



# 部署图

# 内容

- 概述
- 基本概念
- 建模方法



# 内容

## ➤ 概述

- 基本概念
- 建模方法



# 部署图

- 部署图又称配置图，实施图
- 部署图描述系统硬件的拓扑结构和软件在硬件上的配置。
- 部署图描述系统中包含的计算机和其它硬件设备、这些硬件之间的相互连接和硬件上运行的软件。

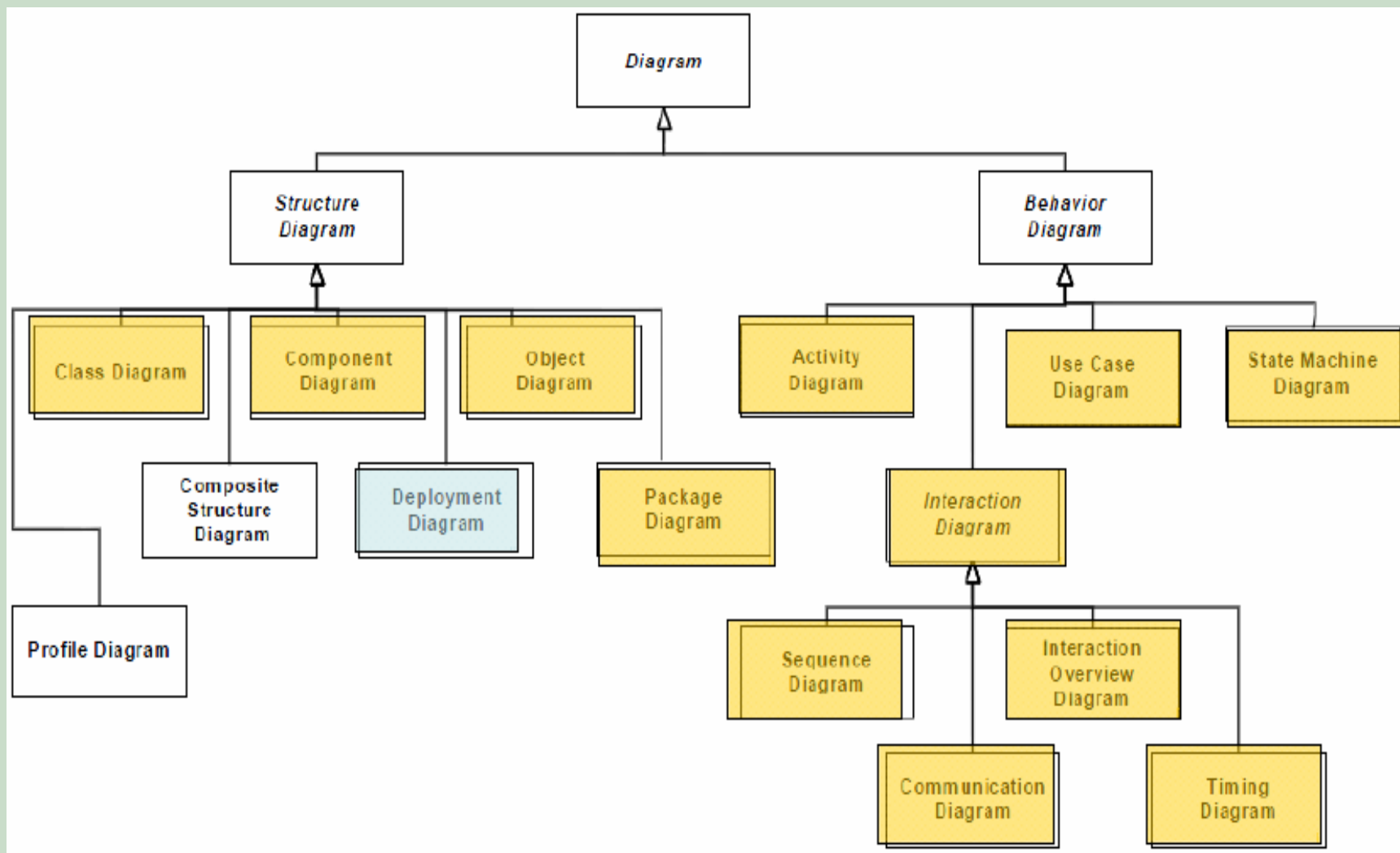


# 部署图

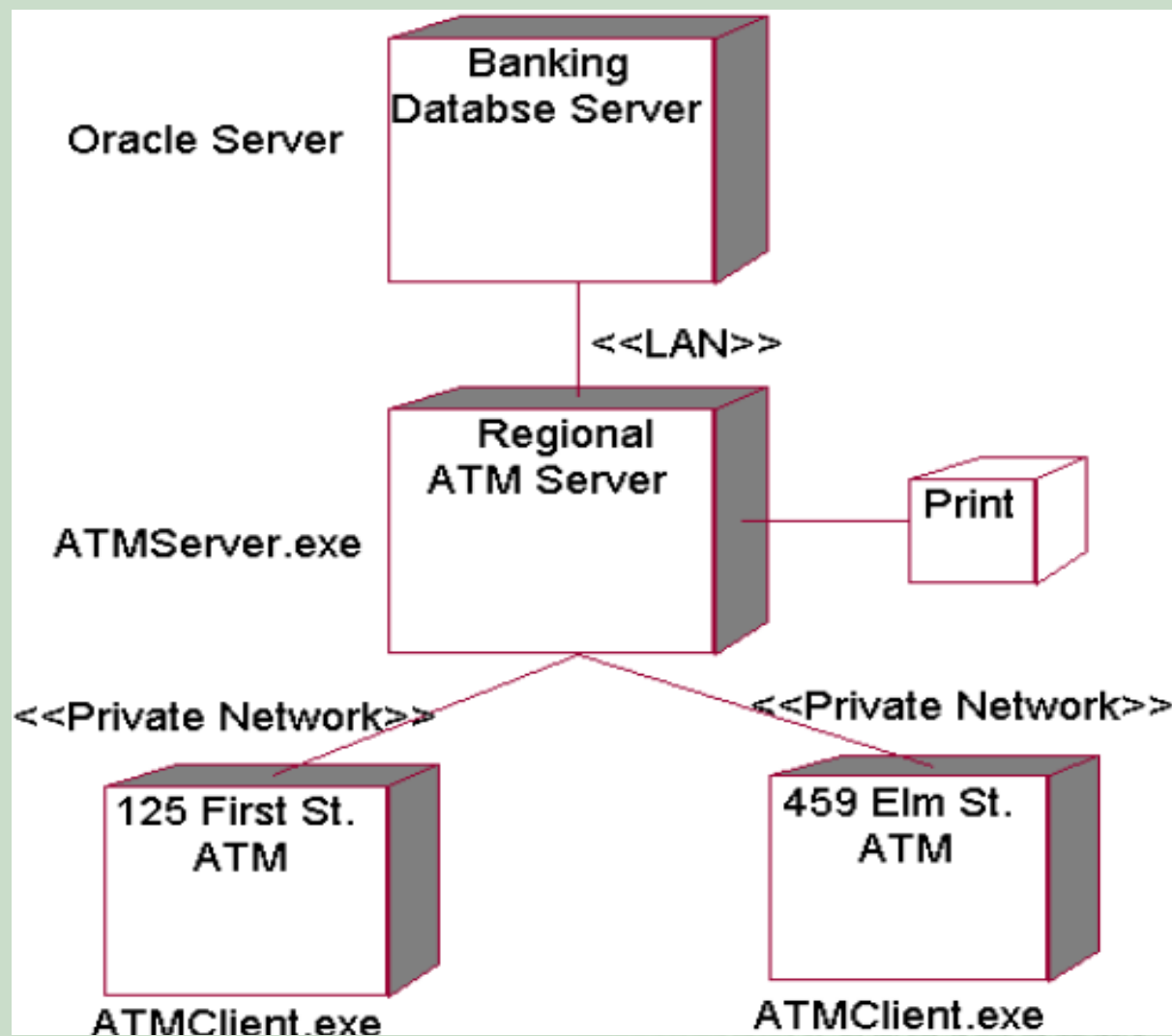
- 部署图将软件构架映射到硬件构架。
- 部署图在描述较复杂系统的拓扑结构时很有用，常常用于描述分布式系统。



