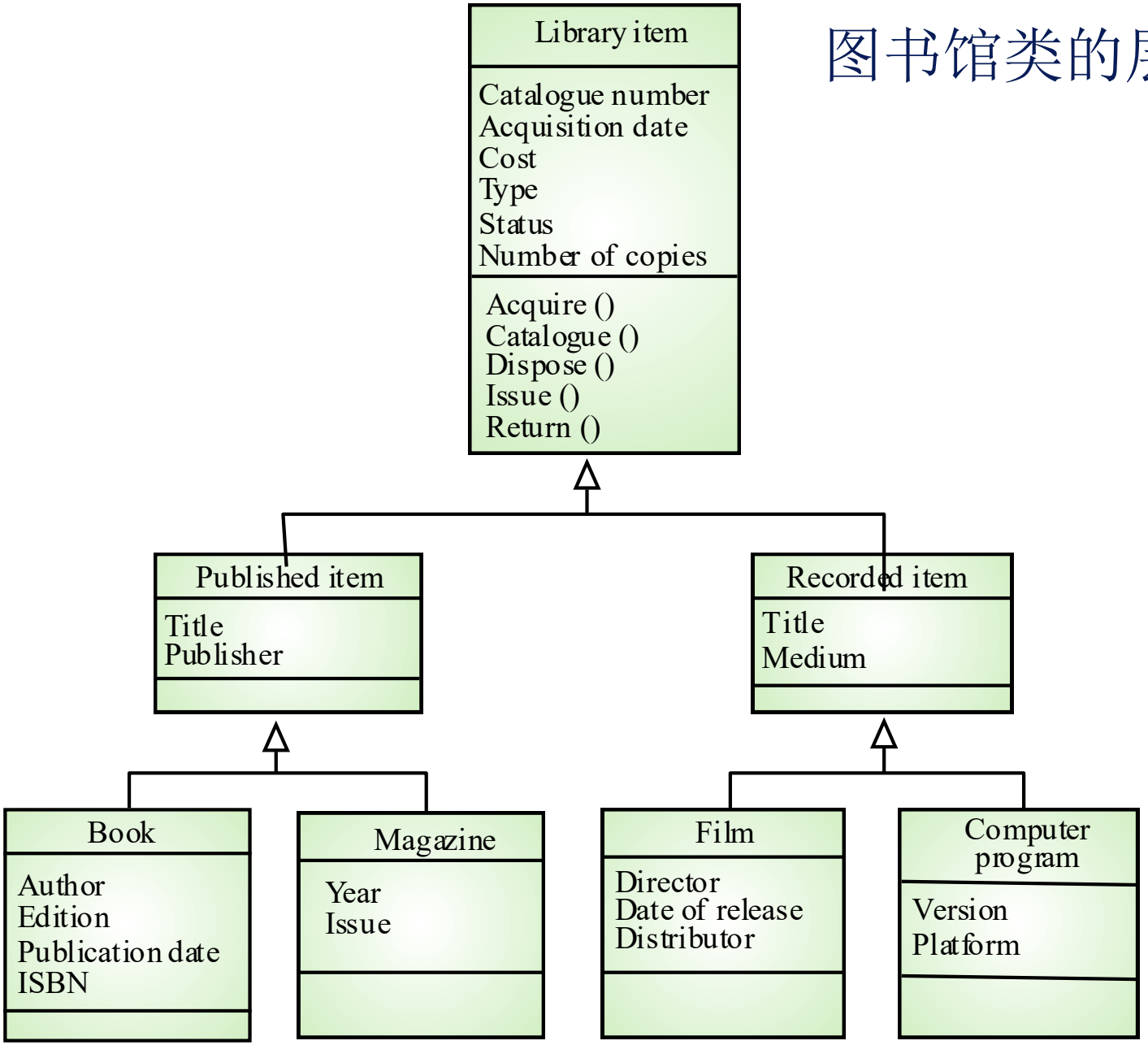
继承模型

- 将领域对象类组织成层级结构
- 在层级顶端的类反映了所有类的共同特征
- 对象类从一个或多个超级类集成它们的属性和服务。根据需要进行具体化。
- 类层级设计是困难的，如果要避免在不同的分支中出现重复

