

第 9 章 软件架构设计

9.1 软件架构概述

9.1.1 软件架构的定义

1. 【2009 年题 35】软件架构贯穿于软件的整个生命周期,但在不同阶段对软件架构的关注力度并不相同,在()阶段,对软件架构的关注最多。
A. 需求分析与设计
B. 设计与实现
C. 实现与测试
D. 部署与变更
2. 【2010 年题 37】软件架构是降低成本、改进质量、按时和按需交付产品的关键因素。以下关于软件架构的描述,错误的是()。
A. 根据用户需求,能够确定一个最佳的软件架构,指导整个软件的开发过程
B. 软件架构设计需要满足系统的质量属性,如性能、安全性和可修改性等
C. 软件架构设计需要确定组件之间的依赖关系,支持项目计划和管理活动
D. 软件架构能够指导设计人员和实现人员的工作
3. 【2010 年题 38】软件架构设计包括提出架构模型、产生架构设计和进行设计评审等活动,是一个迭代的过程以下关于软件架构设计活动的描述,错误的是()。
A. 在建立软件架构的初期,一般需要选择一个合适的架构风格
B. 将架构分析阶段已标识的构件映射到架构中,并分析这些构件之间的关系
C. 软件架构设计活动将已标识构件集成到软件架构中,设计并实现这些构件
D. 一旦得到了详细的软件架构设计,需要邀请独立于系统开发的外部人员对系统进行评审
4. 【2011 年题 35】1995 年 Kruchten 提出了著名的“4+1”视图,用来描述软件系统的架构。在“4+1”视图中,()用来描述设计的对象模型和对象之间的关系;()描述了软件模块的组织与管理;()描述设计的并发和同步特征。
A. 逻辑视图 B. 用例视图 C. 过程视图 D. 开发视图
A. 逻辑视图 B. 用例视图 C. 过程视图 D. 开发视图
A. 逻辑视图 B. 用例视图 C. 过程视图 D. 开发视图
5. 【2012 年题 16】ANSI/IEEE 1471-2000 是对软件密集型系统的架构进行描述的标准。在该标准中,()这一概念主要用于描述软件架构模型。在此基础上,通常采用()描述某个利益相关人(Stakeholder)所关注架构模型的某一方面。()则是对所有利益相关人关注点的响应和回答。
A.上下文 B.架构风格 C.组件 D.视图
A.环境 B.资源 C.视角 D.场景
A.架构 B.系统 C.模型 D.使命

6. 【2013 年题 28】软件系统架构是关于软件系统的结构、()和属性的高级抽象。在描述阶段,主要描述直接构成系统的抽象组件以及各个组件之间的连接规则,特别是相对细致地描述组件的()。在实现阶段,这些抽象组件被细化为实际的组件,比如具体类或者对象。软件系统架构不仅指定了软件系统的组织和()结构,而且显示了系统需求和组件之间的对应关系,包括设计决策的基本方法和基本原理。
- (1)A. 行为 B. 组织 C. 性能 D. 功能
(2)A. 交互关系 B. 实现关系 C. 数据依赖 D. 功能依赖
(3)A. 进程 B. 拓扑 C. 处理 D. 数据
7. 【2013 年题 30】以下叙述,()不是软件架构的主要作用。
- A. 在设计变更相对容易的阶段,考虑系统结构的可选方案
B. 便于技术人员与非技术人员就软件设计进行交互
C. 展现软件的结构、属性与内部交互关系
D. 表达系统是否满足用户的功能性需求
8. 【2014 年题 28】以下关于软件架构设计重要性的描述,()是错误的。
- A. 软件架构设计能够满足系统的性能、安全性、可维护性等品质
B. 软件架构设计能够帮助项目干系人(Stakeholder)更好地理解软件结构
C. 软件架构设计能够帮助架构师更好地捕获和细化系统需求
D. 软件架构设计能够有效地管理系统的复杂性,并降低系统维护费用
9. 【2014 年题 29】将系统需求模型转换为架构模型是软件系统需求分析阶段的一项重要工作,以下描述中,()是在转换过程中需要关注的问题。
- A. 如何通过多视图模型描述软件系统的架构
B. 如何确定架构模型中有哪些元素构成
C. 如何采用表格或用例映射保证转换的可追踪性。
D. 如何通过模型转换技术,将高层架构模型逐步细化为细粒度架构模型
10. 【2015 年题 12】以下关于软件架构风格与系统性能的关系叙述中,错误的是()。
- A.对于采用层次化架构风格的系统,划分的层次越多,系统的性能越差
B.对于采用隐式调用架构风格的系统,可以通过处理函数的并发调用提高系统处理性能
C.采用面向对象架构风格的系统,可以通过引入对象管理层提高系统性能
D.对于采用解释器架构风格的系统,可以通过部分解释代码预先编译的方式提高系统性能
11. 【2015 年题 29】软件架构风格是描述某一特定应用领域中系统组织方式的惯用模式。架构风格反映领域中众多系统所共有的结构和(),强调对架构()的重用。
- A.语义特性 B.功能需求 C.质量属性 D.业务规则
A.分析 B.设计 C.实现 D.评估
12. 【2015 年题 38】软件架构设计包括提出架构模型,产生架构设计和进行设计评审等活动,是一个迭代的过程。架构设计主要关注软件组件的结构、属性和(),并通过多种()全面描述特定系统的架构。
- A.实现方式 B.交互作用 C.设计方案 D.测试方式
A.对象 B.代码 C.文档 D.视图
13. 【2016 年题 32】软件架构风格是描述某一特定应用领域中系统组织方式的惯用模式。一个体系结构定义了一个词汇表和一组()。架构风格反映领域中众多系统所共有的结构和()。
- A.约束 B.连接件 C.拓扑结构 D.规则

- A.语义特征 B.功能需求 C.质量属性 D.业务规则

9.1.2 软件架构的重要性

1. 【2009 年题 36】软件架构设计是降低成本、改进质量、按时和按需交付产品的关键活动。以下关于软件架构重要性的叙述中, 错误的是()。
A. 架构设计能够满足系统的性能、一可维护性等品质
B. 良好的架构设计能够更好地捕获并了解用户需求
C. 架构设计能够使得不同的利益相关人(stakeholders) 达成一致的目标
D. 架构设计能够支持项目计划和项目管理等活动
2. 【2014 年 28】以下关于软件架构设计重要性的描述, ()是错误的。
A. 软件架构设计能够满足系统的性能、安全性、可维护性等品质
B. 软件架构设计能够帮助项目干系人(Stakeholder)更好地理解软件结构
C. 软件架构设计能够帮助架构师更好地捕获和细化系统需求
D. 软件架构设计能够有效地管理系统的复杂性, 并降低系统维护费用

9.1.3 架构的模型

1. 【2012 年题 17】架构描述语言(Architecture Description Language, ADL)是一种为明确说明软件系统的概念架构和对这些概念架构建模提供功能的语言。ADL 主要包括以下组成部分: 组件、组件接口、()和架构配置。
A.架构风格 B.架构实现 C.连接件 D.组件约束
2. 【2015 年题 31】架构描述语言(Architecture Description Language, ADL)是一种为明确说明软件系统的概念架构和对这些概念架构建模提供功能的语言。ADL 主要包括以下组成部分: 组件、组件接口、()和架构配置。
A.架构风格 B.架构实现 C.连接件 D.组件约束

9.2 架构需求与软件质量属性

9.2.1 软件质量属性

1. 【2010 年题 46】某 服 务 器 软 件 系 统 对 可 用 性 (Availability)、性 能 (Performance)和可修改性(Modification)的要求较高, ()设计策略能提高该系统的可用性, ()设计策略能够提高该系统的性能, ()设计策略能够提高该系统的可修改性。
A. Ping/Echo B. 限制访问 C. 运行时注册 D. 接口-实现分离
A. 分层结构 B. 事务机制 C. 主动冗余 D. 队列调度
A. 信息隐藏 B. 记录/回放 C. 任务模型 D. 回滚
2. 【2010 年题 47】某服务器软件系统能够正确运行并得出计算结果, 但存在“系统出错后不能在要求的时间内恢复到正常状态”和“对系统进行二次开发时总要超过半年的时间”两个问题, 上述问题依次与质量属性中的()相关。
A. 可用性和性能
B. 性能和可修改性
C. 性能和可测试性
D. 可用性和可修改性

