

《大型软件设计》实验指导书

一、任意选择一个题目，3-4 个人为一组，选择组长作为项目负责人

1. 社区资源管理系统
2. 网上招聘系统
3. 仓库管理系统
4. 人事管理信息系统
5. 题库管理系统
6. 办公用品管理系统
7. 车辆管理系统
8. 仿 QQ 聊天小系统
9. 网上人才招聘系统
10. 俄罗斯方块游戏
11. 程序设计语言计算机辅助教学系统
12. 网上排课专家系统
13. 科研档案管理系统

二、建立问题领域的需求模型（3 周）

目标：用例图、类图、顺序图。

三、建立分析模型（4 周）

目标：建立用例图、类图（控制类、实体类、界面类）、顺序图。

四、建立设计模型（4 周）

目标：细化类图、顺序图、状态图、进行界面设计、核心算法设计

五、编码测试（4 周）

目标：编写程序，设计单元测试用例，测试报告。

注意：每一个组员在每一个阶段都要完成一部分任务，组长起到协调管理整个项目，每一个组提交一份实验报告。

第一章、餐馆系统业务建模



1.1 非正式需求（一）

- 目的：改进为顾客分配餐桌，支持餐馆的日常经营活动
- 餐馆的业务
 - 预约餐桌（登记哪些信息）
 - 取消预约
 - 临时分配餐桌



1.1 非正式需求（二）

- 手工预约的缺点
 - 慢
 - 涂涂改改，难以理解
 - 理解不好，影响业务
 - 没有备份



武汉大学

WUHAN UNIVERSITY

1.2 用例建模

- 用例视图：UML中起支配作用的视图，展示系统功能的结构化视图
- 角色(Actor)
- 用例(Use Case)



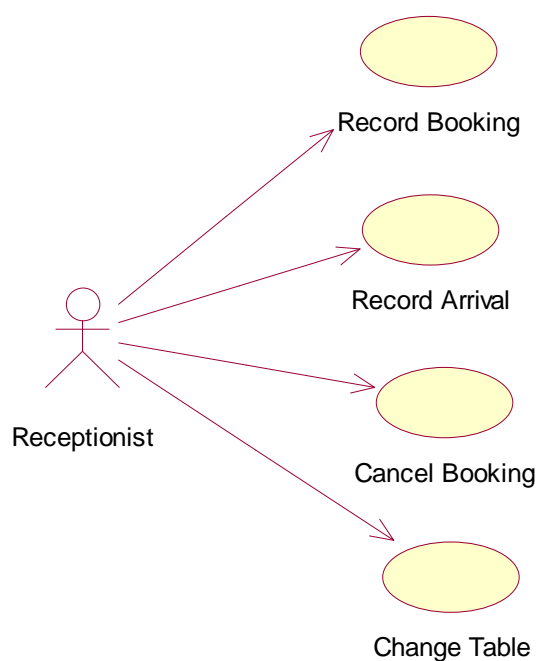
武汉大学

WUHAN UNIVERSITY

1.2.1 用例（一）

- 记录一个新的预约信息（记录预约）
- 取消一个预约（取消预约）
- 记录一位顾客的到来
- 将一位顾客从一张餐桌移到另外一张餐桌

1.2.1 用例（二）



1.2.2 角色

- 用例分为两组：
 - 与维护提前预约信息相关的用例
 - 在餐馆营业时执行，临时分配餐桌，调换餐桌
- 角色分为两组
 - 接待员
 - 领班
- 一个人可以担任多个角色



武汉大学

WUHAN UNIVERSITY

1.3 描述用例

- 用例描述了系统和它用户之间在一定层次上的完整的交互，并得到一个期望目标；
- 用例的不同的实例会有许多不相同的细节；
- 用例的实例会发生差错；
- 事件路径：基本路径事件、可选和例外；
- 用户界面原型。



武汉大学

WUHAN UNIVERSITY

1.3.1 用例：记录预约（一）

- 基本事件路径
 - 接待员输入要预约的日期；
 - 系统显示该日的预约；
 - 有一张合适的餐桌可以使用接待员，接待员输入顾客的姓名和电话号码、预约的时间、用餐人数和餐桌号
 - 系统记录并显示预约。



用例：记录预约（二）

- 记录预约—没有可用的餐桌（可选）
 - 接待员输入要求预约的日期
 - 系统显示该日的预约
 - 没有合适的餐桌可以使用，用例终止



用例：记录预约（三）

- 记录预约—餐桌过小（例外）
 - 接待员输入要求预约的日期
 - 系统显示该日的预约
 - 接待员输入顾客的姓名和电话，预约的时间，用餐人数和餐桌号
 - 预约用餐人数大于餐桌能容纳的人数，系统发出警告信息询问用户是否预约
 - 如果回答否，终止预约
 - 如果回答是，确定预约，并附有警告标志。



1.3.2 用户界面原型

- 用例重点定义系统和用户之间交互的总体结构，包含过多的用户界面细节会使之不清晰。



1.4 组织用例模型

- 记录到达：基本路径
 - 领班输入当前日期
 - 系统显示当天的预约
 - 领班确认一个预约已经到达
 - 系统对此记录并更新餐桌状态
- 记录到达：可选路径
 - 领班输入当前日期
 - 系统显示当天的预约
 - 领班输入到达时间，用餐人数和餐桌号，创建一个零时就餐
 - 系统记录并更新餐桌状态



武汉大学

WUHAN UNIVERSITY

1.4.1 用例：包含（一）

- 已描述的用例事件中，冗余
- 餐馆经理试图计算某天的服务员是否足够，看那天的预约
- 当前的模型并不能满足这一点。

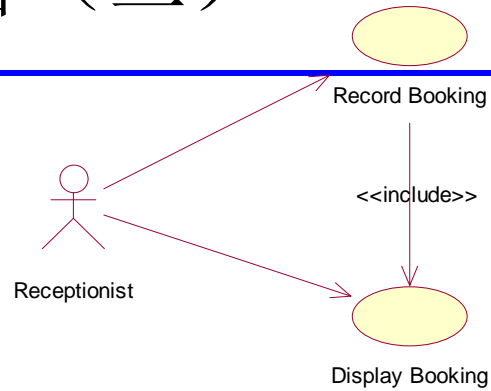


武汉大学

WUHAN UNIVERSITY

用例：包含（二）

- 显示预约：基本事件
 - 用户输入一个日期
 - 系统显示当前的预约
- 记录预约
 - 接待员执行“显示预约”用例
 - 接待员输入顾客姓名和电话号码、预订的时间、用餐人数以及预留的餐桌
 - 系统记录和显示新预约

