



深圳大学
Shenzhen University

第一章

中间件概论

中间件技术 Middleware Technology

深圳大学计算机与软件学院

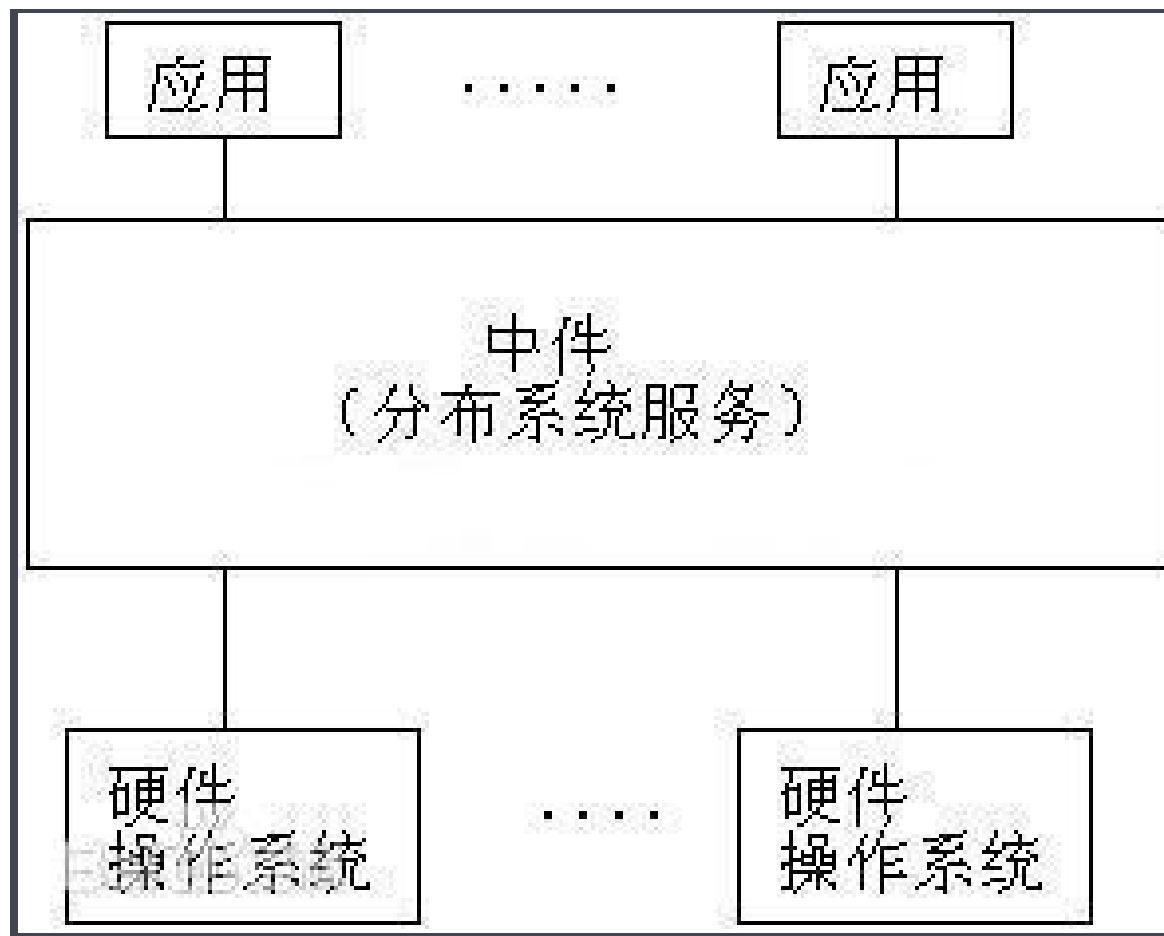


主要内容

- ◆ 1.1中间件的产生背景与历史
- ◆ 1.2中间件相关概念
- ◆ 1.3中间件的作用与特性
- ◆ 1.4中间件的分类
- ◆ 1.5中间件的应用与发展趋势