3. 【2011 年题 43】某公司在对一家用车库门嵌入式软件系统进行架构设计时, 识别出两个关键的质量属性场景, 其中“当车库门正常下降时, 如果发现下面有障碍物, 则系统停止下降的时间需要控制在 0.1 秒内”与 () 质量属性相关; “系统需要为部署在远程 PC 机上的智能家居系统留有控制接口, 并支持在智能家居系统中对该系统进行远程错误诊断与调试”与 () 质量属性相关。
- A. 可用性 B. 性能 C. 可修改性 D. 可测试性
- A. 可用性 B. 性能 C. 可修改性 D. 可测试性
4. 【2011 年题 44】软件质量属性通常需要采用特定的设计策略实现。例如, () 设计策略能提高该系统的可用性, () 设计策略能够提高该系统的性能, () 设计策略能够提高该系统的安全性。
- A. 心跳机制 B. 数据驱动 C. 关注点分离 D. 信息隐藏
- A. 引入中间层 B. 事务机制 C. 主动冗余 D. 优先级队列
- A. 信息隐藏 B. 内置监控器 C. 限制访问 D. 检查点
5. 【2012 年题 10】某公司欲开发一个在线交易系统, 在架构设计阶段, 公司的架构师识别出 3 个核心质量属性场景。其中“在并发用户数量为 1000 人时, 用户的交易请求需要在 0.5 秒内得到响应”主要与 () 质量属性相关, 通常可采用 () 架构策略实现该属性; “当系统由于软件故障意外崩溃后, 需要在 0.5 小时内恢复正常运行”主要与 () 质量属性相关, 通常可采用 () 架构策略实现该属性; “系统应该能够抵挡恶意用户的入侵行为, 并进行报警和记录”主要与 () 质量属性相关, 通常可采用 () 架构策略实现该属性。
- A. 性能 B. 安全性 C. 可用性 D. 可修改性
- A. 操作串行化 B. 资源调度 C. 心跳 D. 内置监控器
- A. 可测试性 B. 易用性 C. 可用性 D. 互操作性
- A. 主动冗余 B. 资源调度 C. 抽象接口 D. 记录/回放
- A. 可用性 B. 安全性 C. 可测试性 D. 可修改性
- A. 内置监控器 B. 记录/回放 C. 追踪审计 D. 维护现有接口
6. 【2014 年题 36】某公司欲开发一个在线交易系统, 在架构设计阶段, 公司的架构师识别出 3 个核心质量属性场景。其中“当系统面临断电故障后, 需要在 1 小时内切换至备份站点并恢复正常运行”主要与 () 质量属性相关, 通常可采用 () 架构策略实现该属性; “在并发用户数量为 1000 人时, 用户的交易请求需要在 0.5 秒内得到响应”主要与 () 质量属性相关, 通常可采用 () 架构策略实现该属性; “对系统的消息中间件进行替换时, 替换工作需要在 5 人/月内完成”主要与 () 质量属性相关, 通常可采用 () 架构策略实现该属性。
- A. 性能 B. 安全性 C. 可用性 D. 可修改性
- A. 操作隔离 B. 资源调度 C. 心跳 D. 内置监控器
- A. 性能 B. 易用性 C. 可用性 D. 互操作性
- A. 主动冗余 B. 资源调度 C. 抽象接口 D. 记录/回放
- A. 可用性 B. 安全性 C. 可测试性 D. 可修改性
- A. 接口-实现分离 B. 记录/回放 C. 内置监控器 D. 追踪审计
7. 【2015 年题 30】软件架构是降低成本、改进质量、按时和按需交付产品的关键因素。软件架构设计需满足系统的 (), 如性能、安全性和可修改性等, 并能够指导设计人员和实现人员的工作。
- A. 功能需求 B. 性能需求 C. 质量属性 D. 业务属性

8. 【2015 年题 40】某公司欲开发一个网上商城系统,在架构设计阶段,公司的架构师识别出 3 个核心质量属性场景,其中“系统主站断电后,能够在 2 分钟内自动切换到备用站点,并恢复正常运行”主要与()质量属性相关,通常可采用()架构策略实现该属性;“在并发用户数不超过 1000 人时,用户的交易请求应该在 0.5s 内完成”主要与()质量属性相关通常可采用()架构策略实现该属性;“系统应该能够抵挡恶意用户的入侵行为,并进行报警和记录”主要与()质量属性相关,通常可采用()架构策略实现该属性。
- A.性能 B.可用性 C.易用性 D.可修改性
A.主动冗余 B.信息隐藏 C.抽象接口 D.记录/回放
A.可测试性 B.易用性 C.性能 D.互操作性
A.操作串行化 B.资源调度 C.心跳 D.内置监控器
A.可用性 B.安全性 C.可测试性 D.可修改性
A.内置监控器 B.记录/回放 C.追踪审计 D.维护现有接口
9. 【2016 年题 30】()不属于可修改性考虑的内容。
- A. 可维护性 B.可扩展性 C.结构重构 D.可变性
10. 【2016 年题 37】某公司欲开发一个智能机器人系统,在架构设计阶段,公司的架构师识别出 3 个核心质量属性场景。其中“机器人系统主电源断电后,能够在 10 秒内自动启动备用电源并进行切换,恢复正常运行”主要与(1)质量属性相关,通常可采用(2)架构策略实现该属性;“机器人在正常运动过程中如果发现前方 2 米内有人或者障碍物,应在 1 秒内停止并在 2 秒内选择一条新的运行路径”主要与(3)质量属性相关,通常可采用(4)架构策略实现该属性;“对机器人的远程控制命令应该进行加密,从而能够抵挡恶意的入侵破坏行为,并对攻击进行报警和记录”主要与(5)质量属性相关,通常可采用(6)架构策略实现该属性。
- (1) A.可用性 B.性能 C.易用性 D.可修改性
(2) A.抽象接口 B.信息隐藏 C.主动冗余 D.记录/回放
(3) A.可测试性 B.易用性 C.互操作性 D.性能
(4) A.资源调度 B.操作串行化 C.心跳 D.内置监控器
(5) A.可用性 B.安全性 C.可测试性 D.可修改性
(6) A.内置监控器 B.追踪审计 C.记录/回放 D.维护现有接口
11. 【2017 年题 39】某公司欲开发一个在线交易网站,在架构设计阶段,公司的架构师识别出 3 个核心质量属性场景。其中“网站正常运行时,用户发起的交易请求应该在 3 秒内完成”主要与()质量属性相关,通常可采用()架构策略实现该属性;“在线交易主站宕机后,能够在 3 秒内自动切换至备用站点并恢复正常运行”主要与()质量属性相关,通常可采用()架构策略实现该属性;“系统应该具备一定的安全保护措施,从而能够抵挡恶意的入侵破坏行为,并对所有针对网站的攻击行为进行报警和记录”主要与()质量属性相关,通常可采用()架构策略实现该属性。
- A.可用性 B.性能 C.易用性 D.可修改性
A.抽象接口 B.信息隐藏 C.主动冗余 D.资源调度
A.可测试性 B.易用性 C.可用性 D.互操作性
A.记录/回放 B.操作串行 C.心跳 D.增加计算资源
A.可用性 B.安全性 C.可测试性 D.可修改性
A.追踪审计 B.Ping/Echo C.选举 D.维护现有接口

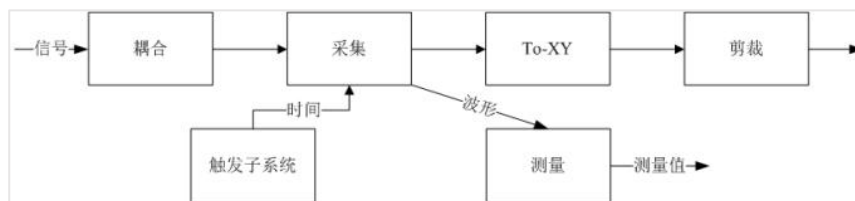
12. 【2018 年题 37】某公司欲开发一个人员管理系统，在架构设计阶段，公司的架构师识别出 3 个核心质量属性场景。其中“管理系统遭遇断电后，能够在 15 秒内自动切换至备用系统并恢复正常运行”主要与（58）质量属性相关，通常可采用（59）架构策略实现该属性；“系统正常运行时，人员信息查询请求应该在 2 秒内返回结果”主要与（60）质量属性相关，通常可采用（61）架构策略实现该属性；“系统需要对用户的操作情况进行记录，并对所有针对系统的恶意操作行为进行报警和记录”主要与（62）质量属性相关，通常可采用（63）架构策略实现该属性。
- （58）A. 可用性
B. 性能
C. 易用性
D. 可修改性
- （59）A. 抽象接口
B. 信息隐藏
C. 主动冗余
D. 影子操作
- （60）A. 可测试性
B. 易用性
C. 可用性
D. 性能
- （61）A. 记录/回放
B. 操作串行化
C. 心跳
D. 资源调度
- （62）A. 可用性
B. 安全性
C. 可测试性
D. 可修改性
- （63）A. 追踪审计
B. Ping/Echo
C. 选举
D. 维护现有接口

