

MIDDLEWARE PARA SISTEMAS DISTRIBUÍDOS CENTRADOS EM REDE

ANGELO HENRIQUE ALVES MUNIZ, ELINEY SABINO, MARCIO FRANÇA DA MOTTA JUNIOR, VICTOR ELIAS SILVA DE MORAIS, MARCELO DIAS PEDROSO, LUIZ CLAUDIO BARRETO.

RESUMO

Este trabalho tem como finalidade apresentar o software Middleware de uma forma sucinta, pois seu estudo no modo geral é extenso, entretanto, este presente trabalho tenta buscar um entendimento sobre o software, compartilhando informações pesquisadas na web através de estudos de saberes docentes, de forma resumida porém diretas do software, tendo um inicial histórico, seguido, tratando sobre o que é, o porquê da importância de seu uso, alguns de seus conceitos e por fim seus tipos mais populares.

Palavras Chaves: *Middleware, Sistemas, Rede*

ABSTRACT

This paper aims to present the Middleware software in a succinct way, since its study in the general way is extensive, however, this present work tries to seek an understanding about the software, sharing information researched on the web through studies of teacher knowledge, in a way summarized but straightforward software, having a historical beginning, followed, dealing with what is, why the importance of its use, some of its concepts and finally its most popular types.

Key words: *Middleware, Systems, Network*

INTRODUÇÃO

Middleware, em termo geral, conhecido como um software que serve para colar programas separados ou já existentes, tratando de solicitações, baseadas em rede geradas por um cliente no qual está solicitando. Todavia, tentaremos esclarecer seu conceito, seus tipos de middlewares, sua estrutura padrão, como o objetivo de conhecer melhor esse software no qual muitas vezes é informalmente chamado de “Encanamento”.

HISTÓRIA DE MIDDLEWARE

Numa sucinta história sobre middleware afirma BAKKEN (2013), que o termo middleware apareceu pela primeira vez no final de 1980 para descrever o software de gerenciamento de conexão de rede, mas não entrou em uso, generalizado até meados dos anos 1990, quando a tecnologia de rede tinha conseguido uma penetração visibilidade suficientes. Contudo, o tempo tinha evoluído para um conjunto de paradigmas e serviços oferecidos para ajudar a tornar mais fácil e mais gerenciável para construir

aplicações distribuídas. O termo foi associado principalmente com bancos de dados relacionais para muitos praticantes no mundo dos negócios através do início de 1990. Conceitos semelhantes ao middleware de hoje já passou sob os nomes de sistemas operacionais de rede, sistemas operacionais distribuídos e ambientes de computação distribuída.

MIDDLEWARE O QUE É?

Middleware é semelhante a um sistema no qual pode suportar outros programas aplicativos fornecendo interações controladas, assim evitando interferência entre cálculos e facilitando sua comunicação entre diversos computadores por meios de serviços em rede, isso nos propõe deduzir que o middleware nada mais é que uma camada que está situada entre os aplicativos e o sistemas operacionais ou seja um software multiuso, suas principais funções nada mais é do que simplificar desenvolvimentos de aplicativos encobrindo suas diferenças entre sistemas operacionais.

Segundo BAKKEN (2013), Middleware é definido como uma camada acima dos sistemas operacionais, mas abaixo do programa de aplicação, sendo assim, fornece uma abstração de programação comum através de um sistema distribuído, Middleware às vezes chamado informalmente de ‘Encanamento’, porque ele conecta partes de um aplicativo distribuído com pipes de dados e em seguida, passar dados entre eles.

PORQUE USAR O MIDDLEWARE?

Middleware serve para uma variedade de funções, ele gerencia conectividade com vários recursos de back-end, ROUSE (2014), um componente de middleware pode criar um pool de conexão para fornecer acesso rápido e eficiente a um banco de dados, também pode criar conexões para filas e tópicos de mensagens.

Segundo ROUSE (2014), middleware desempenha um papel importante no processamento simultâneo, no balanceamento de carga e até no gerenciamento de transações, tem capacidade de dimensionar vertical e horizontalmente para ajudar máquinas virtuais ou zonas de disponibilidade da nuvem. O software Middleware também pode lidar com problemas que podem surgir quando dois ou mais clientes tentam acessar ou atualizar um determinado recurso de back-end ao mesmo tempo.

Podemos afirmar então que middleware tem um papel muito importante para proteger recursos back-end e capacidade de desafiar clientes que exigem uma conexão segura e criptografada.