1.3 中间件的作用与特点

- 回顾：中间件是独立的系统级软件，连接操作系统层和应用程序层，将不同操作系统提供应用的接口标准化，协议统一化，屏蔽具体操作的细节。





1.3 中间件的作用与特点

■ 中间件的基本功能

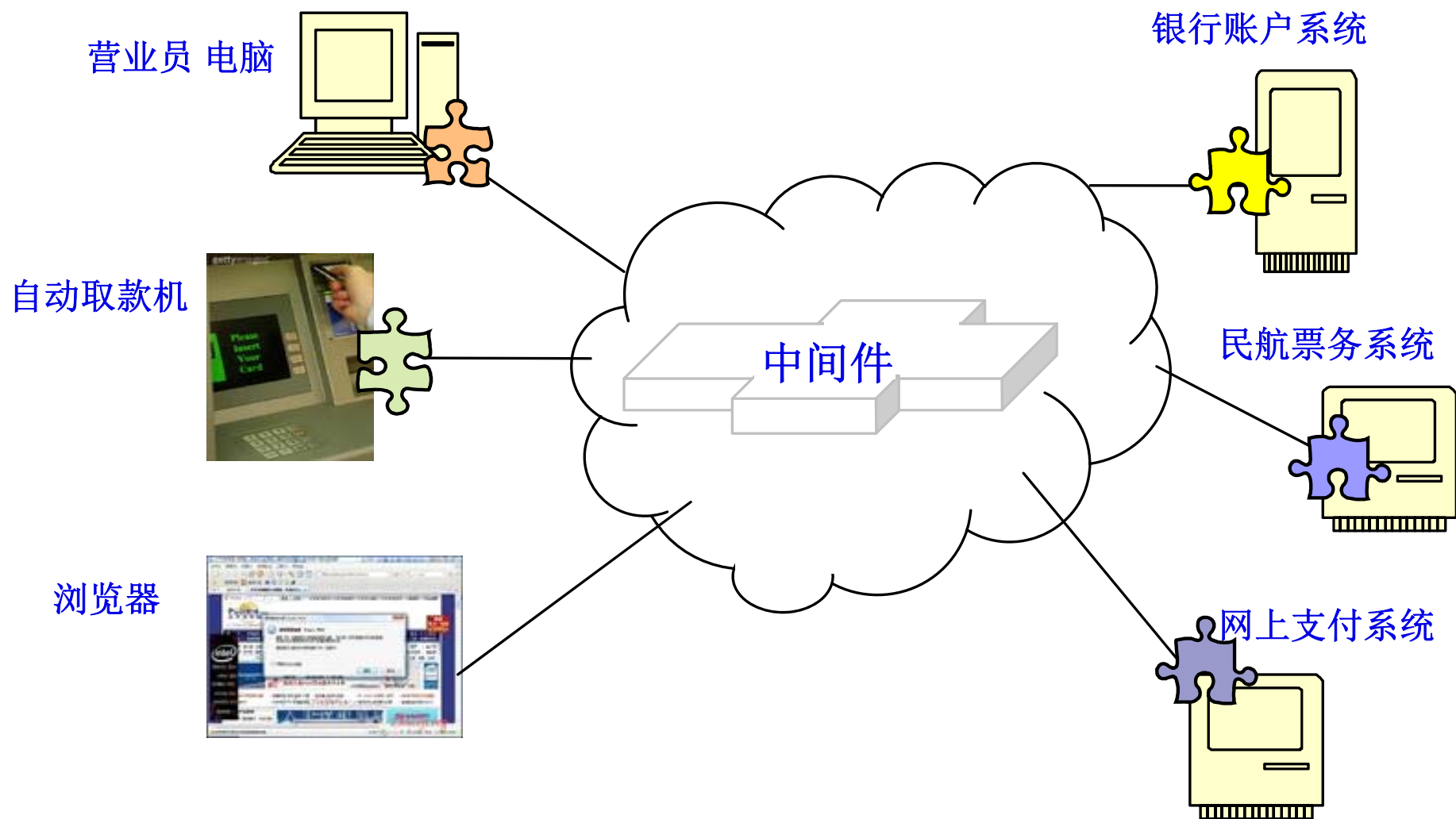
(1) **通信支持（最基本的功能）**：中间件为其所支持的应用软件提供平台化的运行环境，该环境屏蔽底层通信之间的接口差异，实现互操作。

(2) **应用支持**：服务上层应用，提供应用层不同服务之间的互操作机制。为上层应用开发提供统一的平台和运行环境，并封装不同操作系统提供API接口，向应用提供统一的标准接口，使应用的开发和运行与操作系统无关，实现其独立性。中间件松耦合的结构，标准的封装服务和接口，有效的互操作机制，给应用结构化和开发方法提供有力的支持。

