

# 软件体系结构

---

## 第一讲 软件体系结构概述

### 一、什么是软件体系结构 P1

- 1、软件体系结构是软件工程的一门新兴学科
- 2、在系统科学中，系统的定义
  - 系统的特性
    - 集合性、相关性、结构性、整体性、功能性、环境适应性在
  - 系统的体系结构
- 3、软件体系结构的定义
  - （网上资料的定义）软件体系结构为软件系统提供了一个结构、行为和属性的高级抽象，由构成系统的元素的描述、这些元素的相互作用、指导元素集成的模式以及这些模式的约束组成
  - 软件体系结构是具有一定形式的结构化元素，即构件的集合，包括处理构件、数据构件和连接构件。
- 4、软件设计的层次性
  - 结构级、代码级、执行级
- 5、体系结构的类别
  - 概略型、需求性、设计型
  - 软件系统结构的主要类别：设计型
- 6、软件体系结构的重要性
  - 规划阶段、需求阶段、设计阶段、实施阶段、测评阶段、维护阶段

### 二、模块及其设计 P2

- 1、什么是模块
  - 定义
    - 模块的重要性
      - 解决复杂问题的一种有效方法
      - 集体分工协作的前提
      - 产品维护的保障
- 2、模块的内聚性（模块内）
  - 含义：模块内聚性是指一个模块内相互作用的程度
  - 内聚性的层次（7层）
    - 偶然内聚性、逻辑内聚性、暂时内聚性、过程内聚性、通信内聚性（不好）
    - 信息内聚性、功能内聚性（好）
- 3、模块耦合（模块间） P4
  - 含义：模块耦合是指模块间的相互作用程度
  - 耦合的级别（5级）
    - 内容耦合、共用耦合、控制耦合（一般与逻辑内聚性的模块有关）、特征耦合、数据耦合
  - 一个好的模块设计应该使模块具有 高内聚性 和 低耦合性
- 4、重用 P5

- 重用的两种类型
  - 偶然重用、计划重用
- 构件重用：检索和提取构建，理解和评价构件，修改构件，构件组装
- 重用的障碍
  - 自负、经济利益、检索、代价、版权

### 三、软件工程 P6

- 1、软件危机
  - 软件成本日益增长、开发进度难控制、软件质量差、软件维护困难
  - 软件危机
    - 软件萧条->整体质量低
    - 软件危机->软件高度复性
  - 软件的本质
    - 复杂性、一致性、易变性、不可见性
- 2、软件工程的复杂性（软件的根本特性）
  - 软件在执行时处于离散状态
  - 软件运行环境具有不可再现性
  - 硬件的复杂性
- 3、软件工程的内容
  - 软件工程的生命周期

### 四、软件体系结构的意义与目标 P7

- 1、软件体系结构的意义
- 2、软件体系结构的目标
  - 主要目标：一致视图、结构形式
  - 外向目标：系统需求
  - 内向目标：用户需求

### 五、软件体系结构的研究现状 P7

- 1、软件体系结构的发展
  - 程序抽象、软件工程、体系结构
  - （网上资料的发展）
    - "无体系结构" 设计阶段 (以汇编语言进行小规模应用程序开发为特征)
    - 萌芽阶段 (控制流图和数据流图)
    - 初期阶段 (UML)
    - 高级阶段 (高层抽象结构为中心)
- 2、软件体系结构的研究现状
  - 软件体系结构的研究内容：风格、设计模式、结构描述语言

## 第二讲 软件体系结构风格

## 一、什么是软件体系结构风格 P9

- 1、什么是结构风格
  - 组织系统可用的各种结构模式（系统组件+连接器+它们的组合约束条件）
- 2、体系结构风格的分类
  - 数据流风格（批处理序列；管道/过滤器）
  - 调用/返回风格（主程序/子程序；面向对象；层次结构）
  - 独立组件风格（进程通讯；事件系统）
  - 虚拟机风格（解释器；基于规则的系统）
  - 仓库风格（数据库系统；超文本系统；黑板系统）

## 二、常用结构风格 P9

- 1、管道/过滤器（P/F）
  - 组件：过滤器 连接器：管道
  - 过滤器的独立性、拓扑结构
  - **P/F的优点**
- 2、数据抽象与面向对象组织（OO） P10
  - 组件：对象 连接器：激活对象方法的消息
  - 对象间的相互作用是通过函数和过程调用实现
  - OO的优点
- 3、事件及隐含激活 P11
  - 组件：一些模块（过程和事件的集合）
  - 优点
- 4、层次系统
  - 层次系统最广泛的应用是分层通信协议
  - 优点
- 5、仓库 P12
  - 黑板系统
    - 组成：知识源、黑板数据结构、控制
    - 应用
- 6、解释器
  - 解释器广泛用于建立虚拟机
  - 组成