# 部署图



# 部署图示例



# 内容

- 概述
- 基本概念
- 建模方法





# 部署图

- 部署图的基本建模元素：
  - 节点（Node）
  - 连接（Connection）
- 部署图可以显示硬件节点的拓扑结构和通信路径，以及节点上运行的软件。



# 节点（Node）

- 节点代表一个物理硬件，如一台**Unix**主机、一个**PC**终端、一台打印机、一个传感器等
- 节点中可以包含所配置的软件
- 节点有两种类型：处理机和设备



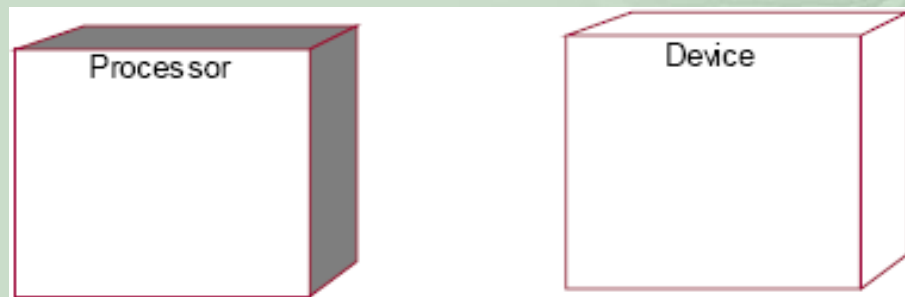
# 处理机和设备

## ➤ 处理机

- 能够执行软件，具有计算能力的节点
- 如服务器、工作站.....

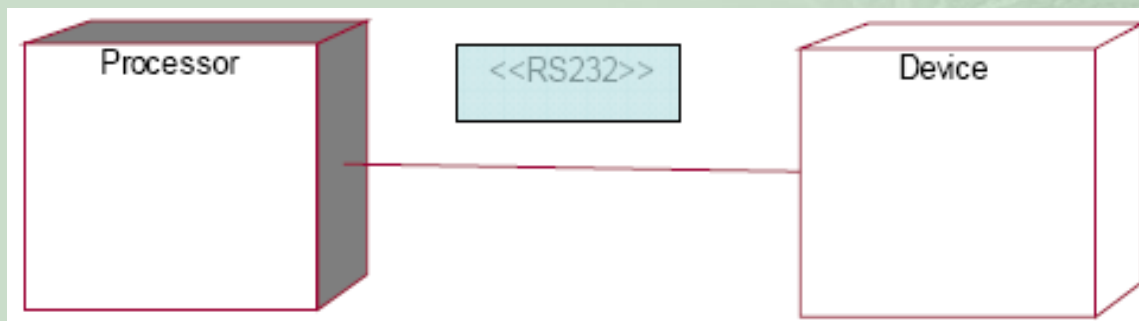
## ➤ 设备

- 没有计算能力的节点，通常通过其接口为外部提供服务
- 打印机，IC卡读写器.....

