<https://searchapparchitecture.techtarget.com/definition/middleware>

中间件

中间件是将客户端生成的基于网络的请求连接到客户端正在请求的后端数据的软件。它是用于“胶合”单独的、通常是复杂的和已经存在的程序的软件的通用术语。

中间件程序包含在本地软件和云服务中，它们可以根据用例独立使用或一起使用。尽管云提供商将中间件捆绑到云服务套件中，例如中间件即服务（MWaaS）或集成PaaS（iPaaS），但行业研究人员注意到，许多企业仍然选择适合其特定需求的独立中间件产品。

为什么叫中间件？

通过基于网络的交互，客户端或请求程序可以发出请求。该客户端通常是驻留在前端（用户与软件进行交互）的应用程序。诸如数据库，消息队列，NoSQL 数据存储和文件服务器之类的资源通常被称为后端的一部分。

中间件的名称源于以下事实：它是位于前端的客户端请求和所请求的后端资源之间的软件。

中间件如何工作

本质上，所有基于网络的请求都是与后端数据进行交互的尝试。这些数据可能像显示图像或播放视频一样简单，也可能像银行交易历史一样复杂。

所请求的数据可以采用许多不同的形式，并且可以以多种方式存储，例如来自文件服务器，从消息队列中提取或持久保存在数据库中。中间件的作用是启用和简化对这些后端资源的访问。

为什么要使用中间件？

中间件具有多种功能。首先，它管理与各种后端资源的连接。因此，中间件组件可能会创建一个连接池，以提供对流行的后端数据库的快速有效访问。它还可能会创建到消息队列和主题的连接。中间件软件也可能管理与基于云的资源的连接，例如Amazon Simple Storage Service（S3）。

其次，中间件软件具有根据客户端的请求实施逻辑的能力。例如，中间件组件可能会认识到发出给定请求的客户端浏览器将语言标头设置为英语，因此，可能会对它对后端的查询进行了调整，以返回仅基于英语的结果。或者，也许服务器可以根据客户端的IP地址来识别发出请求的客户端的地理位置，然后将优先考虑地理位置上接近的结果的数据返回给客户端。接受用户请求，执行逻辑然后自定义结果的能力是中间件软件执行的一项重要工作。

第三，中间件在并发处理，负载平衡甚至事务管理中起着重要作用。中间件软件通常具有垂直和水平扩展的能力，以帮助在多个服务器，虚拟机或云的可用性区域上分配传入的客户端请求。中间件软件还可以处理并发和事务管理问题，例如，当两个或多个客户端尝试同时访问或更新给定的后端资源时可能出现的问题。

最后，中间件在保护对后端资源的访问方面起着重要的作用。中间件软件具有挑战客户的能力，它既需要使用SSL之类的技术进行安全连接，又需要使用用户名和密码组合甚至数字证书进行身份验证。然后，此安全信息用于检查发出请求的用户是否有权访问所讨论的数据。如果权利得到确认，则使用安全且加密的连接将数据从中间件服务器发送到客户端。

中间件类别

通常，IT行业分析师（例如Gartner Inc.和Forrester Research）将中间件分为两类：企业集成中间件和平台中间件。

企业应用程序集成中间件使程序员能够创建业务应用程序，而不必为每个新应用程序定制集成。在这里，中间件可帮助软件和服务组件协同工作，为数据一致性和多企业或B2B集成提供一层功能。通常，集成中间件提供消息传递 服务，因此不同的应用程序可以使用消息传递框架进行通信，例如简单对象访问协议（SOAP），Web服务， 表示状态传输 （REST）或 JavaScript对象表示法 （JSON）。此类别中使用的其他中间件技术包括对象请求代理（ORB），数据表示技术（例如XML和JavaScript Object Notation（JSON）等。

企业可以购买单独的中间件产品或本地或基于云的应用程序集成套件。

平台中间件通过为应用程序逻辑提供运行时托管环境（例如容器）来支持软件开发和交付。它的主要组件是内存和企业应用程序服务器，以及Web服务器和内容管理。中间件包括Web服务器，应用程序服务器，内容管理系统以及支持应用程序开发和交付的类似工具。通常，嵌入式或外部通信中间件允许不同的通信工具一起工作。这些通信工具实现了应用程序和服务的交互。资源管理服务，例如Microsoft Azure资源管理器，在运行时托管应用程序逻辑，这是平台中间件的另一个关键功能。其他组件包括可信平台模块（TPM）和内存中数据网格（IMDG）。

平台中间件产品也可以作为特定的本地或云服务工具以及多工具套件提供。在云套件站点上，中间件即服务提供了一组集成的平台工具和运行时环境。

中间件的类型

中间件的例子很多，每个中间件的创建都是为了实现将应用程序以及Web和云服务连接在一起的特定功能。以下是一些使用最广泛的中间件类型。

消息中间件促进了分布式应用程序和服务之间的通信。

对象或ORB中间件使软件组件或对象能够跨分布式系统与程序（例如容器）进行通信和交互。

远程过程调用（RPC）中间件提供了一种协议，该协议允许程序从位于另一台计算机或网络上的另一个程序请求服务。

数据或数据库中间件可直接访问数据库并与之交互；它通常包括SQL数据库软件。

事务或事务中间件可通过事务过程监视来确保事务从一个阶段转移到下一阶段。

以内容为中心的中间件允许客户端对特定内容和摘要的请求并进行交付；它类似于Apache Kafka之类的发布/订阅中间件。

嵌入式中间件可促进嵌入式应用程序与实时操作系统之间的通信和集成。

中间件的演变

从2000年到大约2010年，中间件软件是应用程序服务器概念的代名词。但是，随着云计算的普及和微服务取代更多的基于SOA的单片式系统，中间件正在发生变化。

与其将完整的应用程序部署到中间件服务器，不如将其发展为创建较小的，基于微服务的应用程序，将这些微服务打包为具有连接到它们所需的后端服务所需的全部知识，然后在内部部署应用程序。轻量级的基于容器的系统，例如Docker。然后，将容器部署到基于云的服务。

在这种情况下，中间件将是容器内托管的微服务与运行容器的云计算基础架构的组合。这种安排是中间件当前发展的方向