

考试复习

考试复习 2023/02/12

考试形式：闭卷考试

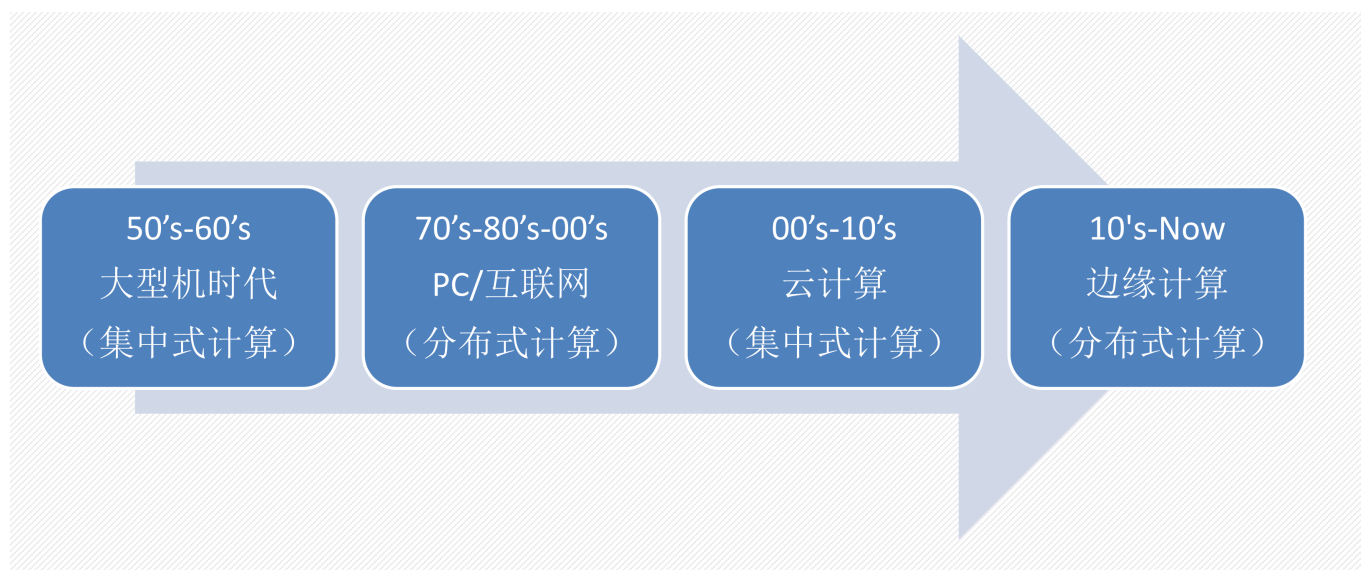
考试内容：名词解释、判断题、简答题、论述题。

重要内容：文档标红内容

分布式计算课程学习目标

- 了解什么是分布式系统，分布式系统有何特点
- 如何设计和描述一个分布式系统
- 有哪些重要的分布式应用
- 为了开发分布式应用，有哪些技术可以使用
- 掌握了这些知识，能够解决什么问题

天下大势，分久必合，合久必分



第一章

分布式系统的特征

- **分布式系统的定义**

分布式系统是一种由多个独立计算机组成的系统，它们共同完成一个任务或提供一种服务。分布式系统是通过网络进行通信，共享资源和协调任务完成情况，以实现高效的计算和数据处理能力。

- **分布式系统的目标**

1. 可扩展性
2. 高可用性
3. 负载均衡
4. 数据一致性
5. 安全性
6. 透明性

- **分布式系统的主要特征**

1. 分布式资源
2. 分布式数据
3. 分布式通信
4. 分布式协调
5. 可扩展性
6. 高可用性
7. 负载均衡
8. 数据一致性

- **分布式系统的发展趋势**

1. 云计算
2. 容器化
3. 微服务架构
4. 分布式存储
5. 机器学习
6. 区块链
7. 人工智能

- **分布式系统面临的挑战，每一个挑战中具体的问题是什么？**

1. 异构性：网络、计算机硬件、操作系统、编程语言和由不同开发者完成的软件实现中存在异构性
2. 开放性：
3. 安全性：有两个安全方面的挑战没有得到完美解决
 1. 拒绝服务攻击
 2. 移动代码的安全性