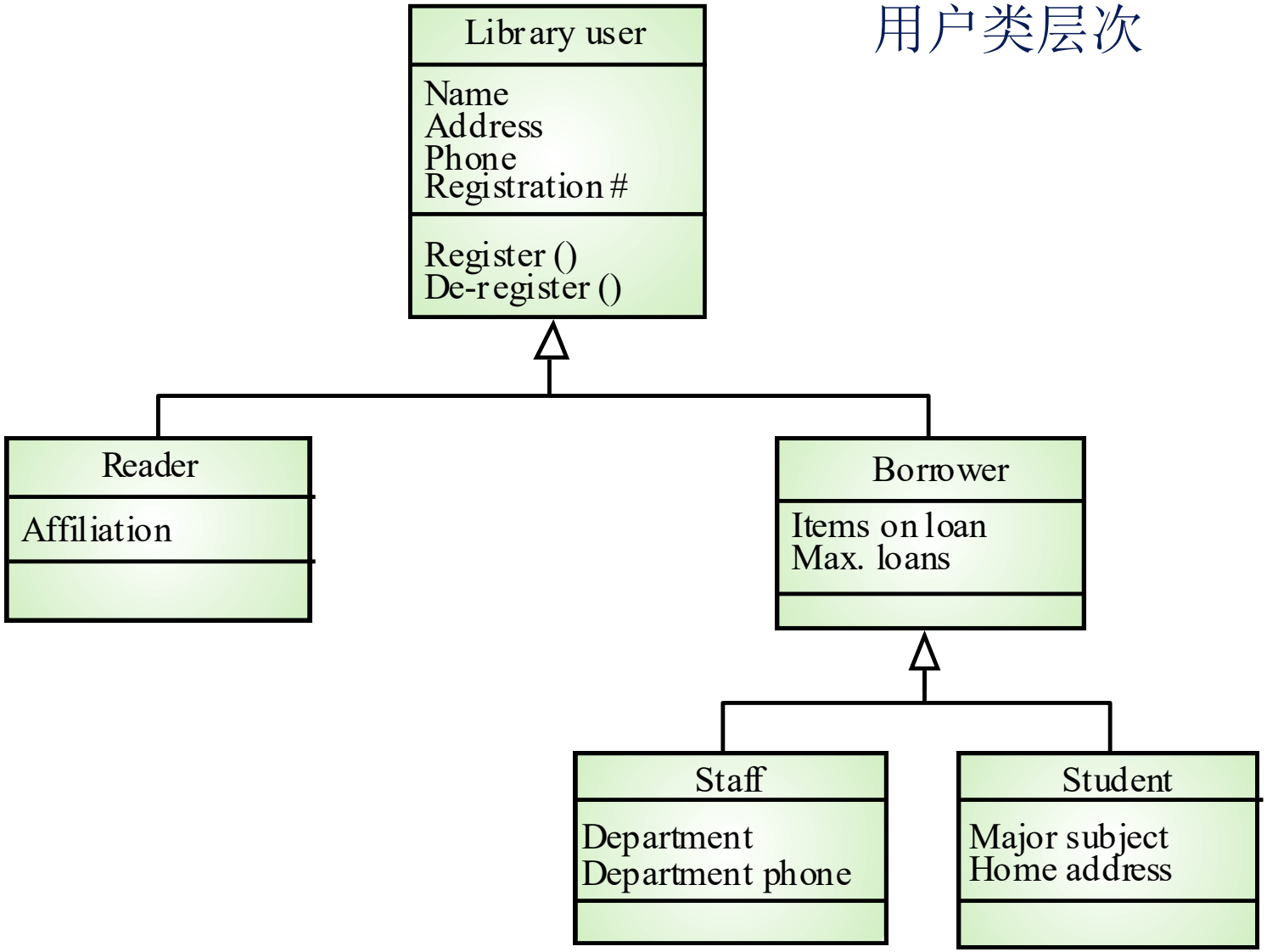
统一建模语言

- 广泛采用面向对象分析和设计方法的开发人员设计的。已经成为面向对象建模的标准化方法
- 符号
  - 对象类用一个矩形表示，名称在上部、属性在中间、操作在下部
  - 对象类之间的关系用对象间的连接线表示
  - 继承是很常用的，在层次图中一般采用从上继承

图书馆类的层次


Library item
Catalogue number
Acquisition date
Cost
Type
Status
Number of copies
Acquire ()
Catalogue ()
Dispose ()
Issue ()
Return ()
Published item
Title
Publisher
Recorded item
Title
Medium
Book
Author
Edition
Publication date
ISBN
Magazine
Year
Issue
Film
Director
Date of release
Distributor
Computer program
Version
Platform