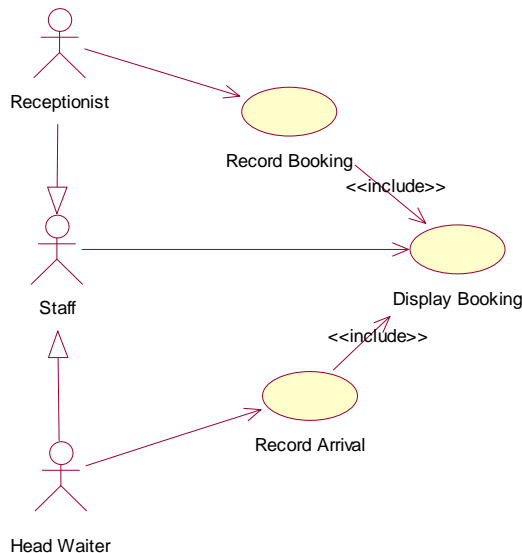


武汉大学

WUHAN UNIVERSITY

1.4.2 参与者泛化

- 任何人可以查询已有的预约



武汉大学

WUHAN UNIVERSITY

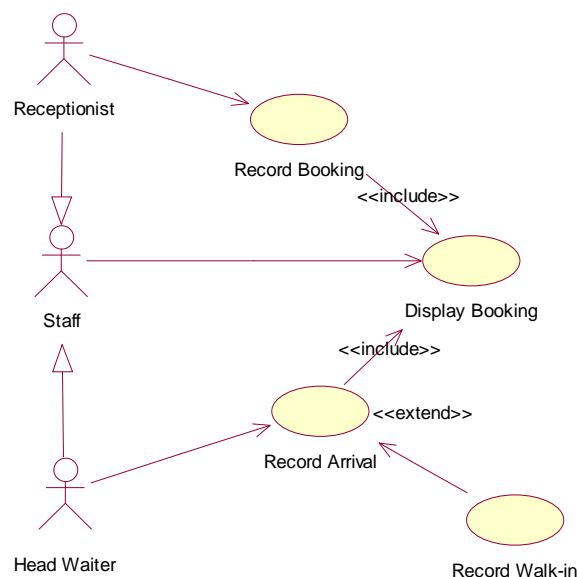
1.4.3 用例扩展

- 记录到达包含记录已预约的和临时顾客
- 记录临时分配：基本事件
 - 领班执行“显示预约”用例
 - 领班输入时间、用餐人数和分配给顾客的餐桌
 - 系统记录并显示预约
- 记录临时分配和预约有很大的重合，包含关系不合适。



武汉大学

WUHAN UNIVERSITY



武汉大学

WUHAN UNIVERSITY

1.5 完成用例模型

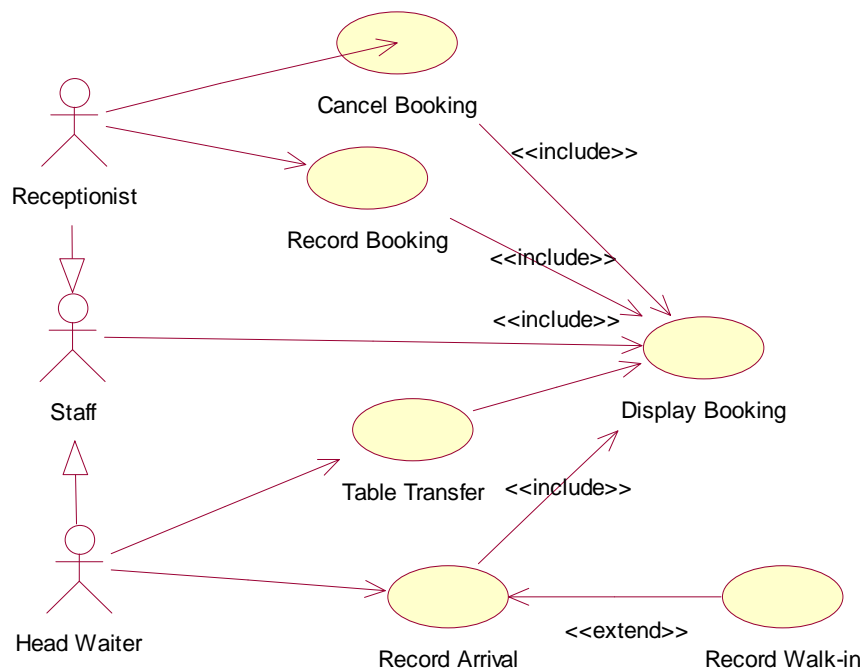
- 取消预约
 - 接待员选择要求的预约
 - 接待员取消预约
 - 系统询问接待员确认取消
 - 接待员回答是，系统记录取消并更新显示
- 调换餐桌
 - 领班选择需要的预约
 - 领班改变该预约的餐桌分配
 - 系统记录并更新显示



武汉大学

WUHAN UNIVERSITY

完整的用例模型



武汉大学

WUHAN UNIVERSITY

1.6 领域建模

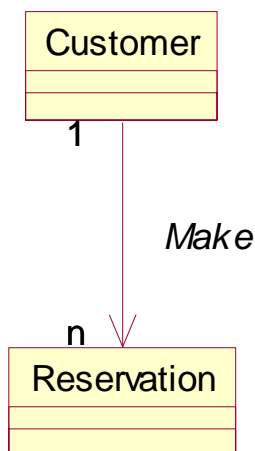
- 用例使系统开发人员和用户都容易理解，因此要用来自业务领域术语进行描述；
- 系统的文档化在用例描述中的业务概念；
- 方法：产生类图，描述最重要的业务概念和他们之间的关系
- 餐馆系统关键的业务需求：用户预约。



武汉大学

WUHAN UNIVERSITY

1.6 领域建模



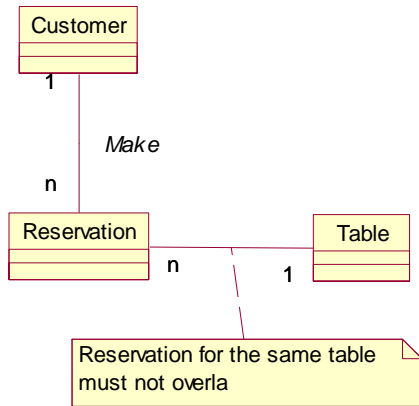
- Customer的属性
 - 姓名
 - 电话
 - 人数
 - 需要列出每个人的具体姓名吗？
- Reservation 属性
 - 桌号
 - 日期
 - 时间