9.3 软件架构风格

1. 【2009 年题 41】Windows 操作系统在图形用户界面处理方面采用的核心架构风格是（ ）风格。Java 语言宣传的“一次编写，到处运行”的特性，从架构风格上看符合（ ）风格的特点。
- (1) A. 虚拟机
B. 管道-过滤器
C. 事件驱动
D. 微内核-扩展
- (2) A. 虚拟机
B. 管道-过滤器
C. 事件驱动

- D. 微内核-扩展
2. 【2009 年题 42】某软件开发公司负责开发一个 Web 服务器服务端处理软件,其核心部分是对客户端请求消息的解析与处理,包括 HTTP 报头分离、SOAP 报文解析等功能。该公司的架构师决定采用成熟的架构风格指导整个软件的设计,以下()架构风格,最适合该服务端处理软件。
- A. 虚拟机
 - B. 管道-过滤器
 - C. 黑板结构
 - D. 分层结构
3. 【2009 年题 43】某公司欲开发一个基于图形用户界面的集成调试器。该调试器的编辑器和变量监视器可以设置调试断点。当调试器在断点处暂停运行时,编辑程序可以自动卷屏到断点,变量监视器刷新变量数值。针对这样的功能描述,采用()的架构风格最为合适。
- A. 数据共享
 - B. 虚拟机
 - C. 隐式调用
 - D. 显式调用
4. 【2009 年题 44】某公司欲开发一种工业机器人,用来进行汽车零件的装配。公司的架构师经过分析与讨论,给出了该机器人控制软件的两种候选架构方案:闭环控制和分层结构。以下对于这两种候选架构的选择理由,错误的是()。
- A. 应该采用闭环控制架构,因为闭环结构给出了将软件分解成几个协作构件的方法,这对于复杂任务特别适合
 - B. 应该采用闭环控制结构,因为闭环控制架构中机器人的主要构件监控器、传感器、发动机等)是彼此分开的,并能够独立替换
 - C. 应该采用分层结构,因为分层结构很好地组织了用来协调机器人操作的构件,系统结构更加清晰
 - D. 应该采用分层结构,因为抽象层的存在,满足了处理不确定性的需要:在较低层次不确定的实现细节在较高层次会变得确定
5. 【2009 年题 45】一个软件的架构设计是随着技术的不断进步而不断变化的。以编译器为例,其主流架构经历了管道-过滤器到数据共享为中心的转变过程。以下关于编译器架构的叙述中,错误的是()。
- A. 早期的编译器采用管道-过滤器架构风格,以文本形式输入的代码被逐步转化为各种形式,最终生成可执行代码
 - B. 早期的编译器采用管道-过滤器架构风格,并且大多数编译器在词法分析时创造独立的符号表,在其后的阶段会不断修改符号表,因此符号表并不是程序数据的一部分
 - C. 现代的编译器采用以数据共享为中心的架构风格,主要关心编译过程中程序的中间表示
 - D. 现代的编译器采用以数据共享为中心的架构风格,但由于分析树是在语法分析阶段结束后才产生作为语义分析的输入,因此分析树不是数据中心的共享数据
6. 【2009 年题 46】(1)的选择是开发一个软件系统时的基本设计决策;(2)是最低层的模式,关注软件系统的设计与实现,描述了如何实现构件及构件之间的关系。引用一计数是 C++管理动态资源时常用的一种(3)。
- (1)A. 架构模式

- B. 惯用法
C. 设计模式
D. 分析模式
- (2) A. 架构模式
B. 惯用法
C. 设计模式
D. 分析模式
- (3) A. 架构模式
B. 惯用法
C. 设计模式
D. 分析模式
7. 【2010 年题 33】分布式系统开发中, 通常需要将任务分配到不同的逻辑计算层。业务数据的综合计算分析任务属于()。
- A. 表示逻辑层
B. 应用逻辑层
C. 数据处理层
D. 数据层
8. 【2010 年题 34】在客户机/服务器系统开发中, 采用()时, 应将数据层和数据处理层放置于服务器, 应用逻辑层、表示逻辑层和表示层放置于客户机。
- A. 分布式表示结构
B. 分布式应用结构
C. 分布式数据和应用结构
D. 分布式数据结构
9. 【2010 年题 40】某游戏公司欲开发一个大型多人即时战略游戏, 游戏设计的目标之一是能够支持玩家自行创建战役地图, 定义游戏对象的行为和之间的关系。针对该目标, 公司应该采用()架构风格最为合适。
- A. 管道-过滤器
B. 隐式调用
C. 主程序-子程序
D. 解释器
10. 【2010 年题 41】某公司欲为某种型号的示波器开发内置软件。该公司的架构师设计了如下图所示的软件架构。在软件架构评审时, 专家认为该架构存在的问题是()。



- A. 在功能划分上将各个模块独立起来
B. 在硬件构件的混合和替换方面不是很灵活
C. 没有清晰地说明用户怎样与其交互
D. 没有明确的层次关系, 没有强调功能之间的交互

11. 【2010 年题 42】某公司承接了一个开发家用空调自动调温器的任务，调温器测量外部空气温度，根据设定的期望温度控制空调的开关。根据该需求，公司应采用()架构风格最为合适。
- A. 解释器
 - B. 过程控制
 - C. 分层
 - D. 管道-过滤器
12. 【2010 年题 43】某公司欲开发一个漫步者机器人，用来完成火星探测任务。机器人的控制者首先定义探测任务和任务之间的时序依赖性，机器人接受任务后，需要根据自身状态和外界环境进行动态调整，最终自动完成任务。针对这些需求，该机器人应该采用()架构风格最为合适。
- A. 解释器
 - B. 主程序-子程序
 - C. 隐式调用
 - D. 管道-过滤器
13. 【2010 年题 44】某公司欲开发一个语音识别系统，语音识别的主要过程包括分割原始语音信号、识别音素、产生候选词、判定语法片断、提供语义解释等。每个过程都需要进行基于先验知识的条件判断并进行相应的识别动作。针对该系统的特点，采用()架构风格最为合适。
- A. 解释器
 - B. 面向对象
 - C. 黑板
 - D. 隐式调用
14. 【2011 年题 34】() 描述了一类软件架构的特征，它独立于实际问题，强调软件系统中通用的组织结构选择。垃圾回收机制是 Java 语言管理内存资源时常用的一种()。
- A. 架构风格
 - B. 开发方法
 - C. 设计模式
 - D. 分析模式
15. 【2011 年题 37】某公司研发一种语音识别软件系统，需要对用户的语音指令进行音节分割、重音判断、语法分析和语义分析，最终对用户的意图进行推断。针对上述功能需求，该语音识别软件应该采用()架构风格最为合适。
- A. 隐式调用
 - B. 管道-过滤器
 - C. 解释器
 - D. 黑板
16. 【2011 年题 38】某企业内部现有的主要业务功能已经封装为 Web 服务。为了拓展业务范围，需要将现有的业务功能进行多种组合，形成新的业务功能。针对业务灵活组合这一要求，采用()架构风格最为合适。
- A. 管道-过滤器
 - B. 解释器
 - C. 显式调用
 - D. 黑板