(3) **公共服务**：是对应用软件中共性功能或约束的提取。将这些共性的功能或者约束分类实现，并支持复用，作为公共服务，提供给应用程序使用。通过提供标准、统一的公共服务，可减少上层应用的开发工作量，缩短应用的开发时间，并有助于提供应用软件的质量。

中间件的作用

中间件在具体系统中的作用：银行系统的例子





中间件的作用

中间件主要是用来
支持网络环境中软件实体之间的有效交互

“软件实体”是具有不同特征的软件模块
它们通常是具有不同粒度的软件构造模块

“有效”是指所支持的实体之间的交互
必须具有可靠、安全、快速等特点

中间件的作用

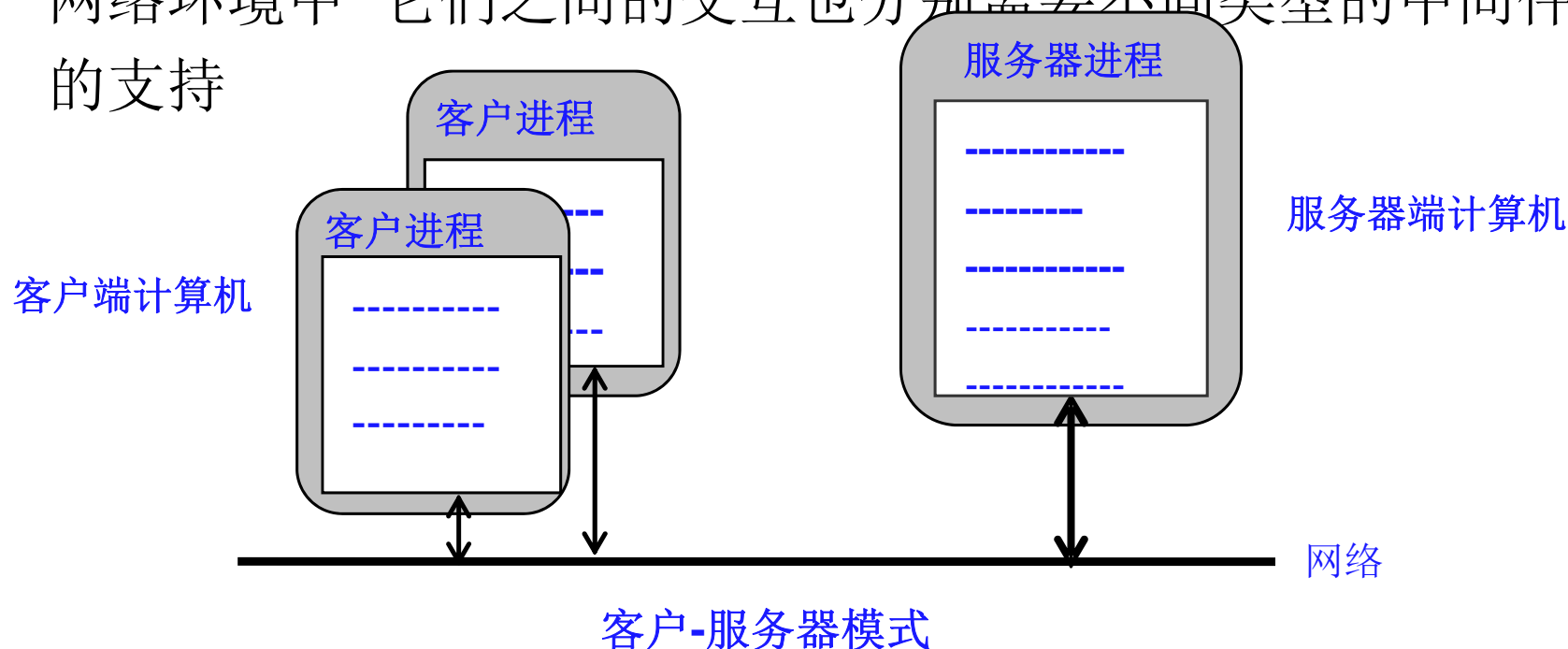
1.支持软件实体的交互模式，起桥梁作用，实现异构应用资源之间的互联互通。

不同特征的模块：过程、对象、构件与服务等等

它们不仅是构造软件系统的不同粒度的模块

也通常在运行阶段表现出不同的性质

网络环境中 它们之间的交互也分别需要不同类型的中间件的支持





中间件的作用

■ 过程（**Procedure**）

最基本的软件模块

是大型程序从混沌向结构化走出的第一步

每个过程都有具体的调用格式

在 C/C++ 中，该格式用头文件来说明

■ 对象（**Object**）

方法与数据的封装体

是类（**Class**）的实例

其中的方法与过程有直接的对应关系

在一些基于对象的系统中，对象仅仅在编程阶段存在，编译器将对象映射为传统的过程，这样，在运行阶段不再保持对象实体了



中间件的作用

■ 构件（**Component**）

更大粒度的构造模块

通常在运行阶段保持构件形态

通常构件是由一个或多个类组成的实体

也可以直接由一个或多个过程组成

■ 服务（**Service**）

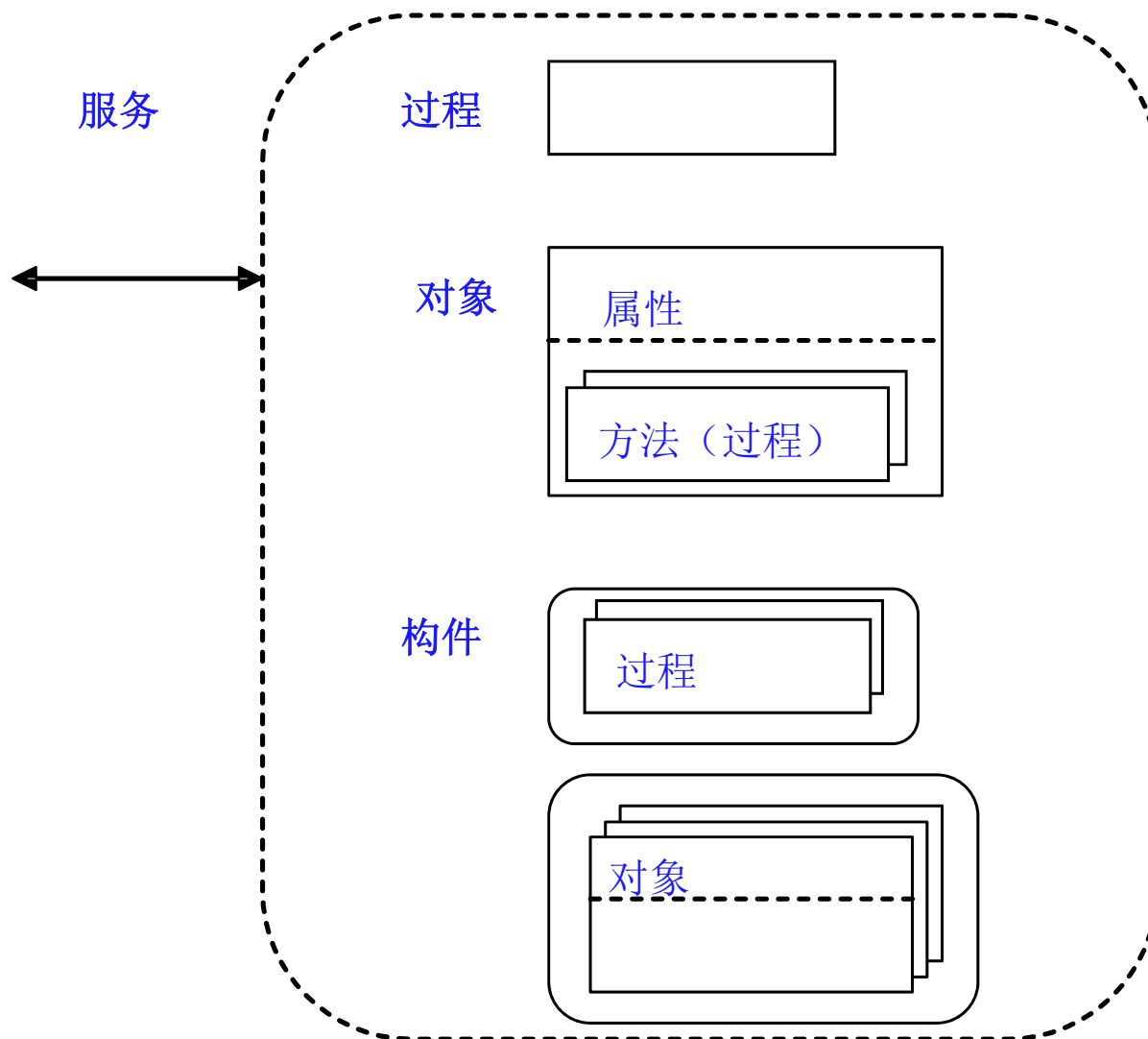