- 4. 可伸缩性:
 - 1. 控制性能损失
 - 2. 防止软件资源占用
 - 3. 避免性能瓶颈
- 5. 故障处理
- 6. 并发性
- 7. 透明性
- 8. 服务质量

讨论

- WWW技术是谁发明的？哪一年发明的？
World Wide Web (WWW) 技术是由英国计算机科学家**Tim Berners-Lee**在**1989年**发明的。
- WWW技术的重要性是什么？
 - 1. 信息共享
 - 2. 商业和经济
 - 3. 教育
 - 4. 社交和通信
 - 5. 总的来说，WWW技术改变了人们生活和工作的方式，并为全球范围内的人们提供了更多的信息和资源
- WWW技术的系统架构和主要功能是什么？
 - 1. HTML (HyperText Markup Language): 一种文档格式，用于创建网页。
 - 2. HTTP (HyperText Transfer Protocol): 一种网络协议，用于在网络上传输HTML 文档。
 - 3. URL (Uniform Resource Locator): 一种在网络上标识网页的方法。
 - 4. Web 服务器：用于存储和发布网页的计算机。
 - 5. Web 浏览器：用于浏览网络上的文档的应用程序。
 - 6. WWW 的核心功能是允许用户通过点击超链接 (Hyperlinks) 在不同的网页之间进行导航，从而访问网络上的信息。
- WWW技术中包含哪些主要的协议？
 - HTTP
 - HTML
 - URL
 - FTP
- WWW技术如何体现分布式计算的思想？

- WWW技术并不是一种分布式计算技术，但是WWW技术可以体现分布式计算的思想，主要体现在以下几点
 1. 数据分布
 2. 负载均衡
 3. 可扩展性
 4. 高可用性

第二章

系统模型

- 物理模型、体系结构模型、基础模型的定义和主要内容
 - 物理模型：物理模型是从计算机和所使用网络技术的特定细节中抽象出来的分布式系统底层硬件元素的表示
 - 体系结构模型：一个系统的提携结构是用独立指定的组件以及这些组件之间的关系来表示的结构
 - 基础模型：上面的各种系统模型完全不同，但具有一些基本特性。

- **重点：体系结构模型**

- 通信实体有哪几种？
- 通信范型：

通信实体（什么在通信）		通信范型（它们怎样通信）		
面向系统的实体	面向问题的实体	进程间通信	远程调用	间接通信
结点	对象	消息传递	请求 - 应答	组通信
进程	组件	套接字	RPC	发布 - 订阅
	Web 服务	多播	RMI	消息队列
				元组空间
				DSM

图 2-2 通信实体和通信范型

- 角色和责任：
 - C/S架构
 - P2P架构
- 放置
- 2.3.2节中
 - 分层体系结构：在分层方法中，一个复杂的系统被分成若干层，每层利用下层提供的服务。就分布式系统而言，这等同于把服务垂直组织成服务层。
 - 层次化体系结构
 - 与分层体系结构是互补的
 - 分层是垂直的，层次化是水平方向上的划分

- 层次化把不同的系统功能放在合适的服务器上

讨论

- 给出一个互联网应用，说明其中的异构性是如何体现的。

电子邮件是一种互联网应用，它涵盖了许多不同的系统、设备和技术。例如，电子邮件可以通过Web浏览器、桌面客户端、手机应用程序等不同的方式收发。这些不同的方式可能使用不同的协议，如IMAP、POP3、SMTP等。