17. 【2011 年题 39】编译器的主要工作过程是将以文本形式输入的代码逐步转化为各种形式,最终生成可执行代码。现代编译器主要关注编译过程和程序的中间表示,围绕程序的各种形态进行转化与处理。针对这种特征,现代编译器应该采用()架构风格最为合适。
- A. 数据共享
B. 虚拟机
C. 隐式调用
D. 管道-过滤器
18. 【2012 年题 33】以下关于软件架构风格与系统性能关系的叙述,错误的是()。
- A. 对于采用层次化架构风格的系统,划分的层次越多,系统的性能越差
B. 对于采用管道-过滤器架构风格的系统,可以通过引入过滤器的数据并发处理提高系统性能
C. 对于采用面向对象架构风格的系统,可以通过减少功能调用层次提高系统性能
D. 对于采用过程调用架构风格的系统,可以通过将显式调用策略替换为隐式调用策略提高系统性能
19. 【2013 年题 29】软件架构风格是描述某一特定应用领域中系统组织方式的惯用模式。架构风格定义了一类架构所共有的特征,主要包括架构定义、架构词汇表和架构()。
- A. 描述 B. 组织 C. 约束 D. 接口
20. 【2013 年题 32】“编译器”是一种非常重要的基础软件,其核心功能是对源代码形态的单个或一组源程序依次进行预处理、词法分析、语法分析、语义分析、代码生成、代码优化等处理,最终生成目标机器的可执行代码。考虑以下与编译器相关的软件架构设计场景:传统的编译器设计中,上述处理过程都以独立功能模块的形式存在,程序源代码作为一个整体,依次在不同模块中进行传递,最终完成编译过程。针对这种设计思路,传统的编译器采用()架构风格比较合适。
- 随着编译、链接、调试、执行等开发过程的一体化趋势发展,集成开发环境 (IDE) 随之出现。IDE 集成了编译器、连接器、调试器等多种工具,支持代码的增量修改与处理,能够实现不同工具之间的信息交互,覆盖整个软件开发生命周期。针对这种需求,IDE 采用()架构风格比较合适。IDE 强调交互式编程,用户在修改程序代码后,会同时触发语法高亮显示、语法错误提示、程序结构更新等多种功能的调用与结果呈现,针对这种需求,通常采用()架构风格比较合适。
- 某公司已经开发了一款针对某种嵌入式操作系统专用编程语言的 IDE,随着一种新的嵌入式操作系统上市并迅速占领市场,公司决定对 IDE 进行适应性改造,支持采用现有编程语言进行编程,生成符合新操作系统要求的运行代码,并能够在现有操作系统上模拟出新操作系统的运行环境,以支持代码调试工作。针对上述要求,为了使 IDE 能够生成符合新操作系统要求的运行代码,采用基于()的架构设计策略比较合适;为了模拟新操作系统的运行环境,通常采用()架构风格比较合适。
- (1)A.管道-过滤器 B.顺序批处理 C.过程控制 D.独立进程
(2)A.规则引擎 B.解释器 C.数据共享 D.黑板
(3)A.隐式调用 B.显式调用 C.主程序-子程序 D.层次结构
(4)A.代理 B.适配 C.包装 D.模拟
(5)A.隐式调用 B.仓库结构 C.基于规则 D.虚拟机

21. 【2014 年题 34】软件架构风格描述某一特定领域中的系统组织方式和惯用模式,反映了领域中众多系统所共有的()特征。对于语音识别、知识推理等问题复杂、解空间很大、求解过程不确定的这一类软件系统。通常会采用()架构风格。
- (1)A. 语法和语义
B. 结构和语义
C. 静态和动态
D. 行为和约束
- (2)A. 管道-过滤器
B. 解释器
C. 黑板
D. 过程控制
22. 【2015 年题 33】某公司拟开发了个轿车巡航定速系统,系统需要持续测量车辆当前的实时速度,并根据设定的期望速度启动控制轿车的油门和刹车。针对上述需求,采用()架构风格最为合适。
- A. 解释器 B.过程控制 C.分层 D.管道-过滤器
23. 【2015 年题 34】某公司拟开发一个地面清洁机器人。机器人的控制者首先定义清洁任务和任务之间的关系,机器人接受任务后,需要响应外界环境中触发的一些突发事件,根据自身状态进行动态调整,最终自动完成任务。针对上述需求,该机器人应该采用()架构风格最为合适。
- A.面向对象
B.主程序-子程序
C.规则系统
D.管道-过滤器
24. 【2015 年题 35】某公司拟开发一个语音识别系统,其语音识别的主要过程包括分割原始语音信号、识别音素、产生候选词、判定语法片断、提供语义解释等,每个过程都需要进行基于先验知识的条件判断并进行相应的识别动作。针对该系统的特点,采用()架构风格最为合适。
- A.解释器
B.面向对象
C.黑板
D.隐式调用
25. 【2015 年题 36】某公司拟开发一套在线游戏系统,该系统的设计目标之一是支持用户自行定义游戏对象属性,行为和对象之间的交互关系。为了实现上述目标,公司应该采用()架构风格最为合适。
- A.管道-过滤器 B.隐式调用 C.主程序-子程序 D.解释器
26. 【2015 年题 37】某公司为其研发的硬件产品设计实现了一种特定的编程语言,为了方便开发者进行软件开发,公司拟开发一套针对该编程语言的集成开发环境,包括代码编辑、语法高亮、代码编译、运行调试等功能。针对上述描述,该集成开发环境应采用()架构风格最为合适。
- A.管道-过滤器 B.数据仓储 C.主程序-子程序 D.解释器
27. 【2016 年题 28】()架构风格可以概括为通过连接件绑定在一起按照一组规则运作的并行构件。
- A.C2 B.黑板系统 C.规则系统 D.虚拟机

28. 【2016 年题 31】某公司拟为某种新型可编程机器人开发相应的编译器。该编译过程包括词法分析、语法分析、语义分析和代码生成四个阶段，每个阶段产生的结果作为下一个阶段的输入，且需独立存储。针对上述描述，该集成开发环境应采用（ ）架构风格最为合适。
- A. 管道-过滤器 B.数据仓储 C.主程序-子程序 D.解释器
29. 【2016 年题 33】某公司拟开发一个扫地机器人。机器人的控制者首先定义清洁流程和流程中任务之间的关系，机器人接受任务后，需要响应外界环境中触发的一些突发事件，根据自身状态进行动态调整，最终自动完成任务。针对上述需求，该机器人应该采用（ ）架构风格最为合适。
- A. 面向对象 B.主程序-子程序 C.规则系统 D.管道-过滤器
30. 【2016 年题 34】某企业内部现有的主要业务功能已封装成为 Web 服务。为了拓展业务范围，需要将现有的业务功能进行多种组合，形成新的业务功能。针对业务灵活组合这一要求，采用（ ）架构风格最为合适。
- A.规则系统 B.面向对象 C.黑板 D.解释器
31. 【2016 年题 35】某公司拟开发一个语音搜索系统，其语音搜索系统的主要工作过程包括分割原始语音信号、识别音素、产生候选词、判定语法片断、提供搜索关键词等，每个过程都需要进行基于先验知识的条件判断并进行相应的识别动作。针对该系统的特点，采用（ ）架构风格最为合适。
- A.分层系统 B.面向对象 C.黑板 D.隐式调用
32. 【2017 年题 34】软件架构风格描述某一特定领域中的系统组织方式和惯用模式，反映了领域中众多系统所共有的()特征。对于语音识别、知识推理等问题复杂、解空间很大、求解过程不确定的这一类软件系统，通常会采用()架构风格。对于因数据输入某个构件，经过内部处理，产生数据输出的系统，通常会采用()架构风格。
- A.语法和语义 B.结构和语义 C.静态和动态 D.行为和约束
- A.管道-过滤器 B.解释器 C.黑板 D.过程控制
- A.事件驱动系统 B.黑板 C.管道-过滤器 D.分层系统
33. 【2017 年题 35】某公司拟开发一个 VIP 管理系统，系统需要根据不同商场活动，不定期更新 VIP 会员的审核标准和 VIP 折扣系统。针对上述需求，采用()架构风格最为合适。
- A.规则系统 B.过程控制 C.分层 D.管道-过滤器
34. 【2017 年题 36】某公司拟开发一个新闻系统，该系统可根据用户的注册兴趣，向用户推送其感兴趣的新闻内容，该系统应该采用()架构风格最为合适。
- A.事件驱动系统 B.主程序-子程序 C.黑板 D.管道-过滤器
35. 【2017 年题 37】系统中的构件和连接件都有一个顶部和一个底部，构件的顶部应连接到某连接件的底部，构件的底部则应连接到某连接的顶部，构件和构件之间不允许直接连接，连接件直接连接时，必须由其中一个的底部连接到另一个的顶部。上述构件和连接件的组织规则描述的是()架构风格。
- A.管道-过滤器 B.分层系统 C.C2 D.面向对象
36. 【2018 年题 34】在仓库风格中，有两种不同的构件，其中，(49)说明当前状态，(50)在中央数据存储上执行。
- (49) A. 注册表
B. 中央数据结构
C. 事件
D. 数据库