武汉大学

WUHAN UNIVERSITY

1.6 领域建模



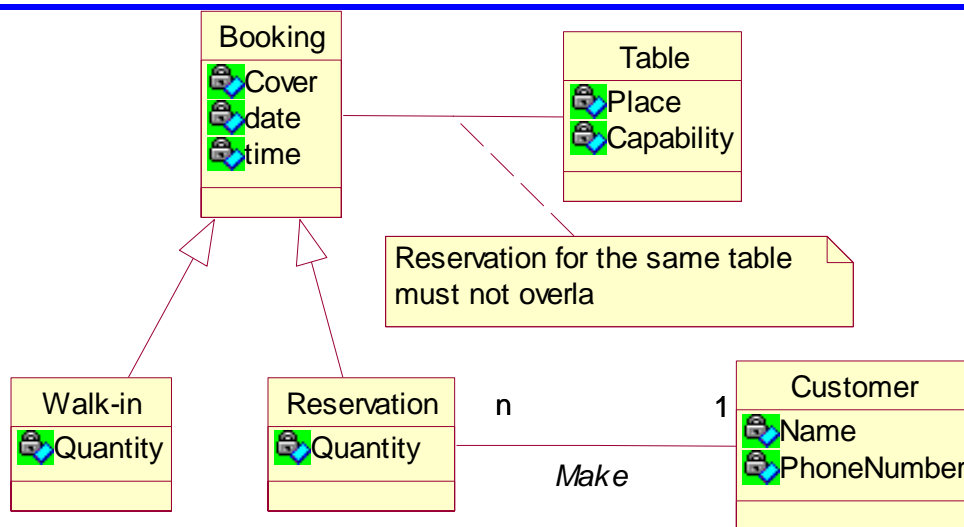
- 是否需要增加一个Table类
- 约束描述
- 没有反映临时就餐的信息



武汉大学

WUHAN UNIVERSITY

1.6 领域建模



还有什么可以增加的属性...



武汉大学

WUHAN UNIVERSITY

1.7 术语表

- 预约
- 用餐人数
- 顾客
- 用餐者
- 位子
- 预定
- 未预约



武汉大学

WUHAN UNIVERSITY

第二章 餐馆系统分析



武汉大学

WUHAN UNIVERSITY

2.1 分析的目的

- 分析的任务是构造一个模型，来说明这些交互的对象如何交互，从而完成用例中规定的行为（或者是实现用例的功能）
 - 用例描述用户和系统的交互
 - 领域模型描述重要的类
 - 系统内部究竟如何交互，
- UML交互图
 - 协作图
 - 顺序图



武汉大学

WUHAN UNIVERSITY

2.2 对象设计

- 为了生成交互图，必须在对象间分配所需的数据和处理（方法）
- 对象
 - 数据
 - 处理
- 内聚：组成一个合理的整体
 - 维护餐桌大小和顾客电话号码（×）
 - 维护所有关于顾客的信息（√）



武汉大学

WUHAN UNIVERSITY

2.3 软件架构

- 对象设计具有一些通用性的原则：模式
- 软件架构：如何划分子系统
- 层次架构：MVC
 - Model
 - View
 - Control



武汉大学

WUHAN UNIVERSITY

2.3 软件架构

- 类的类型
 - 边界类
 - 实体类
 - 控制类
- 软件架构
 - 显示层
 - 应用层
 - 存储层



武汉大学

WUHAN UNIVERSITY

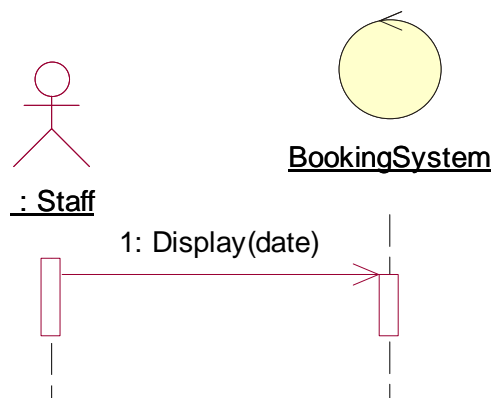
2.4 用例顺序图

- 系统消息：面向对象中，用户与系统打交道是通过发送消息，
- 谁接收消息
 - 领域模型中的类
 - 边界对象（边界对象在系统架构中属于表示层，需要根据消息分析应用层中对象的行为，不合适）
 - 一个用例中可能有許多消息，检查消息的正确性，协调系统产生的响应——控制对象。
 - **BookingSystem**控制类负责接收系统消息



武汉大学

WUHAN UNIVERSITY



如何检索



武汉大学

WUHAN UNIVERSITY

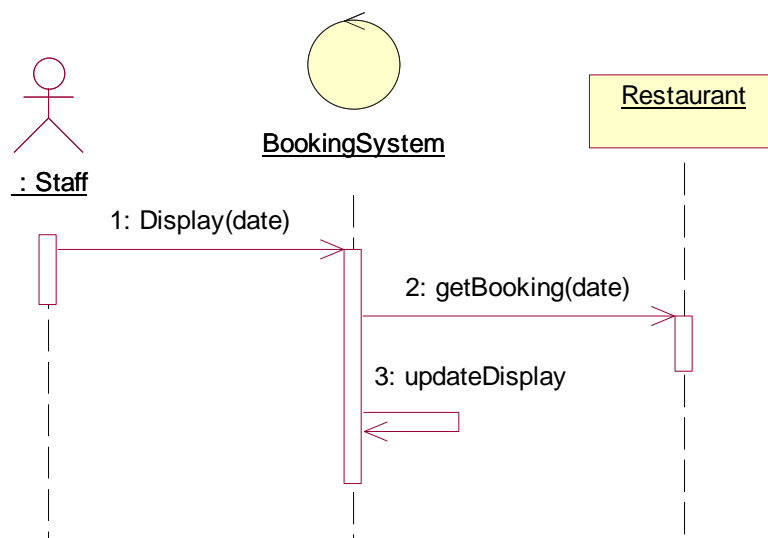
2.4.1 显示预约

- 预约信息存储在Booking对象中，它仅存储单个的预约信息，全部的预约信息如何维护
- 维护单个实体和实体集合的责任最好分给不同的对象
- 分给预约控制系统（**内聚？**）
- 建立一个新的类来维护预约集合：
Restaurant