CONCEITOS

Middleware podemos dizer que ele é uma API (Conjunto de rotinas e padrões de programação), que compõem, uma abstração de protocolos de comunicação e camada de infraestrutura. Programas de middleware são fornecidos em um software local e serviços em nuvem, pode ser usado independentemente ou juntos, dependendo do caso de uso.

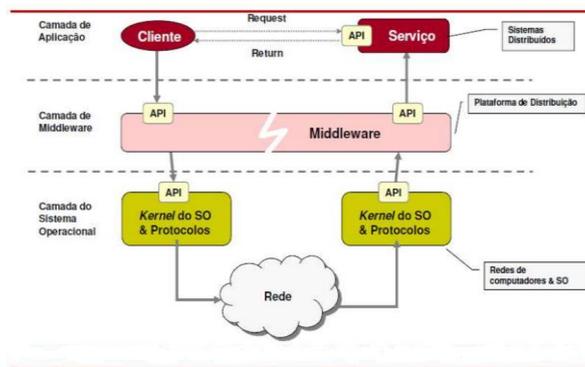


Figura 1: Visão Geral Middleware, ROCHA(2011)

Em sua visão geral, podemos deduzir, todas as solicitações baseadas em rede são tentativas de interagir com dados back-end, esses dados podem ser algo tão simples quanto uma imagem para exibir ou um vídeo para reproduzir ou pode ser tão complexo quanto um histórico de transações bancárias. Segundo ROUSE(2014), dados solicitados podem assumir muitos formulários diferentes e podem ser armazenados de várias maneiras, provenientes de um servidor de arquivo, obtidos de uma fila de mensagens ou persistidos em um banco de dados, o papel do middleware é habilitar e facilitar o acesso a esses recursos de back-end.

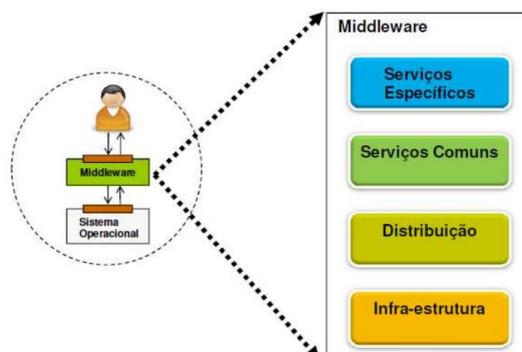


Figura 2: Estrutura Middleware, ROCHA(2011)

Afirma BAKKEN (2013), estruturas de middleware são projetados para mascarar alguns dos tipos de heterogeneidade que os programadores de sistemas distribuídos devem lidar sempre que mascarar a heterogeneidade das redes de hardware.

A definição clássica de um sistema operacional é o software que torna o hardware utilizável, da mesma forma, segundo BAKKEN, middleware pode ser considerado como um software que faz um programável sistema distribuído, assim como um computador simples sem um sistema operacional poderia ser programado com grande dificuldade, a programação de um sistema distribuído é em geral, muito mais difícil sem o middleware, especialmente quando for necessária uma operação heterogênea. Da mesma forma, é possível programar uma aplicação com uma linguagem assemble ou mesmo código de máquina, mas a maioria dos programadores se faz mais produtivo usar linguagens de alto nível para este fim.

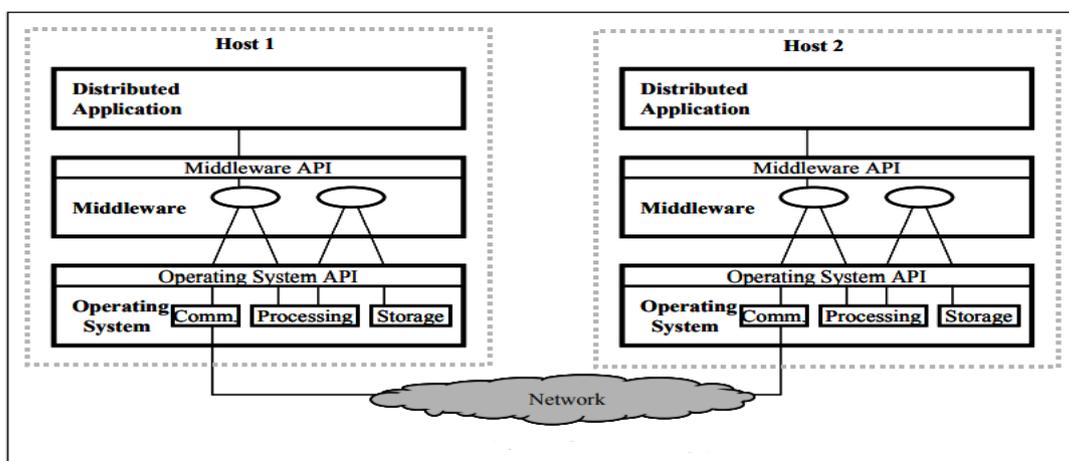


Figura 3: Camada Middleware, ROCHA(2011).