- (50) A. 独立构件
 - B. 数据结构
 - C. 知识源
 - D. 共享数据
37. 【2018 年题 35】某公司欲开发一个大型多人即时战略游戏，游戏设计的目标之一是能够支持玩家自行创建战役地图，定义游戏对象的行为和对象之间的关系。针对该需求，公司应该采用 (51) 架构风格最为合适。在架构设计阶段，公司的架构师识别出 2 个核心质量属性场景。其中，“在并发用户数量为 10000 人时，用户的请求需要在 1 秒内得到响应”主要与 (52) 质量属性相关；“对游戏系统进行二次开发的时间不超过 3 个月”主要与 (53) 质量属性相关。
- (51) A. 层次系统
 - B. 解释器
 - C. 黑板
 - D. 事件驱动系统
- (52) A. 性能
 - B. 吞吐量
 - C. 可靠性
 - D. 可修改性
- (53) A. 可测试性
 - B. 可移植性
 - C. 互操作性
 - D. 可修改性

9.4 层次系统架构风格

9.4.3 MVC 架构风格

1. 【2009 年题 29】在一个典型的基于 MVC(Model-View-Controller) 的 J2EE 应用中，分发客户请求、有效组织其它构件为客户端提供服务的控制器由() 实现。
- A. Entity Bean
 - B. Session Bean
 - C. Servlet
 - D. JSP

9.8 软件架构评估

1. 【2009 年题 50】识别风险点、非风险点、敏感点和权衡点是 ATAM 方法中的关键步骤。已知针对某系统所做的架构设计中，提高其加密子系统的加密级别将对系统的安全性和性能都产生非常大的影响，则该子系统一定属于()。
- A. 风险点和敏感点
 - B. 权衡点和风险点
 - C. 权衡点和敏感点
 - D. 风险点和非风险点

2. 【2010 年题 51】正确识别风险点、非风险点、敏感点和权衡点是进行软件架构评价的关键步骤。其中()是实现一个特定质量属性的关键特征,该特征为一个或多个软件构件所共有。“改变加密的级别可能会对安全性和性能都产生显著的影响”,这是一个对系统()的描述。
- A. 风险点 B. 非风险点 C. 敏感点 D. 权衡点
- A. 风险点 B. 非风险点 C. 敏感点 D. 权衡点
3. 【2011 年题 45】架构权衡分析方法(ATAM)是一种常用的软件架构评估方法,下列关于该方法的叙述中,正确的是()。
- A. ATAM 需要对代码的质量进行评估
B. ATAM 需要对软件系统需求的正确性进行评价
C. ATAM 需要对软件系统进行集成测试
D. ATAM 需要对软件质量属性进行优先级排序
4. 【2011 年题 46】识别风险点、非风险点、敏感点和权衡点是软件架构评估过程中的关键步骤。针对某系统所作的架构设计中,“系统需要支持的最大并发用户数量直接影响传输协议和数据格式”描述了系统架构设计中的一个();“由于系统的业务逻辑目前尚不清楚,因此现有系统三层架构中的第二层可能会出现功能重复,这会影响系统的可修改性”描述了系统架构设计中的一个()。
- A.敏感点 B.风险点 C.非风险点 D.权衡点
- A.敏感点 B.风险点 C.非风险点 D.权衡点
5. 【2013 年题 34】架构权衡分析方法(ArchitectureTradeoffAnalysisMethod, ATAM)是一种系统架构评估方法,主要在系统开发之前,针对性能、()、安全性和可修改性等质量属性进行评价和折中。ATAM 可以分为 4 个主要的活动阶段,包括需求收集、()描述、属性模型构造和分析、架构决策与折中,整个评估过程强调以()作为架构评估的核心概念。
- 某软件公司采用 ATAM 进行软件架构评估,在评估过程中识别出了多个关于质量属性的描述。其中,“系统在进行文件保存操作时,应该与 Windows 系统的操作方式保持一致,主要与()质量属性相关:“系统应该提供一个开放的 API 接口,支持远程对系统的行为进行控制与调试,主要与()质量属性相关。在识别出上述描述后,通常采用()对质量属性的描述进行刻画与排序。在评估过程中,()是一个会影响多个质量属性的架构设计决策。
- (1)A. 可测试性 B. 可移植性 C. 可用性 D. 易用性
- (2)A. 架构视图 B. 架构排序 C. 架构风格 D. 架构策略
- (3)A. 用例 B. 视图 C. 属性 D. 模型
- (4)A. 可测试性 B. 互操作性 C. 可移植性 D. 易用性
- (5)A. 可测试性 B. 互操作性 C. 可移植性 D. 易用性
- (6)A. 期望管理矩阵 B. 决策表 C. 优先队列 D. 效用树
- (7)A. 风险点 B. 决策点 C. 权衡点 D. 敏感点
6. 【2014 年题 35】在对一个软件系统的架构进行设计与确认之后,需要进行架构复审。架构复审的目的是为了标识潜在的风险,及早发现架构设计中的缺陷和错误。在架构复审过程中,主要由()决定架构是否满足需求、质量需求是否在设计中得到体现。
- A. 系统分析师与架构师
B. 用户代表与领域专家
C. 系统所有者与项目经理
D. 系统开发与测试人员

7. 【2014 年题 37】识别风险、非风险、敏感点和权衡点是进行软件架构评估的重要过程。“改变业务数据编码方式会对系统的性能和安全性产生影响”是对()的描述,“假设用户请求的频率为每秒 1 个,业务处理时间小于 30 毫秒,则将请求响应时间设定为 1 秒钟是可以接受的”是对()的描述。
- A. 风险点 B. 非风险 C. 敏感点 D. 权衡点
- A. 风险点 B. 非风险 C. 敏感点 D. 权衡点
8. 【2015 年 41】架构权衡分析方法(Architecture Tradeoff Analysis Method, ATAM)是在基于场景的架构分析方法(Scenarios-based Architecture Analysis Method, SAAM)基础上发展起来的,主要包括场景和需求收集、(1),属性模型构造和分析,属性模型折中等四个阶段。ATAM 方法要求在系统开发之前,首先对这些质量属性进行(2)和折中。
- (1) A. 架构视图和场景实现
B. 架构风格和场景分析
C. 架构设计和目标分析
D. 架构描述和需求评估
- (2) A. 设计 B. 实现 C. 测试 D. 评价
9. 【2018 年 33】体系结构权衡分析方法(Architecture Tradeoff Analysis Method, ATAM)包含 4 个主要的活动领域,分别是场景和需求收集、体系结构视图和场景实现、(47)、折中。基于场景的架构分析方法(Scenarios-based Architecture Analysis Method, SAAM)的主要输入是问题描述、需求声明和(48)。
- (47) A. 架构设计
B. 问题分析与建模
C. 属性模型构造和分析
D. 质量建模
- (48) A. 问题说明
B. 问题建模
C. 体系结构描述
D. 需求建模

9.8.1 软件架构评估的方法

1. 【2009 年题 49】Architecture Tradeoff Analysis Method (ATAM) 是一种软件架构的评估方法,以下关于该方法的叙述中,正确的是()。
- A. ATAM 是一种代码评估方法
B. ATAM 需要评估软件的需求是否准确
C. ATAM 需要对软件系统进行测试
D. ATAM 不是一种精确的评估工具
2. 【2012 年题 9】基于场景的架构分析方法(Scenarios-based Architecture Analysis Method, SAAM)是卡耐基梅隆大学软件工程研究所的 Kazman 等人于 1983 年提出的一种非功能质量属性的架构分析方法,是最早形成文档并得到广泛应用的软件架构分析方法。SAAM 的主要输入是问题描述、()和架构描述文档,其分析过程主要包括场景开发、()、单个场景评估、场景交互和总体评估。
- A. 问题说明 B. 问题建模 C. 需求说明 D. 需求建模
- A. 架构需求 B. 架构描述 C. 架构设计 D. 架构实现

3. 【2014 年题 38】体系结构权衡分析方法(Architecture Tradeoff Analysis Method, ATAM)是一种常见的系统架构评估框架,该框架主要关注系统的(),针对性能()安全性和可修改性,在系统开发之前进行分析、评价与折中。
- A. 架构视图 B. 架构描述 C. 需求说明 D. 需求建模
A. 可测试性 B. 可用性 C. 可移植性 D. 易用性