总的来说，电子邮件中的异构性表现为不同的系统、设备、技术和标准之间的差异，以及如何通过不同的接口和协议进行集成。
- C/S架构和P2P架构的优缺点是什么？
 - C/S架构
 - 优点
 - 易于管理
 - 安全性高
 - 资源访问方便
 - 缺点
 - 单点故障
 - 流量集中
 - 效率低
 - P2P架构
 - 优点
 - 无单点故障
 - 资源分配均衡
 - 效率高
 - 缺点
 - 安全性差
 - 流量分散
 - 管理困难
 - 节点不稳定
 - 带宽不足

第三章

网络与网际互联

- 3.1节是重点：了解分布式系统对基础网络提出的以下要求：

- 新能
- 可伸缩性
- 可靠性
- 安全性
- 移动性
- 服务质量
- 组播

讨论

- 互联网非常大，任何路由器都无法容纳所有网段的路由信息，那么互联网路由方案中采用了哪些方法和技术来处理这个问题呢？
 1. 路由器分层
 2. 路由器信息限制
 3. 默认路由
 4. 分组路由
- 我们能确保在互联网中不会有两台计算机使用同一个IP地址吗？
不能确保，NAT技术
- 如何在互联网中提供服务质量保障QoS？
 1. 资源预留
 2. 路径选择
 3. 连接设置
 4. 流量控制
 5. 网络计算

第四章

进程间通信

- 同步与异步通信
- 套接字的作用
- TCP和UDP如何利用套接字进行通信
- TCP和UDP在JAVA中的实现
- 外部数据表示的原因和三个主要方法
- XML的主要特点
- 远程对象引用的定义和用途
- 覆盖网络(Overlay Network)的定义、主要功能和优点

讨论

- 一台服务器同时能够建立的TCP连接数量是多少？
一台服务器同时能够建立的TCP连接数量取决于多个因素，例如服务器的配置、硬件、操作系统、网络带宽等。（65535个）
通常来说，一台服务器的TCP连接数量可以在几千到几万不等。但实际的数量也可能受到网络延迟、带宽限制等因素的影响。
- 当访问一台服务器的客户端数量持续增加，我们如何对服务器进行扩容 以满足服务需求？
 1. 使用资源更多的服务器进行代替
 2. 同时增加多台服务器来提供服务
 3. 把一些客户资源暂停处理，先去处理最紧急的任务
- 网络层组播的主要问题是什么？应用层组播能解决这些问题吗？
 1. 路由选择问题
 2. 带宽浪费问题
 3. 路由器负载问题

应用层组播可以解决这些问题的一部分。应用层组播在应用程序上实现，因此可以更灵活地选择路径和分配带宽。此外，应用层组播可以通过对数据进行压缩和优化来减少带宽浪费。但是，应用层组播也有一些缺点，例如需要应用程序开发者实现，需要更多的网络带宽等。

第五章

远程调用

- 请求应答协议的工作原理
 1. 客户端发送请求
 2. 服务端接收请求
 3. 服务器响应请求
 4. 客户端接收应答
- RPC的定义
是一种通过网络从远程计算机程序上请求服务，而不需要了解底层网络技术的协议
- 接口编程的定义、优点
接口编程是一种编程技术，它通过定义一个接口来实现多种类型对象之间的交互。
优点：
 1. 增强灵活性

2. 提高代码复用
3. 简化编程
4. 提高代码的可测试性

- RMI的定义

RMI (Remote Method Invocation) 是 Java 语言提供的一种远程方法调用技术。它允许一个 Java 对象在另一个 Java 虚拟机（甚至可能在不同的物理机上）中运行，并与本地 Java 对象一样通过方法调用与远程对象进行通信。通过 RMI，Java 程序员可以编写分布式程序，即使是在不同的物理机上运行的 Java 虚拟机可以相互通信，共享数据和处理。