武汉大学

WUHAN UNIVERSITY



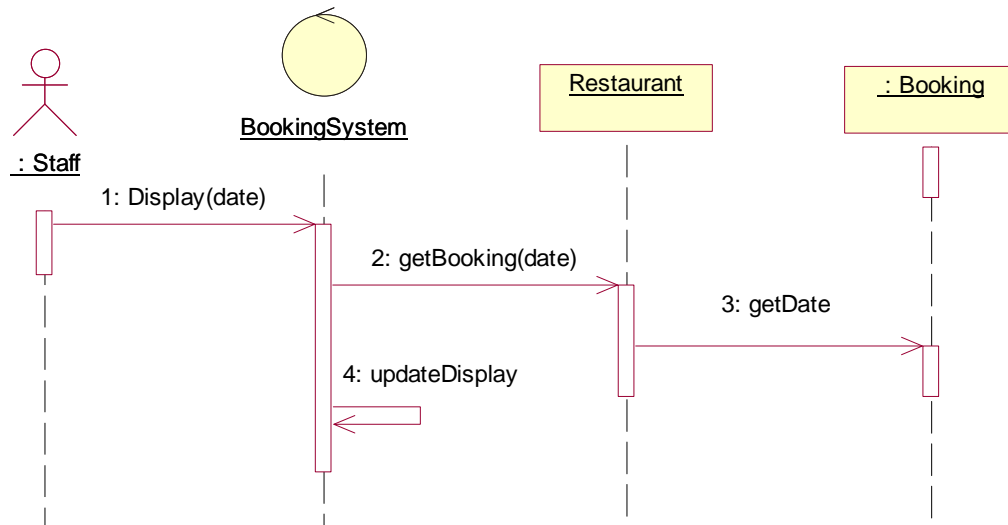
Restaurant对象如何识别返回的预约？



武汉大学

WUHAN UNIVERSITY

2.4.2 显示预约（细节）

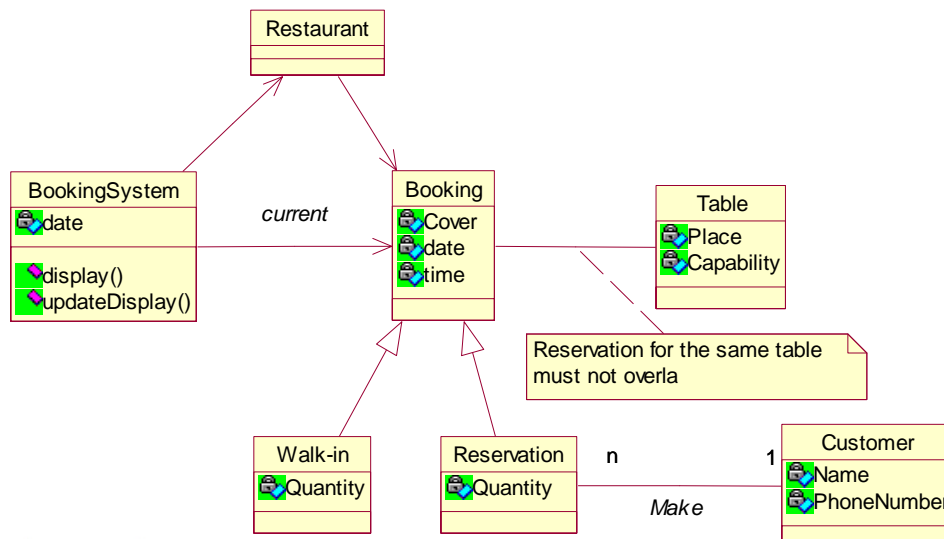


武汉大学

WUHAN UNIVERSITY

2.4.3 细化领域模型

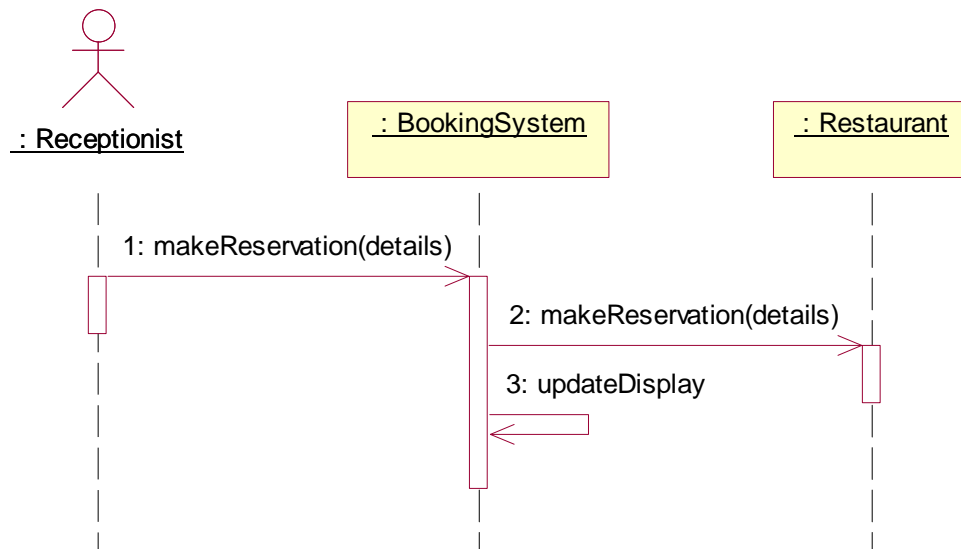
- 保存当前日期的预约信息（或当前屏幕上的预约信息）



武汉大学

WUHAN UNIVERSITY

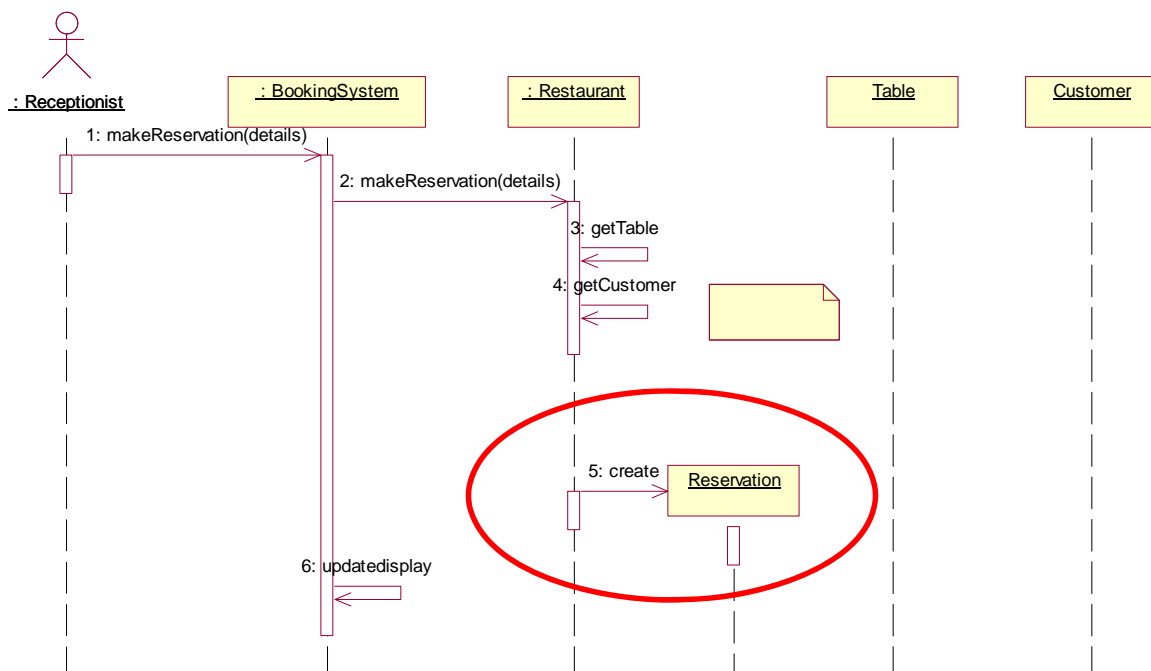
2.5 记录新预约