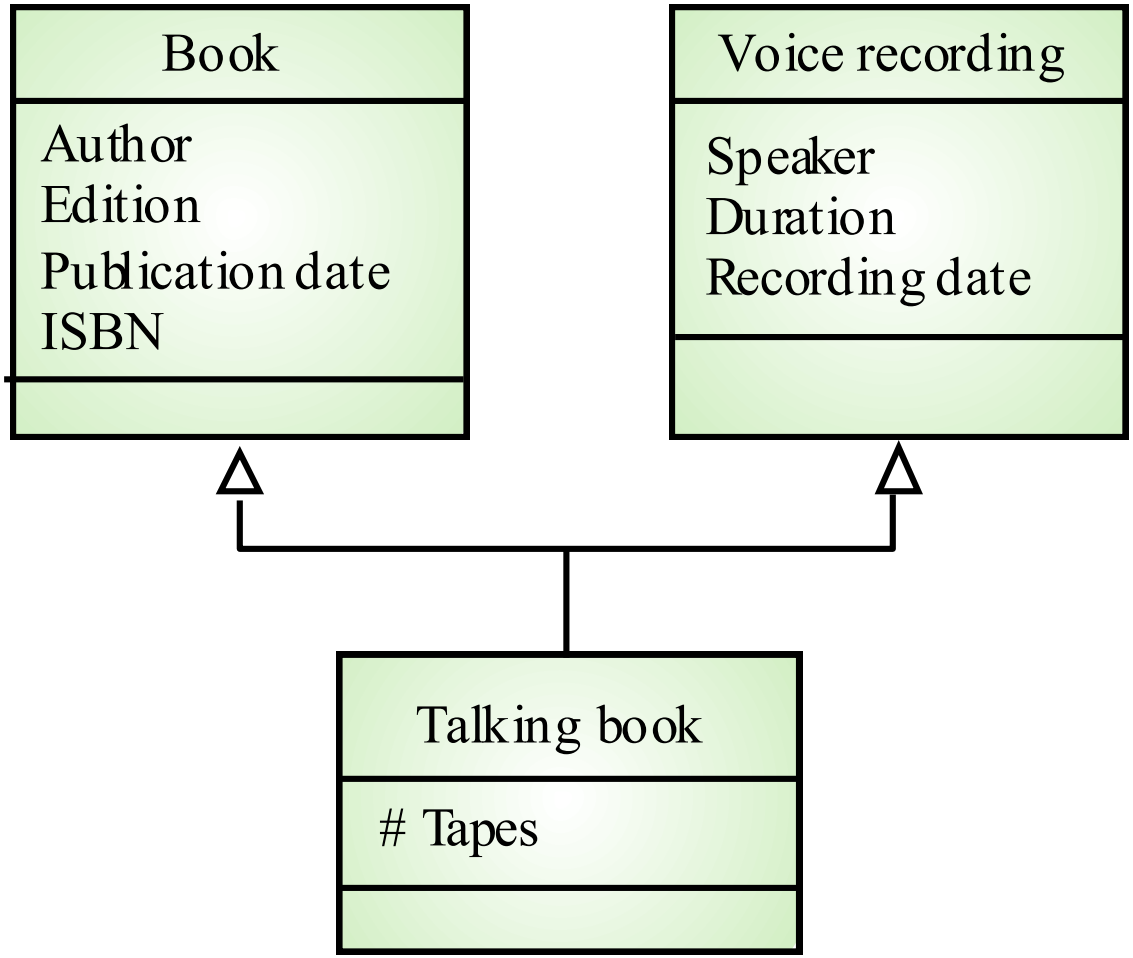
用户类层次


Library user
Name
Address
Phone
Registration #
Register ()
De-register ()
Reader
Affiliation
Borrower
Items on loan
Max. loans
Staff
Department
Department phone
Student
Major subject
Home address

多重继承

- 属性和服务不是从单一的父类继承，一个支持多重继承的系统允许对象类从几个超级类继承
- 会导致语义冲突，当不同的超级类中同样名称的属性和服务具有不同的含义时
- 会使得类层级变得更加复杂