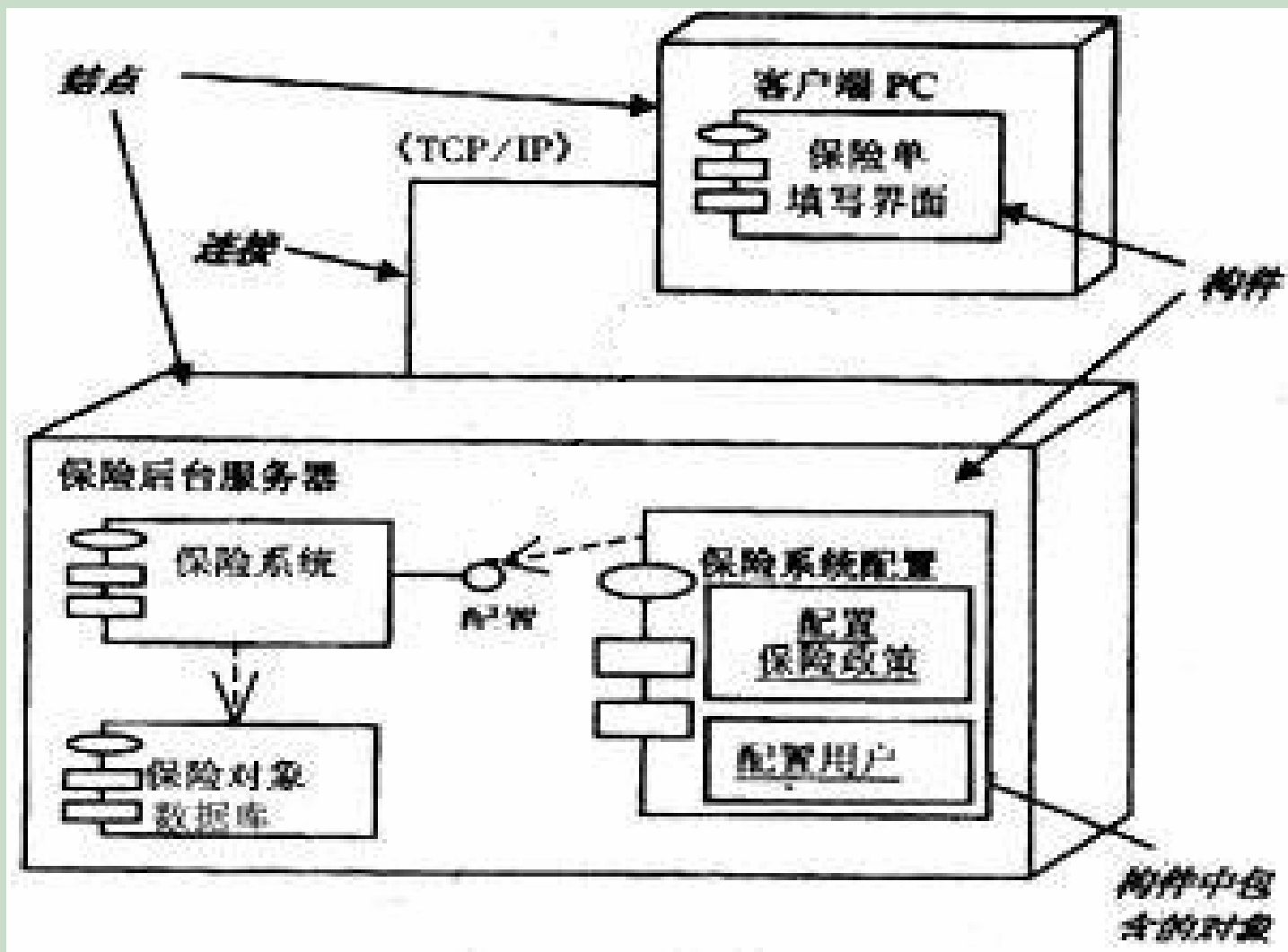


# 连接

- 连接是两个硬件节点之间的关联关系，表示节点之间的通信路径。
- 连接用一条实线表示。连接硬件通常关心是如何连接的，因此一般不使用名称而使用版型来描述，如《Ethernet》、《TCP》或《HTTP》等



# 部署图



# 节点与构件的区别

- 构件往往表示一个软件部件，节点往往表示一个硬件部件
- 构件是参与系统执行的事物，节点是执行构件的事物
- 构件表示逻辑元素的物理打包，节点表示构件的物理部署



# 部署图的应用

- 在设计阶段
  - 关注节点以及节点之间的连接
- 在实现阶段
  - 关注如何将物理构件分配给节点



# 内容

- 概述
- 基本概念
- 建模方法





# 部署图的建模

- 在实际应用中，并不是所有的软件开发项目都需要绘制部署图。
- 如果系统需要使用操作系统之外的设备或系统中的设备分布在多个处理器上，这时就需要绘制部署图，以帮助开发人员理解系统中软硬件之间的映射关系。



# 建模方法

- 对系统中的节点建模
- 对节点之间的关联关系建模
- 对配置在节点上的构件建模
- 对构件间的依赖关系建模
- 对建模结果进行细化



# 部署图建模风格

- 使用描述性术语为节点命名
  - 例：Client, Application Server, Database Server, Mainframe
- 在节点中只建模关键的软件构件
  - 不要把节点上的所有软件构件都画出来
- 为节点使用可视化版型
- 使用版型来表明通信协议
- 部署图是UML中最版型化的部分