武汉大学

WUHAN UNIVERSITY

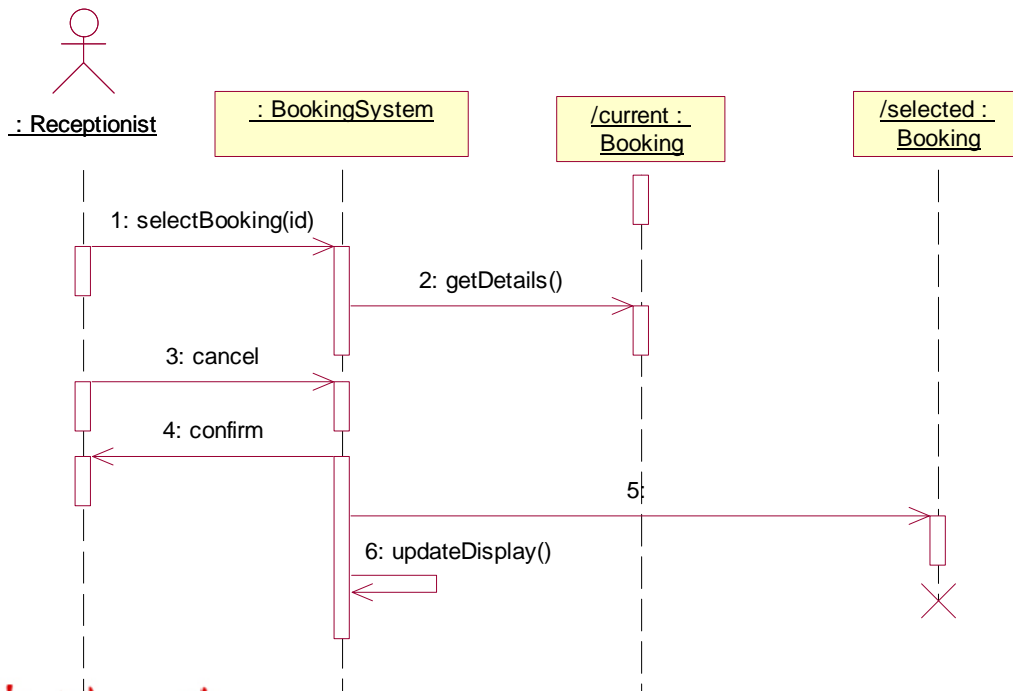
创建新对象



武汉大学

WUHAN UNIVERSITY

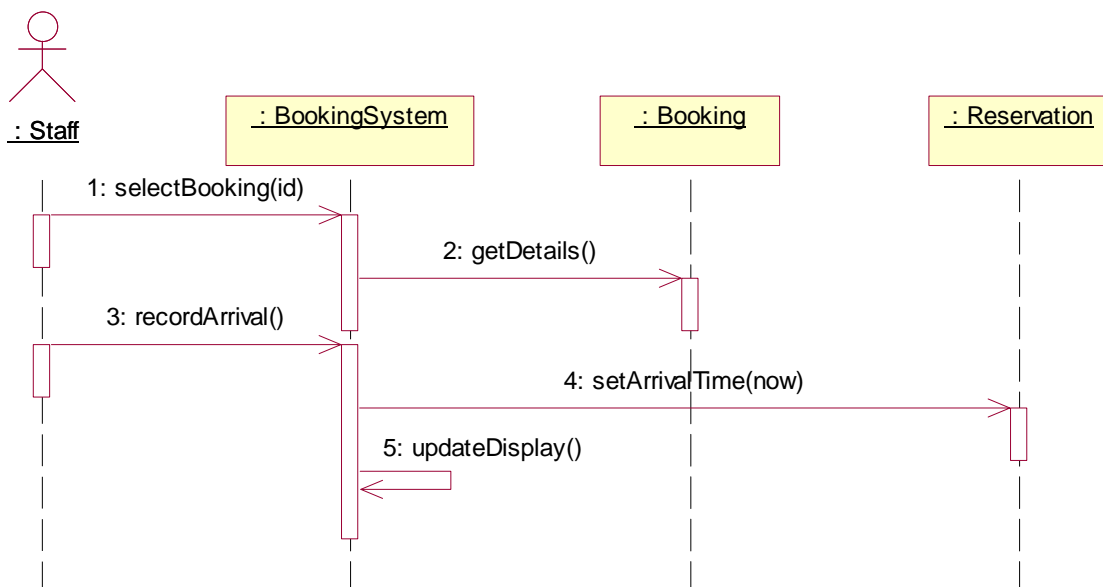
2.6 取消预约



武汉大学

WUHAN UNIVERSITY

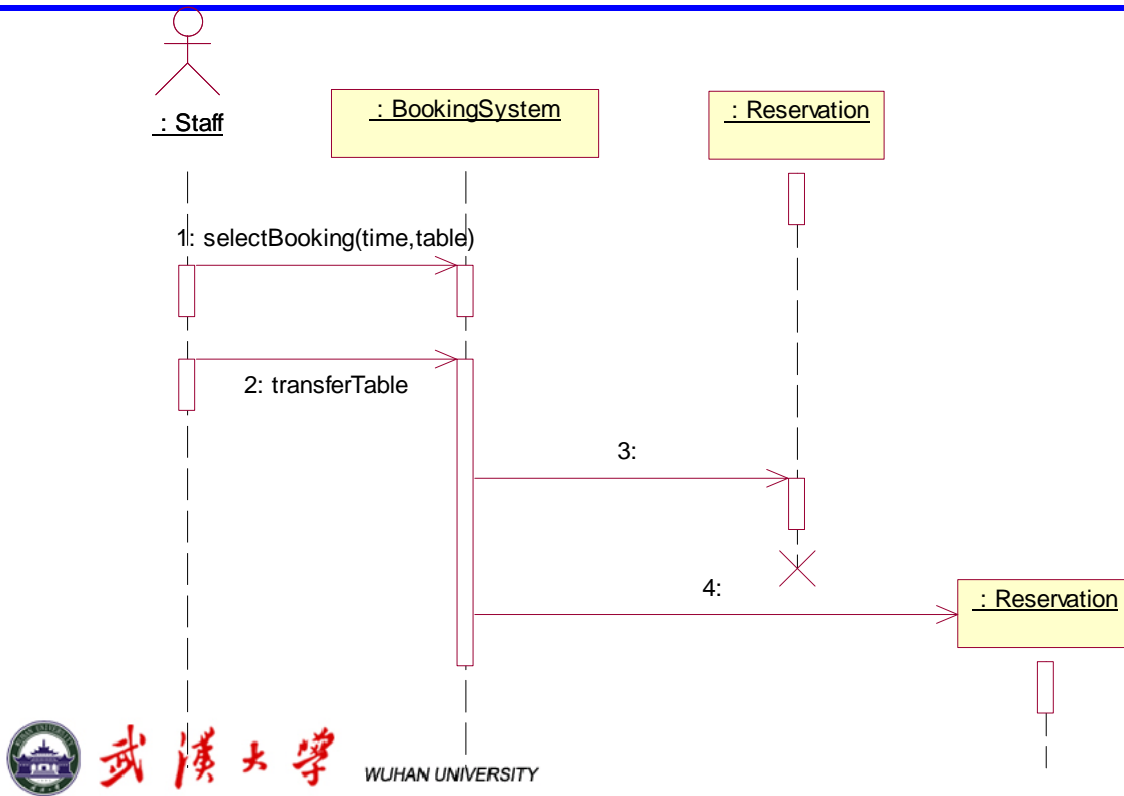
2.7 记录到达



武汉大学

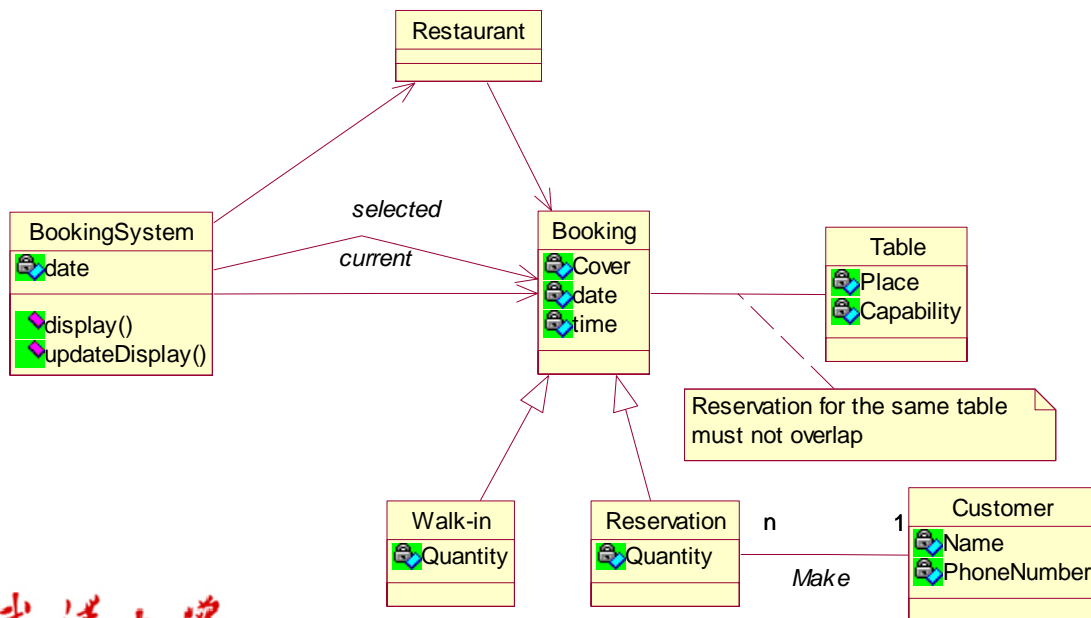
WUHAN UNIVERSITY

2.8 调换餐桌



2.8 完成分析模型

- 主要分析每个用例的基本事件路径



三、餐馆系统设计