讨论

- 在RRA协议中，服务器应该保留未被确认的应答数据多久？为了接收到确认，服务器应该重复发送应答吗？

500ms或者倍数

应该重复发送应答

- 在RMI中使用回调，服务器需要有客户回调对象的最新列表，但是客户并不总是能在它退出之前通知服务器，这会导致服务器中的列表不正确。如何解决这个问题？

服务器可以实时监听客户状态，当客户退出时就可以更新列表

第六章

间接通信

- 间接通信的定义

在分布式系统中实体通过中介者进行通信，没有发送者和接收者（们）之间的直接耦合的通信。

- 空间解耦和时间解耦

	时间耦合	时间解耦
空间耦合	性质：与一个或一些给定的接收者直接通信；接收者（们）必须在那个时刻存在 例子：消息传递、远程调用（参见第4章和第5章）	性质：与一个或一些给定的接收者直接通信；发送者（们）和接收者（们）可以有各自独立的生命周期 例子：参见练习6.3
空间解耦	性质：发送者不需要知道接收者（们）的身份；接收者（们）必须在那个时刻存在 例子：IP 组播（参见第4章）	性质：发送者不需要知道接收者（们）的身份；发送者（们）和接收者（们）有独立的生命周期 例子：本章讲解的大多数间接通信范型

图 6-1 分布式系统中空间和时间的耦合

- 组通信的工作原理和主要特点？

1. 工作原理：通过组播地址将数据发送到一个计算机组

- 2. 主要特点:
 - 1. 广播行
 - 2. 简单性
 - 3. 效率
 - 4. 可靠性
- 消息队列的工作原理和主要特点?
 - 1. 工作原理: 通过一个中间代理, 通常称为消息代理或消息中间件, 来管理消息的发送和接收。
 - 2. 主要特点:
 - 1. 异步通信
 - 2. 高可用性
 - 3. 可扩展性
 - 4. 可靠性

讨论

- 你能设计一个系统, 满足空间耦合和时间解耦吗?
设计一个快递盒子, 只有快递员可以放置快递, 只有用户可以取快递。在空间上只有通过快递盒子才能放/取快递, 并且在时间上不要求放置快递的同时取快递, 实现了空间的耦合和时间的解耦。
- 你能设计一个使用消息队列的应用, 来解决你在工作中或者生活中碰到 的问题吗?
工厂中的流水线就是一种消息队列。

第七章

操作系统支持

- 网络操作系统与分布式操作系统的区别?
主要区别在于它们的架构和目标
 - 网络操作系统: 中心化, 并且以C/S架构为主
 - 分布式操作系统: 去中心化, 高可用性扩展性, C/S或P2P为主
- 为什么分布式操作系统没有得到广泛的应用?
 - 1. 用户在现有操作系统上已经进行了进行了大量的投资
 - 2. 用户更希望自己管理自己的计算资源, 而分布式操作系统使得用户无法控制自己的计算资源