## 三、其他结构风格 P13

- 1、分布式处理（客户/服务器是主要方式）
- 2、主程序/子程序调用
- 3、确定域结构
- 4、状态变迁系统

## 四、案例分析 P13

### 第三讲 分布式体系结构分析

## 一、分布式处理结构风格概述 P16

- 1、分布式处理结构风格的系统组织
  - 组件、透明性、可靠性
- 2、分布式处理结构风格的优缺点
  - 优点：降低模块之间的耦合度；把项目拆分成若干个子项目,不同的团队负责不同的子项目,可以更好的 分配开发任务
  - 缺点：系统之间的交互要使用远程通信,接口开发增大工作量
- 3、构造分布式处理结构风格的主要方式
  - 层次结构、技术基础、RPC、组件技术、云计算

## 二、分布式系统 P16

- 1、分布式系统是计算机发展的必然产物
- 2、分布式系统的优点
  - 提供更高的性能/价格比、运行速度、应用领域的需要、可靠性、扩充性、适应性
- 3、分布式系统的不足
  - 适用于分布式系统的软件很少、网络饱和以及其他问题、保密的数据也变得容易获取

## 三、客户/服务器结构 (C/S) P17

- 1、C/S结构
- 2、C/S模型的软件结构的特点
  - 简化软件产品的设计、提高软件的可靠性、适合分布式计算环境
- 3、C/S模型的工作过程
- 4、数据共享的方式
- 5、C/S的连接（消息传递、过程（RPC）、对象引用）
- 6、C/S的服务器设计、实现关键
  - 服务器的调度任务和调度方式、线程技术、服务器缓冲技术

## 四、C/S案例 P18

# 第四讲 分布式体系结构基础技术

## 一、基于消息传递的通信 P22

- 1、分布式进程的通信
  - 发送原语（格式、参数、作用）
  - 接受原语（格式、参数、作用）
  - 阻塞原语（同步原语）
  - 非阻塞原语（异步原语）
    - 特点
    - 改进方法：带拷贝的非阻塞send原语、带中断的非阻塞send原语
  - 非缓冲原语（抛弃非期望、消息保留一段时间）
  - 缓冲原语（信箱）
  - 不可靠原语
  - 可靠原语

## 二、远程过程调用 (RPC) P24

- 1、什么是RPC
  - 基本思想
  - 过程调用
  - 参数传递
- 2、RPC的透明性 P25
  - 客户代理、服务器代理
  - RPC的工作步骤
- 3、参数传递 P26
  - 参数整理
- 4、动态联编 P27
  - 动态联编的灵活性
    - 均衡工作量、容错性、支持权限、花费系统时间、瓶颈
- 5、RPC表示错误的语义 P28
  - 客户找不到服务器 (client cannot locate the server))
  - 从服务器到客户的应答丢失 (lost reply)
    - 同一有效性
  - 服务器接收了请求后的崩溃 (server crash)
    - 重传机制
  - 客户发送请求后崩溃 (client crash)
    - 孤报 (orphan)
      - 消灭、“再生”、gettlereincarnation(合理再生)、期满
- 6、RPC的实现 P30
  - 面向连接的协议的优点
  - 确认机制: stop-and-wait protocol、blast protocol
  - 临界路径
  - 客户调用 (stub procedure)
- 7、RPC与消息传递通信的比较 P33

## 三、分布式同步算法 P34

- 1、逻辑时钟 (Lamport算法)

## 四、分布式互斥算法 P35

- 1、集中式算法
- 2、**RA算法**
- 3、**令牌环算法**
- 4、三种算法的比较

## 五、分布式系统的可靠性 P36

- 1、选择算法
  - Bull算法
  - 环算法
- 2、k-容错技术
- 3、表决算法

## 第五讲 层次结构技术分析

### 一、线程技术 P40

- 1、引入线程的目的
- 2、线程的概念
- 3、线程与进程的区别、联系
- 4、线程的分类
  - 用户级线程、系统级线程
- 5、线程的执行特性
  - 线程的生命期——动态性、线程同步
- 6、线程的应用