9.9 构件及其复用

1. 【2009 年题 28】以下关于软件构件及其接口的叙述,错误的是()。
- A. 构件是软件系统中相对独立且具有一定意义的构成成分
B. 构件在容器中进行管理并获取其属性或者服务
C. 构件不允许外部对所支持的接口进行动态发现或调用
D. 构件可以基于对象实现,也可以不基于对象实现
2. 【2016 年题 22】面向构件的编程(Component Oriented Programming, COP)关注于如何支持建立面向构件的解决方案。面向构件的编程所需要的基本支持包括(35)。
- A. 继承性、构件管理和绑定、构件标识、访问控制
B. 封装性、信息隐藏、独立部署、模块安全性
C. 多态性、模块封装性、后期绑定和装载、安全性
D. 构件抽象、可替代性、类型安全性、事务管理
3. 【2016 年题 24】关于构件的描述,正确的是()。
- A. 构件包含了一组需要同时部署的原子构件
B. 构件可以单独部署,原子构件不能被单独部署
C. 一个原子构件可以同时多个构件家族中共享
D. 一个模块可以看作带有单独资源的原子构件
4. 【2018 年题 26】EJB 是企业级 Java 构件,用于开发和部署多层结构的、分布式的、面向对象的 Java 应用系统。其中,(36)负责完成服务端与客户端的交互;(37)用于数据持久化来简化数据库开发工作;(38)主要用来处理并发和异步访问操作。
- (36) A. 会话型构件
B. 实体型构件
C. COM 构件
D. 消息驱动构件
- (37) A. 会话型构件
B. 实体型构件
C. COM 构件
D. 消息驱动构件
- (38) A. 会话型构件
B. 实体型构件
C. COM 构件
D. 消息驱动构件
5. 【2018 年题 27】构件组装成软件系统的过程可以分为三个不同的层次:()。
- A. 初始化、互连和集成
B. 连接、集成和演化

- C. 定制、集成和扩展
 - D. 集成、扩展和演化
6. 【2018 年题 28】CORBA 服务端构件模型中, () 是 CORBA 对象的真正实现, 负责完成客户端请求。
- A. 伺服对象 (Servant)
 - B. 对象适配器 (Object Adapter)
 - C. 对象请求代理 (Object Request Broker)
 - D. 适配器激活器 (Adapter Activator)
7. 【2018 年题 29】J2EE 应用系统支持五种不同类型的构件模型, 包括 ()。
- A. Applet、JFC、JSP、Servlet、EJB
 - B. JNDI、IIOP、RMI、EJB、JSP/Servlet
 - C. JDBC、EJB、JSP、Servlet、JCA
 - D. Applet、Servlet、JSP、EJB、Application Client

9.10 产品线及系统演化

9.10.4 特定领域软件架构

1. 【2010 年题 45】特定领域软件架构(DSSA)是在一个特定应用领域为一组应用提供组织结构参考的标准软件架构。实施 DSSA 的过程中包括一系列基本的活动, 其中()活动的主要目的是为了获得 DSSA。该活动参加人员中, ()的主要任务是提供关于领域中系统的需求规约和实现的知识。
- (1)A. 领域需求
 - B. 领域分析
 - C. 领域设计
 - D. 领域实现
 - (2)A. 领域专家
 - B. 领域分析者
 - C. 领域设计者
 - D. 领域实现者
2. 【2012 年题 11】特定领域软件架构(Domain Specific Software Architecture, DSSA)是在一个特定应用领域中, 为一组应用提供组织结构参考的标准软件体系结构。DSSA 的基本活动包括领域分析、领域设计和领域实现。其中领域分析的主要目的是获得(), 从而描述领域中系统之间共同的需求, 即领域需求; 领域设计的主要目标是获得(), 从而描述领域模型中表示需求的解决方案; 领域实现的主要目标是开发和组织可重用信息, 并对基础软件架构进行实现。
- (1)A. 领域边界
 - B. 领域信息
 - C. 领域对象
 - D. 领域模型
 - (2)A. 特定领域软件需求
 - B. 特定领域软件架构
 - C. 特定领域软件设计模型

D.特定领域软件重用模型

3. 【2013 年题 31】特定领域软件架构(Domain Specific Software Architecture, DSSA)是在一个特定应用领域中,为一组应用提供组织结构参考的标准软件体系结构。DSSA 通常是一个具有三个层次的系统模型,包括()环境、领域特定应用开发环境和应用执行环境,其中()主要在领域特定应用开发环境中工作。
- (1)A. 领域需求
B. 领域开发
C. 领域执行
D. 领域应用
- (2)A. 操作员
B. 领域架构师
C. 应用工程师
D. 程序员
4. 【2014 年题 32】特定领域软件架构(Domain Specific Software Architecture, DSSA)是在一个特定应用领域中,为一组应用提供组织结构参考的标准软件体系结构。参加 DSSA 的人员可以划分为多种角色,其中()的任务是控制整个领域分析过程,进行知识获取,将获取的知识组织到领域模型中;()的任务是根据领域模型和现有系统开发出 DSSA,并对 DSSA 的准确性和一致性进行验证。
- A. 领域专家 B. 领域分析者 C. 领域设计者 D. 领域实现者
- A. 领域专家 B. 领域分析者 C. 领域设计者 D. 领域实现者
5. 【2015 年题 39】特定领域软件架构(Domain Specific Software Architecture, DSSA)以一个特定问题领域为对象,形成由领域参考模型,参考需求,(1)等组成的开发基础架构,支持一个特定领域中多个应用的生成。DSSA 的基本活动包括领域分析、领域设计和领域实现。其中领域分析的主要目的是获得(2),从而描述领域中系统之间共同的需求,即领域需求;领域设计的主要目标是获得(3),从而描述领域模型中表示需求的解决方案;领域实现的主要目标是开发和组织可重用信息,并实现基础软件架构。
- (1)A.参考设计 B.参考规约 C.参考架构 D.参考实现
- (2)A.领域边界 B.领域信息 C.领域对象 D.领域模型
- (3)A.特点领域软件需求 B.特定领域软件架构
C.特定领域软件设计模型 D.特定领域软件重用模型
6. 【2016 年题 29】DSSA (特定领域的软件架构, domain-specific software architecture)是在一个特定应用领域中为一组应用提供组织结构参考的软件体系结构,参与 DSSA 的人员可以划分为 4 种角色,包括领域专家、领域设计人员、领域实现人员和(),其基本活动包括领域分析、领域设计和()。
- A.领域测试人员 B.领域顾问 C.领域分析师 D.领域经理
- A.领域建模 B.架构设计 C.领域实现 D.领域评估
7. 【2018 年题 32】特定领域软件架构(Domain Specific Software Architecture, DSSA)的基本活动包括领域分析、领域设计和领域实现。其中,领域分析的主要目的是获得领域模型。领域设计的主要目标是获得(45)。领域实现是为了(46)。
- (45) A. 特定领域软件需求
B. 特定领域软件架构
C. 特定领域软件设计模型
D. 特定领域软件重用模型
- (46) A. 评估多种软件架构