更松散的软件实体

服务强调的是软件实体的外在表现

其内在实现则也是由某个构件、某个对象、甚至某个过程完成

中间件的作用

过程、对象、构件与服务之间的关系





中间件的作用

不同构造模块出现的时间不同，后者直接发展了前者

后者的实现中包含了前者的模块

只是强调的重点有所改变

但后者不是替代了前者：

尽管后续技术适合新的热点

以前的技术仍然有其适合的应用场景

这种现象 构成了软件系统构造方式的多样性

也造成了中间件产品的多样性

不同构造模块在网络环境中的交互模式是不同的：

接口的定义方式、交互的协议等等都不同

这就形成了分别 支持远程过程调用

支持对象互操作

支持构件技术

支持服务概念等不同种类的中间件



中间件的作用

2.起通道和过滤网的作用，支持软件实体的交互质量。

可靠性保障：提供通道功能，可保证应用系统数据的可靠传输，避免基于网络的并发操作发生故障与冲突时产生的危害，支持操作取消，支持重新提交操作。

安全性保障：保证网络中软件间好的交互的同时，防止恶意攻击，起过滤网作用，可有力保护关键信息的安全。

高效性保障：高效地管理资源，避免部分节点饥饿、部分节点超负荷，提高容错能力，负载共享。



中间件的作用

3.共性代码库.

将核心业务逻辑代码之外的与网络环境相关的代码 剥离出来凝练为共性的代码库。

4.集成开发框架和部署平台

是分布式应用的集成开发框架，促成企业分布式应用的有机集成；作为平台，有效实现分布应用的分阶段顺利部署。



中间件的特点

■ 中间件的优势：

| 企业应用方面 | 应用程序方面 |
|-------------------------|----------------------------------|
| 缩短应用开发周期 | 中间件支持分布式计算，提供异构环境的透明性的应用或服务的交互功能 |