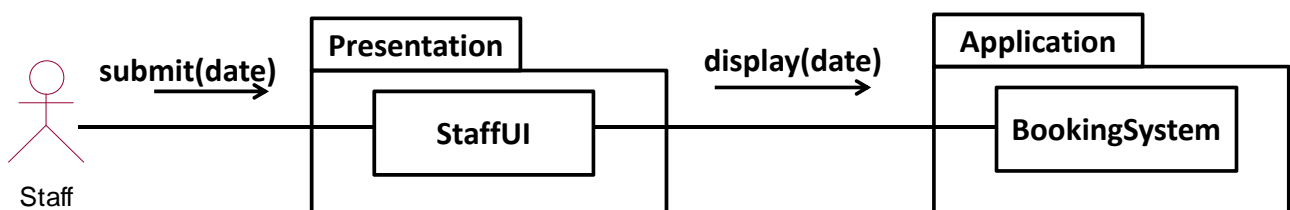
武汉大学

WUHAN UNIVERSITY

3.1 接受用户输入

- 表示层对象输入并转发给控制对象
- 显示预约
 - 选择菜单
 - 弹出对话框
 - 选择时间
 - 用户选择OK

省略细节



武汉大学

WUHAN UNIVERSITY

3.2 产生输出

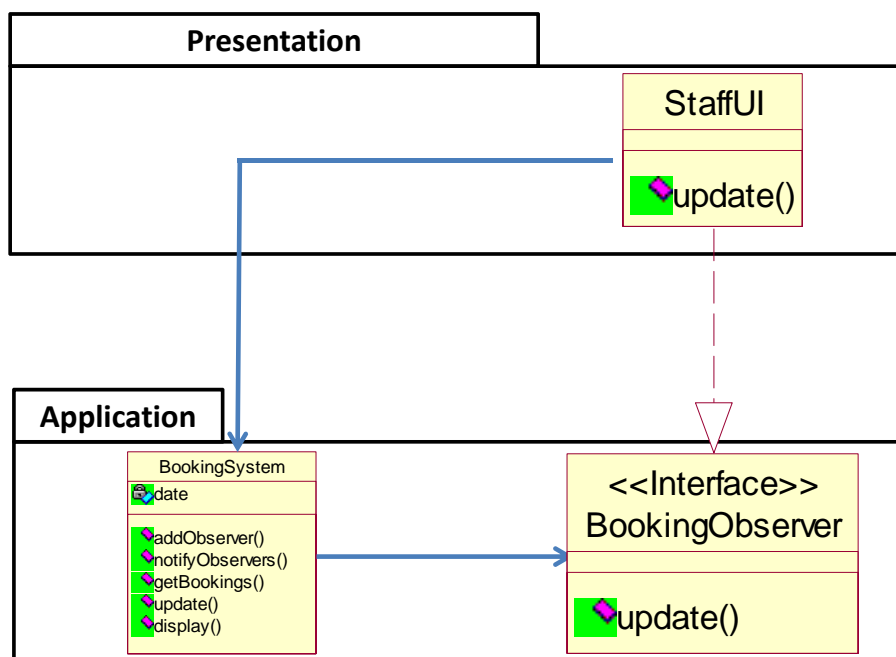
- StaffUI
 - 边界类，接受来自用户的消息，并将消息转发给控制器类
 - 视图类数据，将应用数据或模型呈现给用户
 - 数据发生改变，视图改变
 - 轮询（缺点）
- 观察者模式
 - 一个对象的变化需要改变其它对象，并且你不知道有多少对象需要改变
 - 一个对象应该能够通知其它对象，而无需设想那些对象是谁