- 中间件+网络操作系统的优势是什么？

中间件和网络操作系统的结合为**自治性需求**和**透明的网络资源访问**之间提供了一个可接受的平衡

- 多线程服务器的三种体系结构的特点

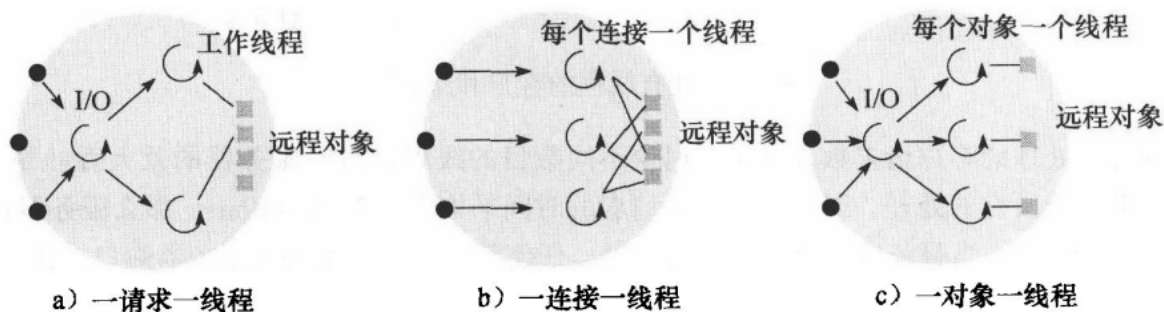


图 7-6 几种服务器线程体系结构（参见图 7-5）

- 整体内核和微内核的对比

讨论

- 系统虚拟化解决的最主要问题是什么？

1. 资源利用率
2. 灵活性
3. 安全性
4. 兼容性
5. 灵活的迁移和扩展

- 系统虚拟化主流的软件有哪些？

- VMware
- Virtual Box
- Hyper-V
- KVM
- Xen

- 系统虚拟化的优缺点是什么？

优点：

1. 资源利用率高
2. 软件隔离
3. 可以灵活部署
4. 容易维护

缺点：

1. 系统性能有损失

2. 系统复杂度高
3. 系统容易收到攻击，安全性低

第八章

分布式对象和组件

- 分布对象中间件的定义和优点

分布式对象中间件是一种技术，用于管理和协调分布式环境中的对象，并使它们可以轻松地互相通信和协作。

- 基于组件的中间件主要解决了分布式中间件中的哪些问题？

开发基于组件的中间件是为了克服使用分布式对象中间件开发应用中所遇到的一系列限制，包括：

- 隐式依赖
- 编程复杂性

- 组件的定义和主要特点

组件是一种独立、可重用的软件单元，具有定义输入和输出的接口，它可以被组转或集成到其他系统中，以提供特定的功能。

主要特点：

1. 可重用
2. 可维护
3. 可组合
4. 独立性
5. 可扩展性

讨论

- EJB是否适合大型多人在线游戏？

不适合，主要有以下原因：

1. 性能较慢
2. EJB架构对游戏开发有不少限制
3. EJB技术不够灵活，可扩展性不够

- 在日常工作或者生活场景中，什么样的应用适合使用EJB？什么样的应用 适合使用Fractal？

EJB：

适合用于开发企业级应用程序。

典型的应用场景有：

- 企业资源计划系统

- 企业客户关系管理系统
- 企业内部管理系统

第九章

Web Service

- Web Service的定义和工作原理

Web Service是一种基于互联网的、利用 XML、SOAP、WSDL 和 UDDI 等标准的应用程序组件通信技术。

工作原理：

- 客户端发送请求
 - Web Service解析请求
 - Web Service处理请求
 - Web Service发送响应
 - 客户端解析响应
- Web Service和Web Server的区别
 - 功能：Web Server 主要负责提供网页内容；而 Web Service 主要是提供数据和应用程序的服务
 - 协议：Web Server 主要使用 HTTP 协议；Web Service 则使用 XML、SOAP、WSDL 和 UDDI 等标准通信
- URI/URL/UNR的关系与区别
 - 关系：URI 是一种抽象的资源标识符，包含了 URL 和 URN；URL 是 URI 的一种实现，代表了互联网上一个资源的位置；而 URN 则代表一个资源的名称，但并不代表该资源的具体位置。
 - 功能：URL 包含了关于资源如何被访问的信息，例如协议、主机地址、端口、路径等。它是互联网上访问资源的具体地址。而 URN 则代表一个资源的名称，但不包含关于该资源的具体位置的信息。
 - 应用：URL 通常用于网页、文件等互联网资源的地址，而 URN 则用于确定资源的名称，以保证资源的可识别性和可持久性。

讨论

- 现在要开发如下系统，可选择的技术方案包括：CORBA, EJB, Fractal, Web Service，请问选择哪种技术方案更合适？

1. 12306

该系统的主要特点是要求高并发和高可靠性，因此可以考虑使用 Web

Service 技术。该技术能够通过定义规范的 API 来支持分布式系统间的数据交互，并具有良好的稳定性。

2. 郑州大学网络空间安全学院职工考勤系统

郑州大学网络空间安全学院职工考勤系统：该系统主要涉及数据存储和访问，可以考虑使用 **EJB 技术**。该技术提供了强大的管理能力，可以方便地实现业务逻辑和数据管理，对于数据安全和管理的保障。