| 减少项目开发风险，有效地保证应用系统的质量 | 支持标准协议和接口，方便应用集成，使应用程序有良好的扩展性 |
| 降低初期资金及运行费用的投入，节省开发费用 | 中间件有效地保证应用系统的质量，应用程序有良好的可靠性和可用性 |
| 运行于多种硬件和OS平台，便于系统的更新与维护 | 应用程序独立于运行平台提供的网络通信服务 |



中间件的特点

■ 中间件的局限性：

- (1) 中间件所应遵循的一些原则离实际还有很大距离
- (2) 中间件之间难互操作：多数流行的中间件服务使用专有的API和专有的协议, 不同厂家的实现很难互操作
- (3) 对可移植性的支持不足：有些中间件服务只提供某些平台的实现, 限制了应用在异构系统之间的移植
- (4) 技术发展风险：应用开发者在中间件服务之上建立自己的应用还要承担相当大的风险, 随着技术的发展他们往往还需重写他们的系统
- (5) 设计选择困难：应用开发者需面临许多艰难的设计选择。过分追求中间件功能的全面性, 过分强调选择国外产品, 不能正确认识中间件的作用。



1.4中间件的分类

底层中间件： 用于支撑单个应用系统或解决一类问题。

- 基于对象组件模型的中间件（面向对象中间件）
- 数据访问中间件
- 远程过程调用中间件（过程式中间件）
- 交易中间件（事务式中间件，TPM）
- 消息中间件（面向消息的中间件，MOM）
- 安全中间件
- RFID中间件
- 移动中间件



高层中间件：用于系统整合

- 企业应用集成中间件（EAIM）
- 工作流中间件
- 门户中间件
- 应用服务器中间件(WASM)



面向对象中间件

- 在分布、异构的网络计算环境中，将各种分布对象有机地结合在一起，完成系统的快速集成。实现对象重用，又称分布对象中间件，简称对象中间件。对象中间件向应用层提供多种形式的通信服务，使得上层应用在事务处理、分布式数据访问、对象处理等方面简单易行。
- 使用对象请求代理（**ORB**）通信机制