武汉大学

WUHAN UNIVERSITY

应用设计模式



武汉大学

WUHAN UNIVERSITY

3.3 持久数据库设计

- 某些数据需要长久保存
- 方式
 - 磁盘文件
 - 数据库
- 对类标记persistent
- 持久类和关联如何保存
 - 持久类对应于数据库中的表
 - 关联对应于外键

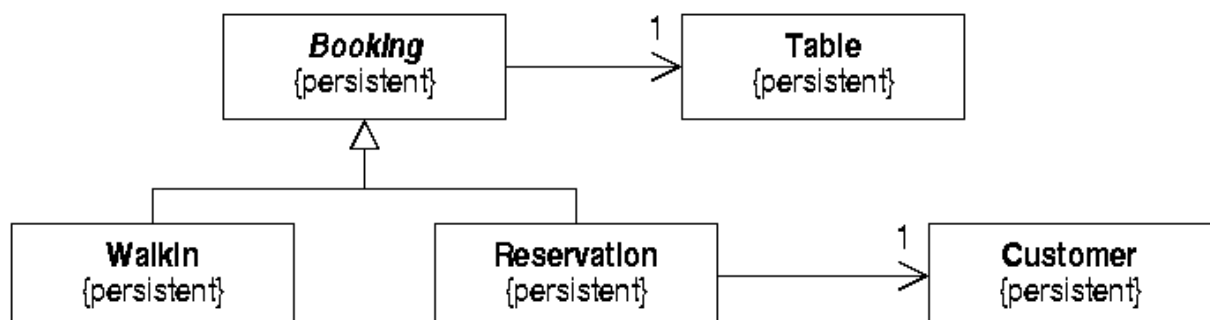


武汉大学

WUHAN UNIVERSITY

3.3 持久数据库设计

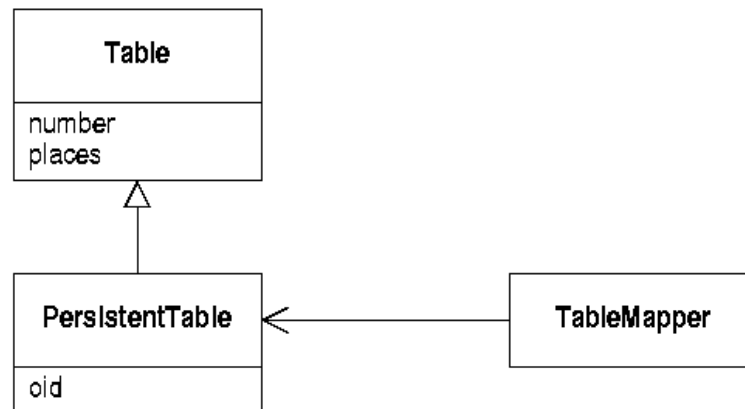
- 表设计
 - Table(oid,number,places)
 - Customer(oid,name,phoneNumber)
 - Walkin(oid,covers,date,time,table_id)
 - Reservation(oid,covers,date,time,table_id,customer_id)



武汉大学

WUHAN UNIVERSITY

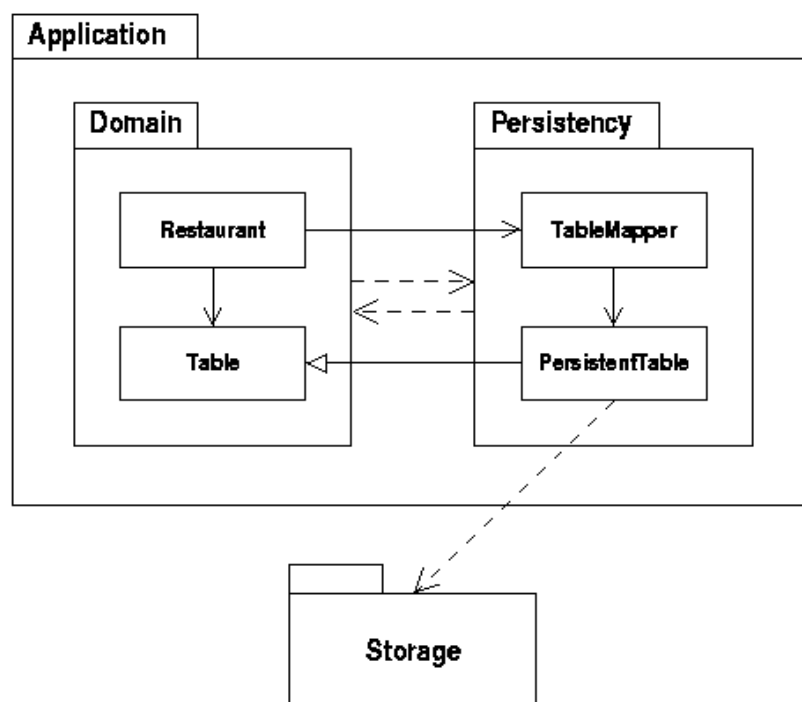
- 保存和装载持久对象
 - 数据库和内存之间移动持久数据
 - 对每个持久类，定义相关联的映射器(mapper)