TIPOS DE MIDDLEWARE

Há muitos exemplos sobre tipos de middleware, cada um criado para sua função específica para conectar aplicativos e serviços web e da nuvem, a imagem abaixo cita alguns mais utilizados do middleware.

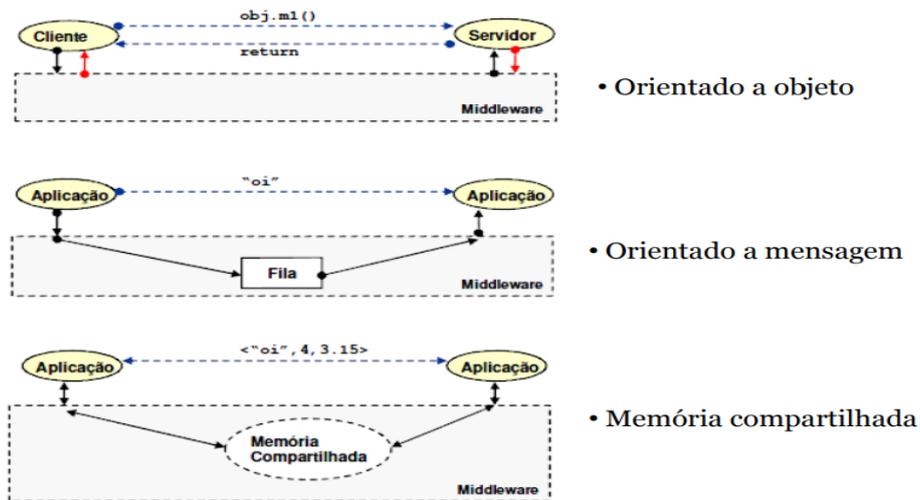


Figura 4: Tipos Middleware, ROCHA(2011).

ORIENTADO A OBJETO

Permite comunicação e interação com programas como contêineres em sistemas distribuídos, fornecendo uma abstração de um objeto ainda que remota, cujo métodos podem ser invocados como aqueles de um objeto no mesmo espaço de endereço.

ORIENTADO A MENSAGEM

Fornecer uma fila de mensagens que podem ser acessados através de uma rede. É uma generalização da construção do sistema operativo conhecido como a caixa de correio.

MEMÓRIA COMPARTILHADA

Interações entre processos para trocar dados de modo a manter um estado global consistente, sendo possível abstrair completamente as interações entre os processos, evitando a necessidade de lidar diretamente com os mecanismos e tornando manutenção de estado global transparente.

CONSIDERAÇÕES FINAIS

A evolução do middleware é um sinônimo de conceitos de um servidor de aplicativos e a medida que a computação em nuvem aumenta sua popularidade o middleware muda. Middleware é uma classe de tecnologia de software que foi projetado para ajudar e gerenciar a complexidade dos sistemas distribuídos, contudo, seu uso é quase indispensável, tornando assim um software poderoso por se tratar de segurança e desempenho.

REFERÊNCIAS BIBLIOGRÁFICAS

P. Bernstein. "Middleware: A Model for Distributed System Services." *Communication of the ACM*, 39:2, February 1996, 86-98

BAKKEN, David E.. MIDDLEWARE. April/2013.

J. Zinky, D. Bakken, and R. Schantz. “Architectural Support for Quality of Service for CORBA Objects”, *Theory and Practice of Object Systems*, 3:1, April 1997.

ROUSE, Margaret. *What is middleware?*, September/2014.

ROCHA, Vinicius. **Middlewares**. August /2011.

A. Campbell, G. Coulson, and M. Kouvanis. “Managing Complexity: Middleware Explained.” *IT Professional*, IEEE Computer Society, 1:5 September/October 1999, 22-28.

D. Schmidt, D. Levine, and S. Mungee. “The Design of the TAO Real- Time Object Request Broker” *Computer Communication*, Elsevier Science, 21:4, April, 1998.

P. Verissimo and L. Rodrigues. *Distributed System for System Architects*, Kluwer Academic Press, 2001