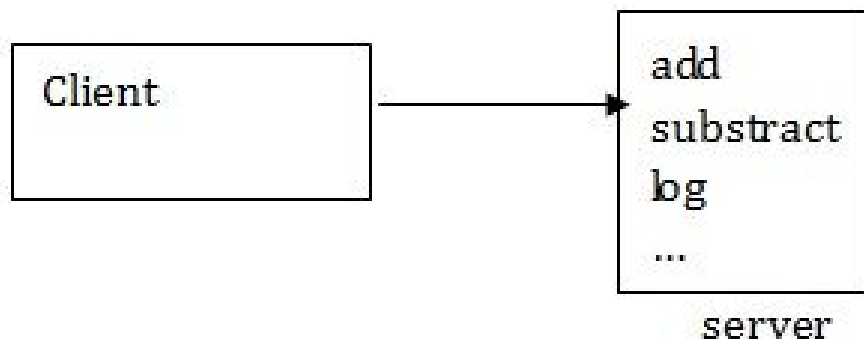


数据访问中间件**Data Access Middleware**

- 数据访问中间件，是连接应用程序和数据库的软件，可以用来实现应用对来自不同厂商数据库的访问，数据访问中间件有直接访问和数据库复制两种形式。
- 是随着数据库技术的发展而逐渐兴起的,也是其中发展较早的一类
- 例：ODBC和JDBC属于数据访问中间件
- 开放数据库连接（Open Database Connectivity，ODBC）是微软公司开放服务结构（WOSA，Windows Open Services Architecture）中有关数据库的一个组成部分，它建立了一组规范，并提供了一组对数据库访问的标准API
- JDBC（Java Data Base Connectivity,java数据库连接）是一种用于执行SQL语句的Java API，可以为多种关系数据库提供统一访问，它由一组用Java语言编写的类和接口组成

过程式中间件

- 又称远程过程调用中间件，使用RPC通信机制实现远程计算服务。从逻辑上分为客户端和服务端两部分。服务器端提供基于过程的服务，客户端调用服务。
- 客户端和服务端既可以运行在同一计算机上，也可以运行在不同的计算机上，甚至客户端和服务端底层的操作系统也可以不同。
- 客户端和服务端之间的通信可以使用同步通信，也可以采用线程式异步调用。
- 程序员方便地编写客户端应用程序调用位于服务器上的过程，这种调用通过网络来实现。





RFID中间件

- RFID中间件 拓展了基础中间件的核心设施和特性，将企业级中间件技术延伸到RFID领域，是RFID产业链的关键共性技术。RFID中间件屏蔽了RFID设备的多样性和复杂性，能够为后台业务系统提供强大的支撑，从而驱动更广泛、更丰富的 RFID 应用。
- RFID中间件技术重点研究的内容包括：并发访问技术、目录服务及定位技术、数据及设备监控技术、远程数据访问、安全和集成技术、进程及会话管理技术等。



移动中间件

- 移动中间件(mobile middleware)用来连接不同的移动应用、程序和系统。
- 移动中间件技术重点研究的内容包括：接入管理、多协议接入网关、认证服务、连接管理、同步/ 异步数据传递服务、安全管理、内容服务管理等。
- 例如MAStudio移动中间件，基于其框架可快速开发手机办公软件。iMAG移动中间件，基于其平台可快速开发移动应用。



交易中间件（事务式中间件）

- 交易中间件是一个联机事务处理平台软件，又称为事务处理管理程序，专门针对联机交易处理系统而设计的，交易中间件是一组程序模块，可减少开发联机交易处理系统所需的编程量，快速建立三层结构的联机事务处理应用。
- 主要功能是提供联机事务处理所需要的通信、并发访问控制、事务控制、资源管理、安全管理、负载平衡、故障恢复和其他必要的服务。
- 支持大量客户进程的并发访问，具有极强的扩展性、可靠性高等特点，主要应用于电信、金融、飞机订票系统、证券等拥有大量客户的领域。



交易中间件产品

- 联机交易处理系统需要处理大量并发进程，涉及到操作系统、文件系统、编程语言、数据通信、数据库系统、系统管理和应用软件，是一个相当艰巨的任务，可以通过采用一个交易中间件来简化。
- 交易中间件种类较多，如BEA的TUXEDO、IBM的CICS、NCR的Top End等，但最具代表性的产品当属BEA的TUXEDO和IBM的CICS，两者占居了大部分市场份额。