武汉大学

WUHAN UNIVERSITY

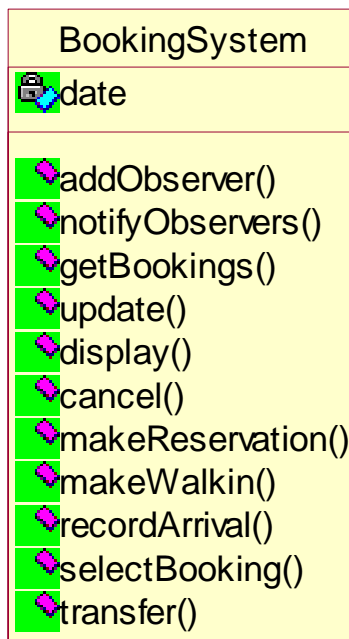
3.4 设计模型



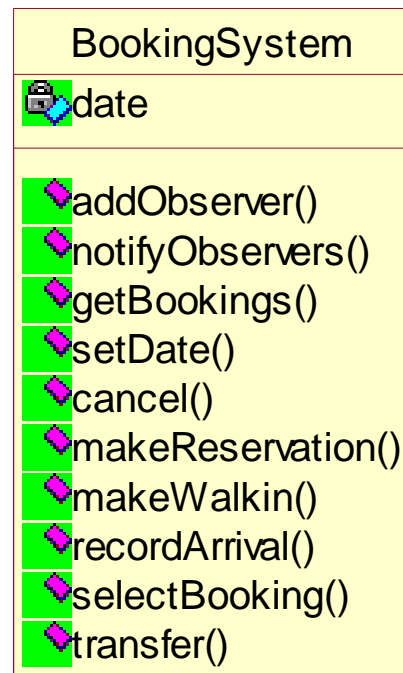
武汉大学

WUHAN UNIVERSITY

3.5 详细的类设计



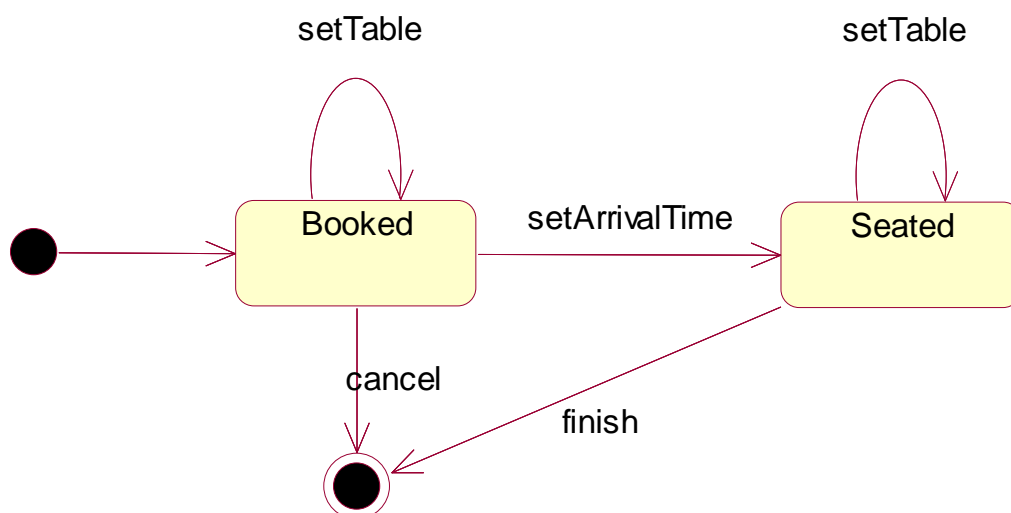
(1) 显示这个操作
应该在表示层,
所以改为 **setDate**;
(2) **update** 改为
notifyObserver



武汉大学

WUHAN UNIVERSITY

3.6 动态行为建模



武汉大学

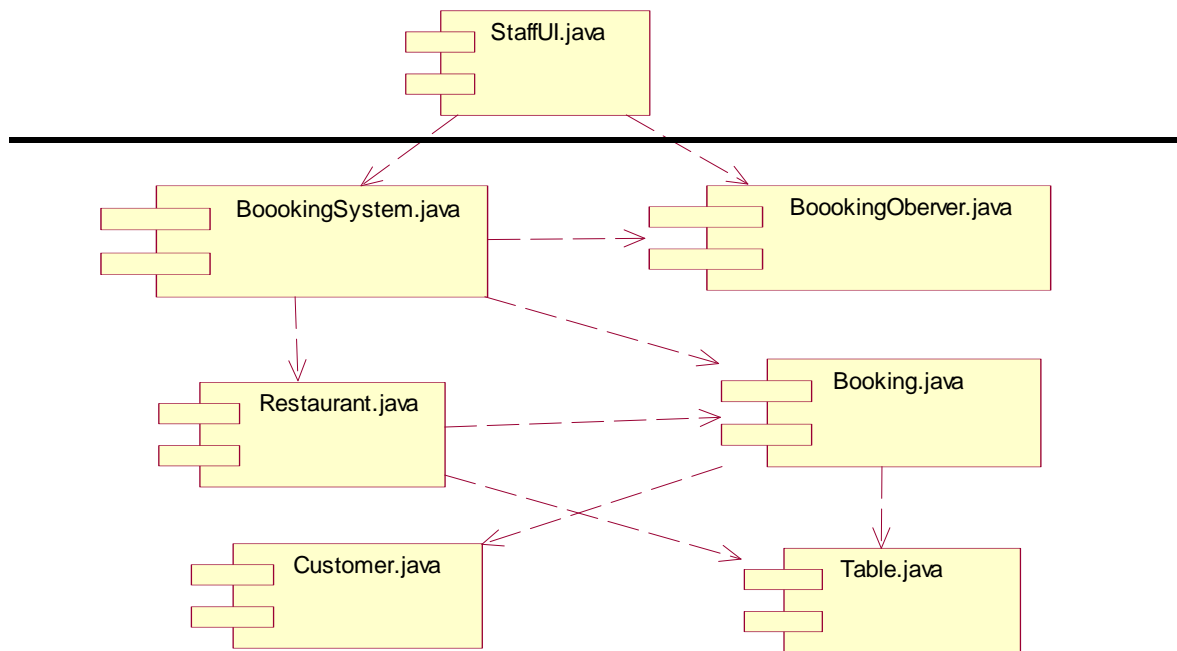
WUHAN UNIVERSITY

四、餐馆系统的实现

4.1 构件

- 构件表示系统部件的物理实体，存在很多不同类型额构件，包括源代码文件、可执行文件、库以及数据库表
- 构件图

构件图



武汉大学

WUHAN UNIVERSITY

4.2 类的实现

- 类：代码，根据不同的语言编写
- 泛化：继承
- 类的重数

系统只能有一个
预约控制类

```
public class BookingSystem
{
    private static BookingSyetem uniqueInstance;
    public static BookingSystem getInstance()
    {
        if(uniqueInstance==null)
            uniqueInstance=new BookingSystem();

        return uniqueInstance;
    }
    ...
}
```



武汉大学

WUHAN UNIVERSITY

-
- 关联的实现
 - 单向关联（指针）
 - 双向关联
 - 聚集
 - 关联的重数约束：链表