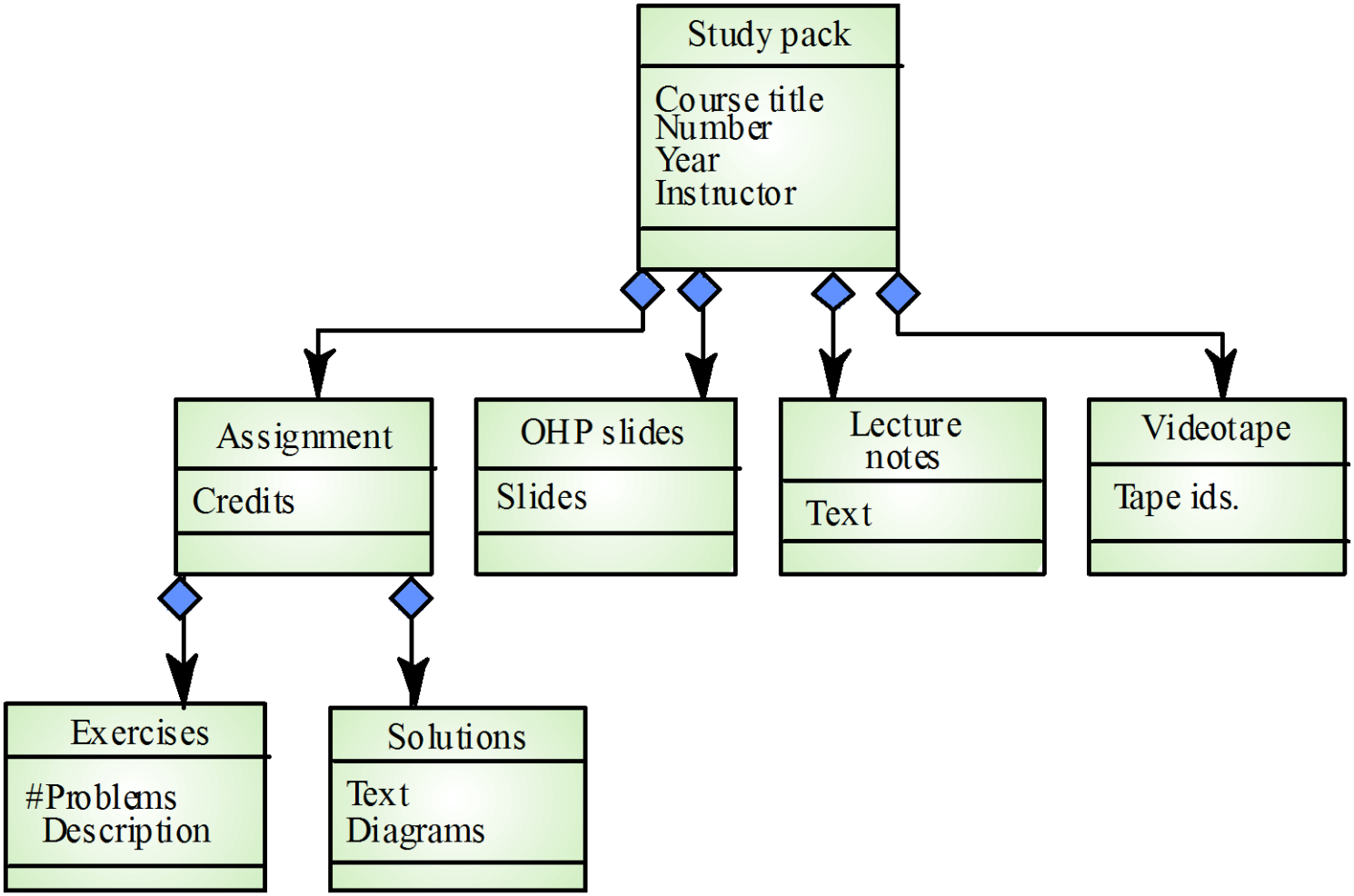
多重继承


Book
Author
Edition
Publication date
ISBN
Voice recording
Speaker
Duration
Recording date
Talking book
Tapes