面向消息的中间件MOM

- 简称为消息中间件，是一类以消息为载体进行通信的中间件，利用高效可靠的消息机制来实现不同应用间大量的数据交换。
- 按其通信模型的不同，消息中间件的通信模型有两类：**消息队列和消息传递**。通过这两种消息模型，不同应用之间的通信和网络的复杂性脱离，摆脱对不同通信协议的依赖，可以在复杂的网络环境中高可靠、高效率的实现安全的异步通信。
- 消息中间件的非直接连接，支持多种通信规程，达到多个系统之间的数据的共享和同步。



消息中间件 产品

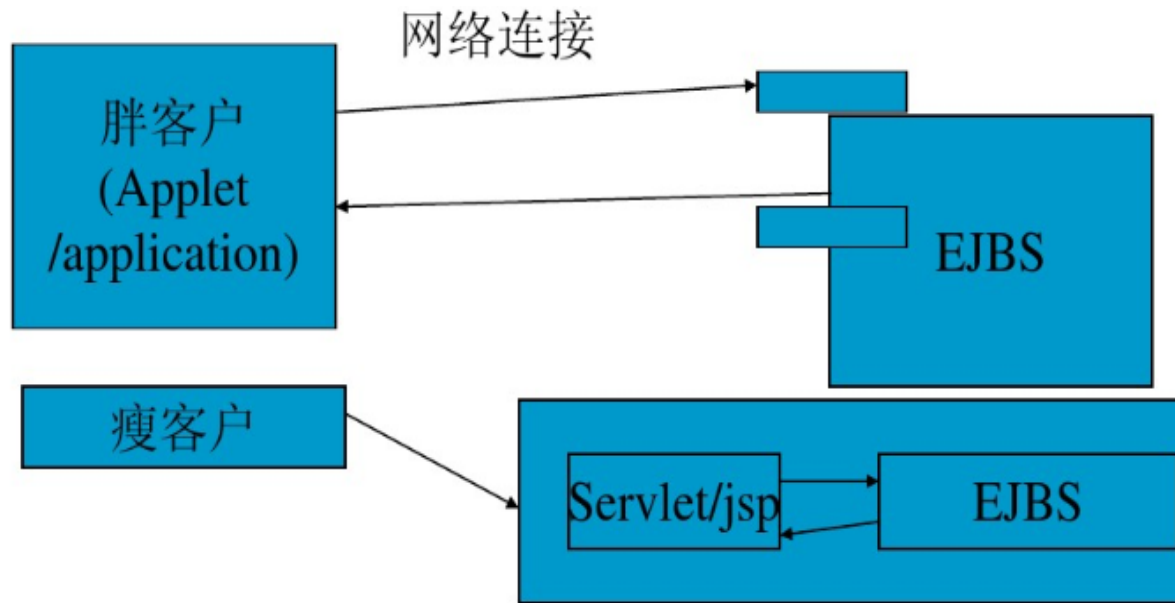
- “消息”是指一个用户可定义的数据结构，包含了应用间传递的数据，一个消息可以包括图像、数据库更新记录、电子邮件等。
- 主要产品有IBM的MQseries、BEA的Message Q、BMC Software的Patrol、Digital Equipment的Message等，但最具代表性的产品当属IBM的MQseries。



应用服务器中间件

- 又称为“应用服务器软件平台”，是当代软件的基础设施，是最为重要的一类中间件。应用服务器能把不同的应用软件作为构件整合到一个协同工作的环境里，并为应用提供了名字、事务、安全、消息、数据访问等服务，此外他还提供给用构件的研发、部署、运行及管理功能。
- 基于应用服务器的应用系统架构有三大流派：基于CORBA、基于J2EE和基于.Net。
- J2EE架构是目前产业界基于应用服务器中间件的web应用系统架构的主流标准。
- Tomcat (Apache), TomEE (Apache), Jboss (RedHat), Weblogic (BEA), WebSphere (IBM) 等等

应用服务器中间件





1.5 中间件技术的应用与发展趋势

- 中间件技术为跨平台的应用开发、异构网络环境的应用，软件复用等提供了有效的解决之道。应用：
 - （1）中间件与电子商务的整合
 - Internet是电子商务发展的基础，商户通过它，把商业扩展到能到达的任意地点。使用B/S模式传输和处理大量的信息。
 - （2）中间件在B/S模式中的架构
 - 中间件在B/S模式下起到了功能层的作用。当用户从WEB界面向服务器提交了数据请求或者应用请求时，功能层负责将这些请求分类为数据或应用请求，再向数据库发出数据交换申请。数据库对请求进行筛选处理之后，再将所需的数据通过功能层传递回到用户端。单一用户可以进行点对面的操作，无需通过其他软件进行数据转换。