### 二、服务器缓冲技术 P41

- 1、无状态信息服务器
- 2、有状态信息服务器
- 3、二者比较与总结

### 三、N层结构的特性 P41

- 1、层次结构（解决复杂问题最常用的软件结构）设计
  - 层次；层次间的单向依赖关系；层次的隐藏性；分层的原则
- 2、软件系统的层次结构
  - 计算机硬件层 -> 软件的硬化层 -> 基础控制描述层 -> 资源和管理调度层 -> 系统结构模式层 -> 应用层
- 3、N层结构
  - C/S或B/S结构称为2层结构，我们把层次结构中3层或3层以上统称为N层结构

### 四、N层结构的实现 P42

- 1、采用N层结构的实施步骤
- 2、Window NT的层次结构

### 五、N层结构的优缺点 P43

### 六、一个用于构造分布式系统的层次结构设计 P43

- 1、表示层 —— 用户界面技术（划分UIC和UIPC）
- 2、业务逻辑层 —— 应用系统的核心（Business Component、Business Workflow、Business interface、Business Entities（BE）
- 3、数据逻辑访问层（DAL）
  - Helper的作用
- 4、数据层

## 第六讲 CORBA技术及应用实例

## 一、CORBA概述 P47

- 1、CORBA及其发展过程
  - 对象管理组 (OMG)
  - 对象模型 (Object Mode) 、引用模型 (Reference Model)
  - CORBA简化了C/S模式

## 二、CORBA特性 P48

- 1、OMG接口定义语言 (IDL)
- 2、语言映射
- 3、操作调用和调度软件
  - 调用请求的两种方法：静态调用和调度、动态调用和调度
  - 调用请求的过程
  - 对象引用(Object Reference)
    - 对象引用起着句柄的作用
  - 引用的获取
  - 请求调用的特征
- 4、对象适配器 (Object Adapter)
  - 伺服程序 (Servant) 、三个要求、基本对象适配器、可移植对象适配器
- 5、内部ORB协议
  - 服务器应用程序的事件处理模型

## 三、CORBA应用程序的一般开发过程 P50

- 1、基于CORBA的系统包括客房客户程序和服务器程序

## 四、CORBA的基本服务 P50

- 1、命名服务 (Naming Service)
  - 命名服务给客户程序带来的好处
  - 命名图 (naming graph)
    - 名称绑定、命名上下文
  - 名称解析 (name resolve)
- 2、交易服务 (Trading Service)
  - 交易服务的基本轮廓
- 3、事件服务 (Event Service)
  - CORBA的事件服务模型
  - 事件服务发送事件的模型：推模型、拉模型

## 五、BES：一个优秀的CORBA开发平台

## 第七讲 结构设计空间及其量化

## 一、设计空间和规则 P58

- 1、设计空间
- 2、维
- 3、设计规则
- 4、设计知识库
  - 定义、使用设计知识库的好处
- 5、设计空间的组成：功能设计空间、结构设计空间

## 二、用户接口结构设计 P59

- 1、用户接口结构的设计空间
  - 基本结构模型：特定应用组件、可共享用户接口组件、依赖设备组件
- 2、功能维例子
  - 6个功能维可分为3组：外部需求；基本交互行为；考虑实际应用
  - 外部事件处理、用户定制、适应跨设备的用户接口、计算机系统组织、基本接口类型、考虑实际应用
- 3、结构维例子
  - 5个结构维可分为3组：模块中功能和信息的划分；表示问题；控制流程、通信、同步问题
  - 应用接口抽象层、抽象设备可变性、用户接口定义符号、基本通信、控制线程机制
- 4、用户接口结构的设计规则
  - 功能维与结构维的关系规则、结构维内部之间的关系规则

## 三、量化设计空间 P63

- 1、简介（目标、思想）
  - 质量配置函数（QFD）
    - 好处
    - 过程（基本原理）
- 2、量化空间设计（QDS）
  - QDS原理

# 第八章 软件体系结构描述

## 一、软件体系结构形式化的意义 P67

## 二、软件体系结构描述的方法 P68

- 1、常用的一般描述方法和形式化描述方法
- 2、主程序和子程序
  - 优缺点
- 3、数据抽象和面向对象设计

## 三、Z Notation简介 P70

- 1、什么是形式规范
- 2、Z notation的思想



#### 四、Z Notation的应用例子 P73

#### N notation具体介绍 P78