对象聚合

- 聚合模型表示对象的组合类是如何由其它类组成的。
- 类似于语义数据模型中的关系的一部分

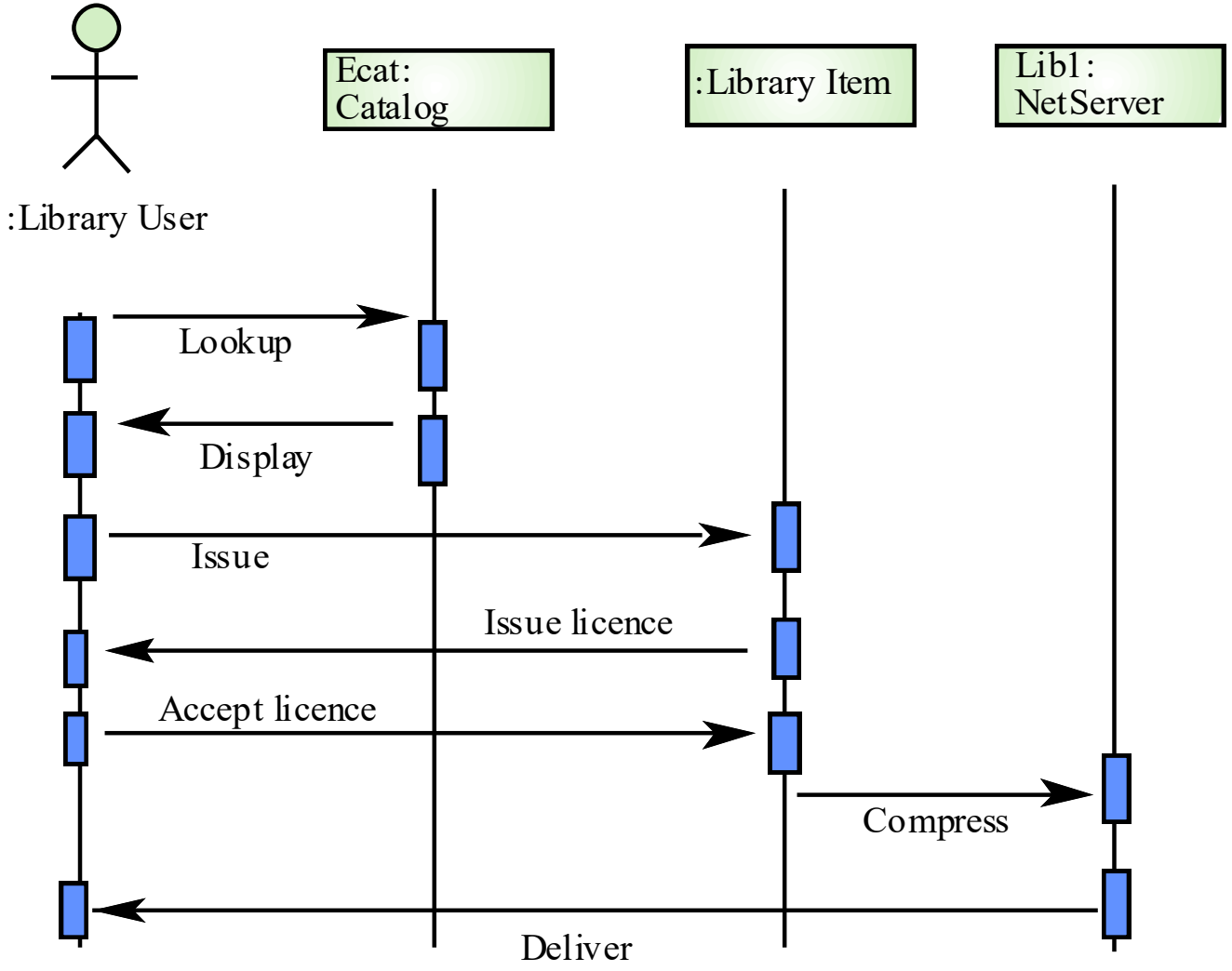
对象聚合


Study pack
Course title
Number
Year
Instructor
Assignment
Credits
OHP slides
Slides
Lecture notes
Text
Videotape
Tape ids.
Exercises
#Problems
Description
Solutions
Text
Diagrams