中间件技术的发展趋势

- 中间件的发展需要规范化
- 应用需求导致技术格局多样化，随着网络化的发展趋势，应用软件需要的支持机制越来越多，中间件会变广变厚
- 中间件也向构件化发展，为上层应用的结构设计和部署提供有效的支持，并为解决软件复用问题提供支持
- 底层中间件持续走稳，越来越多地向传统操作系统层渗透，向平台化发展
- 高层中间件已成为市场新宠，新技术与新产品的推出速度放缓
- 实用化是重要的发展趋势
- 正在呈现出业务化、服务化、一体化、虚拟化等诸多新的重要发展趋势。



思考与讨论

- 结合本节主要内容与自己的专业知识背景,谈谈自己对中间件分类、应用、前景、兴趣等相关方面的观点.
- 搜索、分析和思考不同类型中间件的区别。

课后兴趣作业

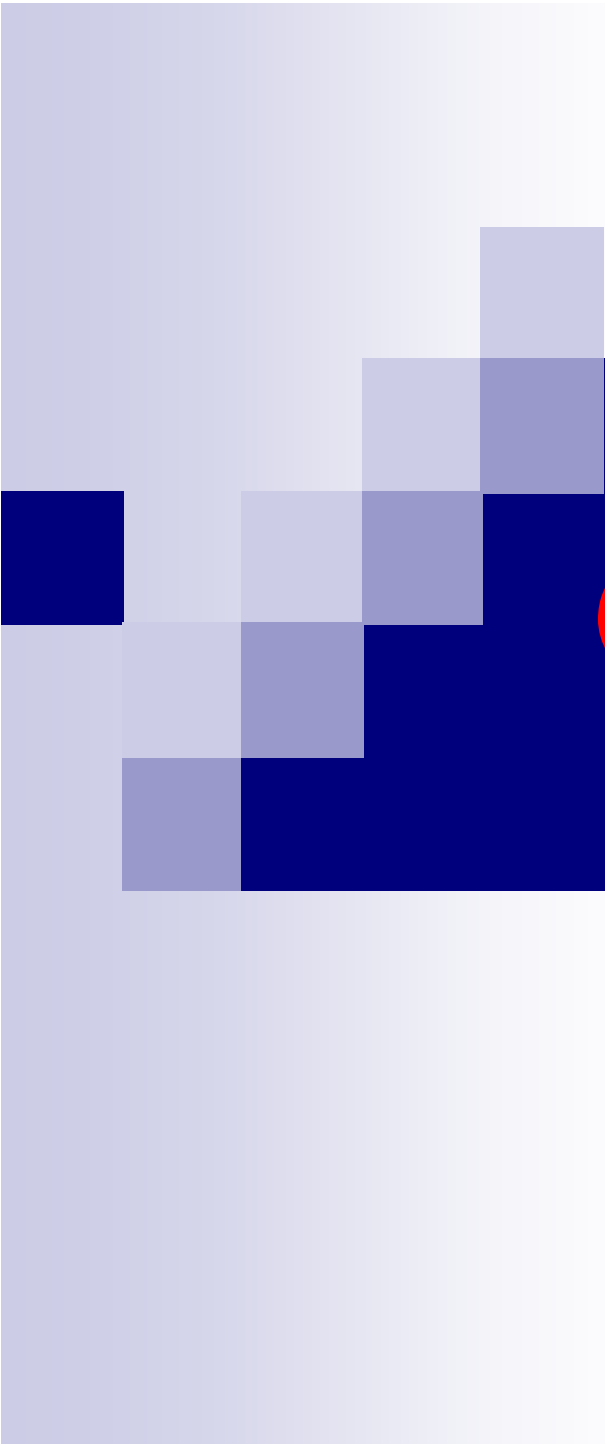
- 搜集整理不同类型中间件应用案例, 尝试模仿使用。



兴趣阅读资料

- 中间件技术现状及发展

https://www.csdn.net/gather_23/MtjaEg3sNjgyNjItYmxvZw00000000000.html.



Q & 谢谢