3. 携程（旅游服务APP）

郑州大学网络空间安全学院职工考勤系统：该系统主要涉及数据存储和访问，可以考虑使用 **EJB 技术**。该技术提供了强大的管理能力，可以方便地实现业务逻辑和数据管理，对于数据安全和管理的保障。

第十章

对等系统

- 对等系统的主要特点
 1. 去中心化
 2. 节点自主性
 3. 动态性
 4. 资源共享
 5. 高可用性
 6. 更低的成本

讨论

你认为Napster系统的主要问题是什么？如果让你改进 Napster系统以提高可靠性和效率，你会采取哪些措施？

主要问题是侵犯版权。

1. 对内容进行审查
2. 实现数字版权管理
3. 提高网络效率
4. 改进系统安全

第十二章

分布式文件系统

- 分布式文件系统的主要目标
通过将文件分布到多个节点，提高数据的可靠性，扩展存储容量，并且允许再多

个节点之间共享文件。

- 了解分布式文件系统的主要需求
 - 透明性
 - 并发文件更新
 - 文件复制
 - 硬件和操作系统异构性
 - 容错
 - 一致性
 - 安全性
 - 效率

讨论

- 谈谈你有哪些方法，能够进一步提高NFS系统的读写效率？
 - 减少网络带宽占用
 - 缓存计智
 - 多播技术
 - 适当提高块大小
 - 使用合适的网络协议
- 如果让你在windows server上开发一个NFS Server端软件，你认为该软件需要实现哪些功能？
 - 访问控制
 - 多客户端支持
 - 冗余和容错功能
 - 文件锁定
 - 性能优化
 - 管理工具

第十三章

名字服务

讨论

- 在一个DNS服务中，客户端可以使用迭代导航或者递归导航，你认为这两种方式哪个查询效率更高？哪个占用的DNS服务器资源更少？
 - 迭代效率最高

- 递归对DNS服务器资源占用更少

第十九章

移动计算与无处不在计算

- 了解一些基本概念：移动计算，无处不在计算，可穿戴计算，上下文敏感计算，易变系统，智能空间
 - 移动计算：研究日常物理世界中移动设备的连通问题
 - 无处不在计算（普适计算）：强调和环境融为一体的计算概念
 - 可穿戴计算
 - 上下文计算：计算机系统根据物理环境自动调整行为
 - 易变系统：计算机系统根据物理环境自动调整行为
 - 智能空间：是易变系统存在的环境

讨论

- 在日常生活中，请你举出一个例子，可以利用无处不在计算的思想来为用户提供服务，简述实现方法和实现技术
家庭自动化系统（智能家居）
实现方法：
 1. 在每个房间安装智能感应器和控制器，这些设备可以监测环境状况并执行相应的控制操作。
 2. 安装一个中央控制系统，连接所有房间的设备，并通过网络进行通信。
 3. 通过智能手机或平板电脑，用户可以随时远程控制家庭自动化系统中的设备。实现技术：
 4. 网络技术，如Wi-Fi，Zigbee等，用于连接设备和中央控制系统。
 5. 传感器技术，如红外感应器，温湿度传感器等，用于监测环境状况。
 6. 控制器技术，如智能插座，灯泡等，用于执行相应的控制操作。
 7. 移动应用技术，如iOS，Android等，用于提供用户控制界面。

第二十章

分布式多媒体系统

- 了解多媒体数据的主要特征
 - 连续性
 - 基于事件的

讨论

- 现在有一个视频网站，承诺为其VIP用户提供1080p的观看体验，请你谈谈 为了保证VIP用户的服务质量，需要采取哪些技术手段？
 - 使用容灾技术提高稳定性和可用性
 - 缓存技术
 - CDN
 - 多媒体转码技术，根据用户网络环境动态调节清晰度
 - 网络协议优化