对象行为建模

- 行为模型表示了对象之间的交互作用，用以生成特定的系统行为，即用例
- 在UML中序列图用来对对象间的交互作用建模

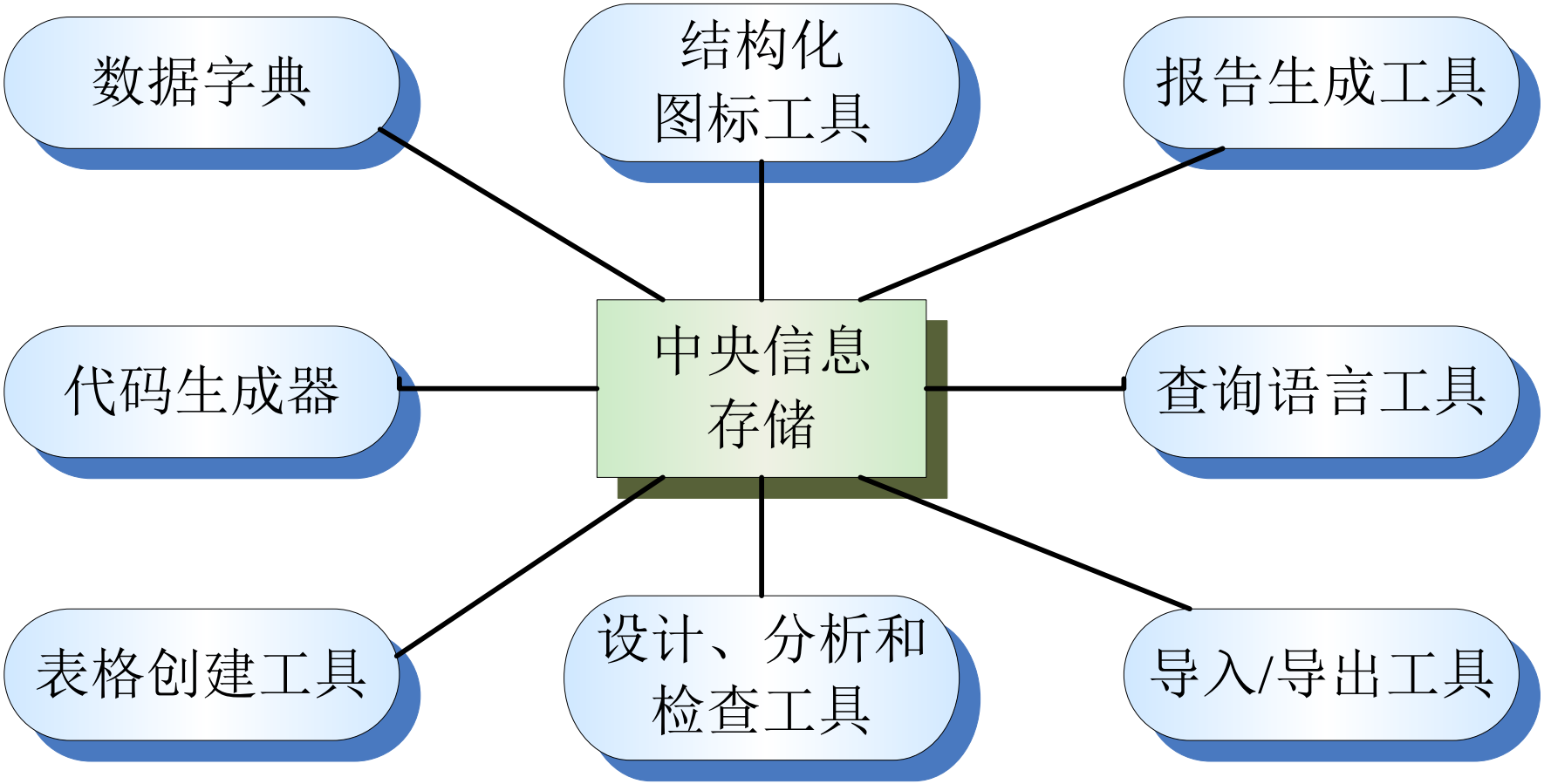
电子科目的发放


:Library User
Ecat:
Catalog
:Library Item
Lib1:
NetServer
Lookup
Display
Issue
Issue licence
Accept licence
Compress
Deliver

CASE工作平台

- 支持诸如分析、设计、测试等相关软件过程活动的一系列工具
- 分析和设计工作平台支持在需求工程和系统设计阶段的系统建模
- 这些工作平台可支持一个特定的设计方法或者为生成不同类型的系统模型提供支持

分析和设计平台


数据字典
结构化
图标工具
报告生成工具
代码生成器
中央信息
存储
查询语言工具
表格创建工具
设计、分析和
检查工具
导入/导出工具

分析工作平台组件

- 框图编辑器
- 模型分析和检查工具
- 知识库和关联查询语言
- 数据字典
- 报告定义和生成工具
- 报表定义工具
- 导入/导出转换器
- 代码生成工具

要点

- 模型是抽象的系统视图. 多种类型的模型提供了不同的系统信息
- 上下文模型表示一个系统在环境中的位置以及和其它系统、过程之间的关系
- 数据流模型可用来对系统的数据处理建模
- 状态机模型是系统对内部和外部事件的响应进行建模

要点

- 语义数据模型是对系统导入导出的数据逻辑结构进行描述
- 对象模型描述逻辑系统实体、实体的分类和集合
- CASE 工作平台支持系统模型的生成