- B. 验证领域模型
- C. 开发和组织可重用信息, 对基础软件架构进行实现
- D. 特定领域软件重用模型

9.11 其他

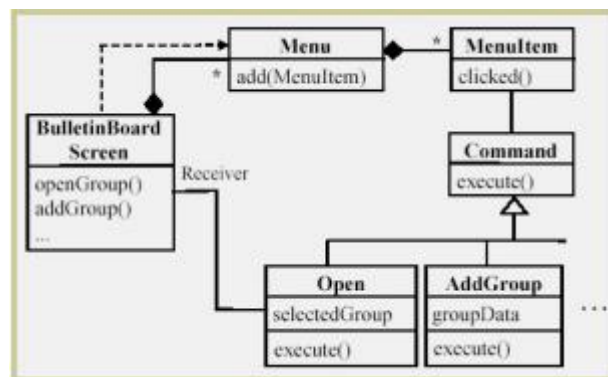
1. 【2010 年题 30】() 是一个独立可交付的功能单元, 外界通过接口访问其提供的服务。
 - A. 面向对象系统中的对象(Object)
 - B. 模块化程序设计中的子程序(Subroutine)
 - C. 基于构件开发中的构件(Component)
 - D. 系统模型中的包(Package)
2. 【2010 年题 32】对象管理组织(OMG)基于 CORBA 基础设施定义了四种构件标准。其中, () 的状态信息是由构件自身而不是由容器维护。
 - A. 实体构件 B. 加工构件 C. 服务构件 D. 会话构件
3. 【2016 年题 23】CORBA (Common Object Request Broker Architecture, 公共对象请求代理体系结构, 通用对象请求代理体系结构) 构件模型中, () 的作用是在底层传输平台与接收调用并返回结果的对象实现之间进行协调, () 是最终完成客户请求的服务对象实现。
 - A. 伺服对象激活器 B. 适配器激活器 C. 伺服对象定位器 D. 可移植对象适配器 POA
 - A. CORBA 对象 B. 分布式对象标识 C. 伺服对象 Servant D. 活动对象映射表
4. 【2016 年题 25】面向服务系统构建过程中, () 用于实现 Web 服务的远程调用, () 用来将分散的、功能单一的 Web 服务组织成一个复杂的有机应用。
 - (1)、A. UDDI (Universal Description, Discovery and Integration)
 - B. WSDL (Web Service Description Language)
 - C. SOAP (Simple Object Access Protocol)
 - D. BPEL (Business Process Execution Language)
 - (2)、A. UDDI (Universal Description, Discovery and Integration)
 - B. WSDL (Web Service Description Language)
 - C. SOAP (Simple Object Access Protocol)
 - D. BPEL (Business Process Execution Language)
5. 【2016 年题 26】基于 JavaEE 平台的基础功能服务构建应用系统时, () 可用来集成遗产系统。
 - A. JDBC、JCA 和 Java IDL B. JDBC、JCA 和 JMS
 - C. JDBC、JMS 和 Java IDL D. JCA、JMS 和 Java IDL

第 10 章 设计模式

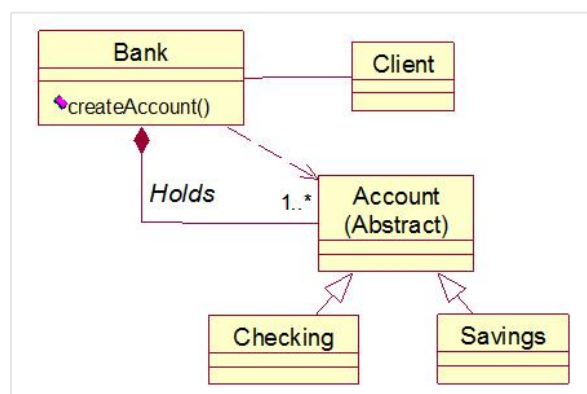
10.1 设计模式概述

10.1.3 GoF 设计模式

1. 【2009 年题 24】某软件公司欲开发一个 Windows 平台上的公告板系统。在明确用户需求后，该公司的架构师决定采用 Command 模式实现该系统的界面显示部分，并设计 UML 类图如下图所示。图中与 Command 模式中的“Invoker”角色相对应的类是()，与“ConcreteCommand”角色相对应的类是()。

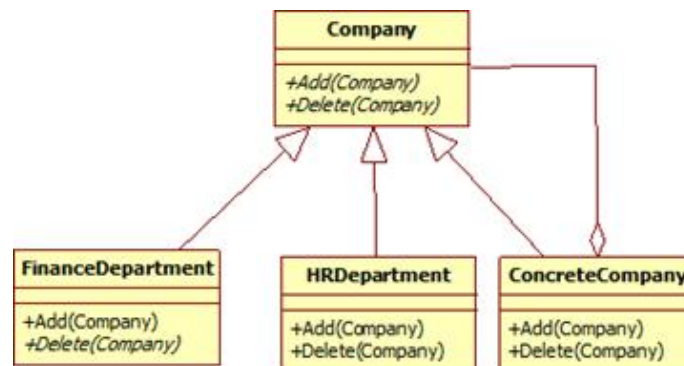


- A. Command B. MenuItem C. Open D. ButktinBoardScreen
- A. Command B. MenuItem C. Open D. BulktinBoardScreen
2. 【2009 年题 27】某软件公司基于面向对象技术开发了一套图形界面显示构件库 VisualComponent。在使用该库构建某图形界面时，用户要求为界面定制一些特效显示效果，如带滚动条、能够显示艺术字体的透明窗体等。针对这种需求，公司采用()最为灵活。
- A. 桥接模式
B. 命令模式
C. 组合模式
D. 装饰模式
3. 【2010 年题 29】某银行系统采用 Factory Method 方法描述其不同账户之间的关系，设计出的类图如下所示。其中与 Factory Method 中的“Creator”角色相对应的类是()；与“Product”角色相对应的类是()。



- A. Bank B. Account C. Checking D. Savings
 A. Bank B. Account C. Checking D. Savings

4. 【2010 年题 49】某公司欲开发一套窗体图形界面类库。该类库需要包含若干预定义的窗格(Pane)对象,例如 TextPane、ListPane 等,窗格之间不允许直接引用。基于该类库的应用由一个包含一组窗格的窗口组成,并需要协调窗格之间的行为。基于该类库,在不引用窗格的前提下实现窗格之间的协作,应用开发者应采用()最为合适。
- A. 备忘录模式
 B. 中介者模式
 C. 访问者模式
 D. 迭代器模式
5. 【2010 年题 50】某公司开发一个文档编辑器,该编辑器允许在文档中直接嵌入图形对象,但开销很大。用户在系统设计之初提出编辑器在打开文档时必须十分迅速,可以暂时不显示当前页面以外的图形。针对这种需求,公司可以采用()避免同时创建这些图形对象。
- A. 代理模式
 B. 外观模式
 C. 桥接模式
 D. 组合模式
6. 【2011 年 25】某公司欲开发一门户网站,将公司的各个分公司及办事处信息进行整合。现决定采用 composite 设计模式来实现公司的组织结构关系,并设计了如图所示的 UML 类图。图中与 Composite 模式中的“Component”角色相对应的类是(1),与“Composite”角色相对应的类是(2)。

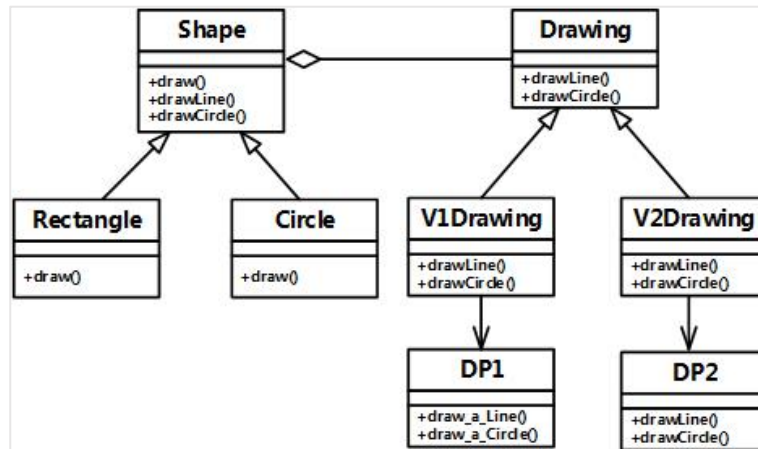


- (1) A. Company
 B. Finance Department
 C. HRDepartment
 D. ConcreteCompany
- (2) A. Company
 B. Finance Department
 C. HRDepartment
 D. ConcreteCompany

7. 【2011 年题 40】某软件公司正在设计一个通用的嵌入式数据处理平台, 需要支持各种数据处理芯片之间的数据传递与交换。该平台的核心功能之一要求能够屏蔽芯片之间的数据交互, 使其耦合松散, 并且可以独立改变芯片之间的交互过程。针对上述需求, 采用()最为合适。
- A. 抽象工厂模式
B. 策略模式
C. 中介者模式
D. 状态模式
8. 【2011 年题 41】某软件公司正在设计一个图像处理软件, 该软件需要支持用户在图像处理过程中的撤销和重做等动作, 为了实现该功能, 采用()最为合适。
- A. 单例模式
B. 命令模式
C. 访问者模式
D. 适配器模式
9. 【2011 年题 42】某互联网公司正在设计一套网络聊天系统, 为了限制用户在使用该系统时发表不恰当言论, 需要对聊天内容进行特定敏感词的过滤。针对上述功能需求, 采用_____能够灵活配置敏感词的过滤过程。
- A. 责任链模式
B. 工厂模式
C. 组合模式
D. 装饰模式
10. 【2012 年题 12】某软件公司欲设计一款图像处理软件, 帮助用户对拍摄的照片进行后期处理。在软件需求分析阶段, 公司的系统分析师识别出了如下 3 个关键需求。
- 图像处理软件需要记录用户在处理照片时所有动作, 并能够支持用户动作的撤销与重做等行为。
- 图像处理软件需要根据当前正在处理的照片的不同特征选择合适的处理操作, 处理操作与照片特征之间具有较为复杂的逻辑关系。
- 图像处理软件需要封装各种图像处理算法, 用户能够根据需要灵活选择合适的处理算法; 软件还要支持高级用户根据一定的规则添加自定义处理算法。
- 在系统设计阶段, 公司的架构师决定采用设计模式满足上述关键需求中对系统灵活性与扩展性的要求。具体来说, 为了支持灵活的撤销与重做等行为, 采用()最为合适; 为了封装图像操作与照片特征之间的复杂逻辑关系, 采用()最为合适; 为了实现图像处理算法的灵活选择与替换, 采用()最为合适。
- A. 工厂模式 B. 责任链模式 C. 中介者模式 D. 命令模式
A. 状态模式 B. 适配器模式 C. 组合模式 D. 单例模式
A. 模板方法模式 B. 访问者模式 C. 策略模式 D. 观察者模式
11. 【2012 年题 13】若系统中的某子模块需要为其他模块提供访问不同数据库系统的功能, 这些数据库系统提供的访问接口有一定的差异, 但访问过程却都是相同的, 例如, 先连接数据库, 再打开数据库, 最后对数据进行查询。针对上述需求, 可以采用()设计模式抽象出相同的数据库访问过程, 该设计模式()。
- (1) A. 外观 B. 装饰 C. 桥接 D. 享元
(2) A. 可以动态、透明地给单个对象添加职责
B. 为子系统定义了一个高层接口, 这个接口使得这一子系统更加容易使用
C. 通过运用共享技术, 有效支持大量细粒度的对象

- D.将抽象部分与它的实现部分分离,使它们都可以独立地变化
12. 【2013 年题 24】某系统中的文本显示类(TextView)和图片显示类(PictureView)都继承了组件类(Component),分别显示文本和图片内容,现需要构造带有滚动条或者带有黑色边框,或者既有滚动条又有黑色边框的文本显示控件和图片显示控件,但希望最多只增加 3 个类。那么采用设计模式()可实现该需求,其优点是()。
- (1)A. 外观 B. 单体 C. 装饰 D. 模板方法
- (2)A. 比静态继承具有更大的灵活性
B. 提高已有功能的重复使用性
C. 可以将接口与实现相分离
D. 为复杂系统提供了简单接口
13. 【2014 年题 22】一组对象以定义良好但是复杂的方式进行通信,产生的相互依赖关系结构混乱且难以理解。采用()模式,用一个特定对象来封装一系列的对象交互,从而使各对象不需要显式地相互引用,使其耦合松散,而且可以独立地改变它们之间的交互。
- A. 解释器(Interpreter)
B. 策略(Strategy)
C. 中介者(Mediator)
D. 迭代器(Iterator)
14. 【2014 年题 23】某广告公司的宣传产品有宣传册、文章、传单等多种形式,宣传产品的出版方式包括纸质方式、CD、DVD、在线发布等。现要求为该广告公司设计一个管理这些宣传产品的应用,采用()设计模式较为合适,该模式()。
- (1)A. Decorator B. Adapter C. Bridge D. Facade
- (2)A. 将一系列复杂的类包装成一个简单的封闭接口
B. 将抽象部分与它的实现部分分离,使它们都可以独立地变化
C. 可在不影响其他对象的情况下,以动态、透明的方式给单个对象添加职责
D. 将一个接口转换为客户希望的另一个接口
15. 【2014 年题 33】某公司欲开发一个用于分布式登录的服务端程序,使用面向连接的 TCP 协议并发地处理多客户端登录请求。用户要求该服务端程序运行在 Linux、Solaris 和 WindowsNT 等多种操作系统平台之上,而不同的操作系统的相关 API 函数和数据都有所不同。针对这种情况,公司的架构师决定采用“包装器外观(Wrapper Facade)”架构模式解决操作系统的差异问题。具体来说,服务端程序应该在包装器外观的实例上调用需要的方法,然后将请求和请求的参数发送给(),调用成功后将结果返回。使用该模式()。
- (1)A. 客户端程序
B. 操作系统 API 函数
C. TCP 协议 API 函数
D. 登录连接程序
- (2)A. 提高了底层代码访问的一致性,但降低了服务端程序的调用性能
B. 降低了服务端程序功能调用的灵活性,但提高了服务端程序的调用性能
C. 降低了服务端程序的可移植性,但提高了服务端程序的可维护性
D. 提高了系统的可复用性,但降低了系统的可配置性

16. 【2015 年题 24】某软件公司欲开发一个绘图软件，要求使用不同的绘图程序绘制不同的图形。在明确用户需求后，该公司的架构师决定采用 Bridge 模式实现该软件，并设计 UML 类图如下图所示。图中与 Bridge 模式中的“Abstraction”角色相对应的类是（ ），与“Implementor”角色相对应的类是（ ）。



- A.Shape B.Drawing C.Rectangle D.V2Drawing
- A.Shape B.Drawing C.Rectangle D.V2Drawing
17. 【2016 年题 36】设计模式基于面向对象技术，是人们在长期的开发实践中良好经验的结晶，提供了一个简单、统一的描述方法，使得人们可以复用这些软件设计办法、过程管理经验。按照设计模式的目的进行划分，现有的设计模式可以分为创建型、（ ）和行为型三种类型。其中（ ）属于创建型模式，（ ）属于行为型模式。（ ）模式可以将一个复杂的组件分成功能性抽象和内部实现两个独立的但又相关的继承层次结构，从而可以实现接口与实现分离。
- A.合成型 B.组合型 C.结构型 D.聚合型
- A.Adaptor B.Facade C.Command D.Singleton
- A.Decorator B.Composite C.Memento D.Builder
- A.Prototype B.Flyweight C.Adapter D.Bridge
18. 【2018 年题 36】设计模式描述了一个出现在特定设计语境中的设计再现问题，并为它的解决方案提供了一个经过充分验证的通用方案，不同的设计模式关注解决不同的问题。例如，抽象工厂模式提供一个接口，可以创建一系列相关或相互依赖的对象，而无需指定它们具体的类，它是一种（54）模式；（55）模式将类的抽象部分和它的实现部分分离出来，使它们可以独立变化，它属于（56）模式；（57）模式将一个请求封装为一个对象，从而可用不同的请求对客户进行参数化，将请求排队或记录请求日志，支持可撤销的操作。
- （54）A. 组合型
B. 结构型
C. 行为型
D. 创建型
- （55）A. Bridge
B. Proxy
C. Prototype
D. Adapter
- （56）A. 组合型
B. 结构型
C. 行为型

- D. 创建型
- (57) A. Command
- B. Facade
- C. Memento
- D. Visitor

10.1.6 设计模式分类

1. 【2017 年题 38】按照设计模式的目的进行划分, 现有的设计模式可以分为三类。其中创建型模式通过采用抽象类所定义的接口, 封装了系统中对象如何创建、组合等信息, 其代表有()模式等; ()模式主要用于如何组合已有的类和对象以获得更大的结构, 其代表有 Adapter 模式等; ()模式主要用于对象之间的职责及其提供服务的分配方式, 其代表有()模式等。

A.Decorator	B.Flyweight	C.Command	D.Singleton
A.合成型	B.组合型	C.结构型	D.聚合型
A.行为型	B.交互型	C.耦合性	D.关联型
A.Prototype	B.Facade	C.Proxy	D.Visitor

10.2 设计模式及实现

10.2.6 Observer 模式

1. 【2009 年题 48】某软件公司承接了为某 workflow 语言开发解释器的工作。该 workflow 语言由多种活动节点构成, 具有类 XML 的语法结构。用户要求解释器工作时, 对每个活动节点进行一系列的处理, 包括执行活动、日志记录、调用外部应用程序等, 并且要求处理过程具有可扩展能力。针对这种需求, 公司采用()最为恰当。
- A. 适配器模式
- B. 迭代器模式
- C. 访问者模式
- D. 观察者模式