

Tecnologia em Sistemas para Internet

Sistemas Distribuídos

Aula 06 - Arquiteturas de Sistemas Distribuídos

Parte 03



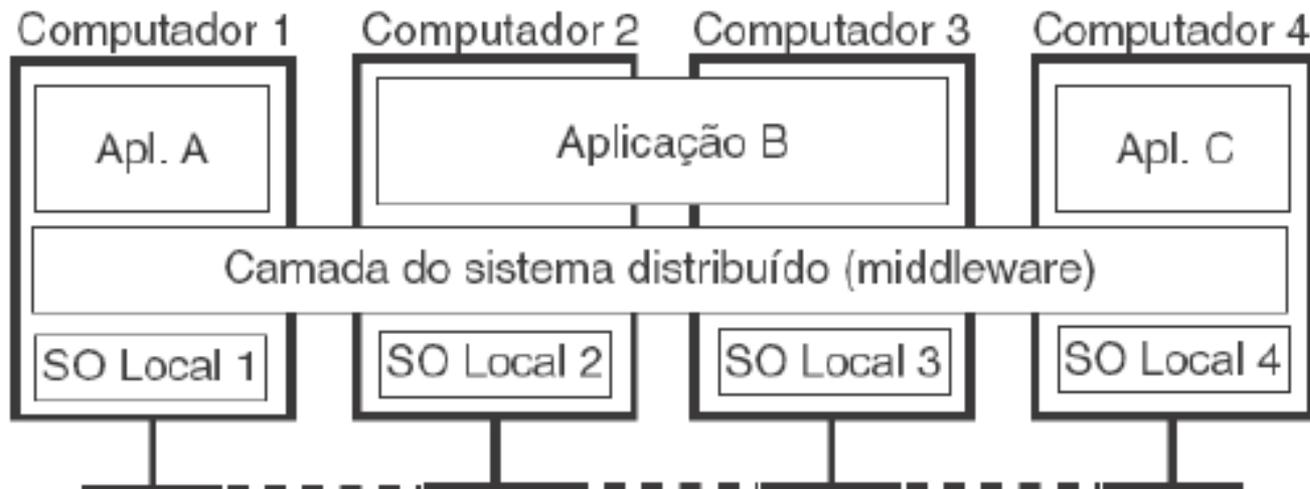
Objetivos

- Relacionar as diversas arquiteturas apresentadas com o comportamento do Middleware;
- Entender os mecanismos de autogerenciamento utilizados pelos sistemas distribuídos;



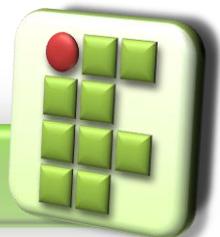
Arquitetura versus Middleware

- Onde o middleware se encaixa?
 - Middleware forma uma camada entre aplicações e plataformas distribuídas;



Arquitetura versus Middleware

- Normalmente, sistemas de middleware seguem um estilo arquitetônico específico;
 - Ex: CORBA -> baseado em objetos
- Moldar o middleware de acordo com um estilo arquitetônico específico tem como benefício a simplificação do projeto de aplicações;
- Contudo, uma óbvia desvantagem é que o middleware pode não ser o ideal para o desenvolvimento de determinados tipos de aplicações;



Arquitetura versus Middleware

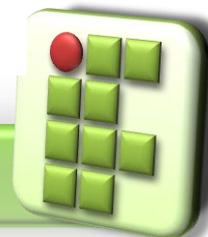
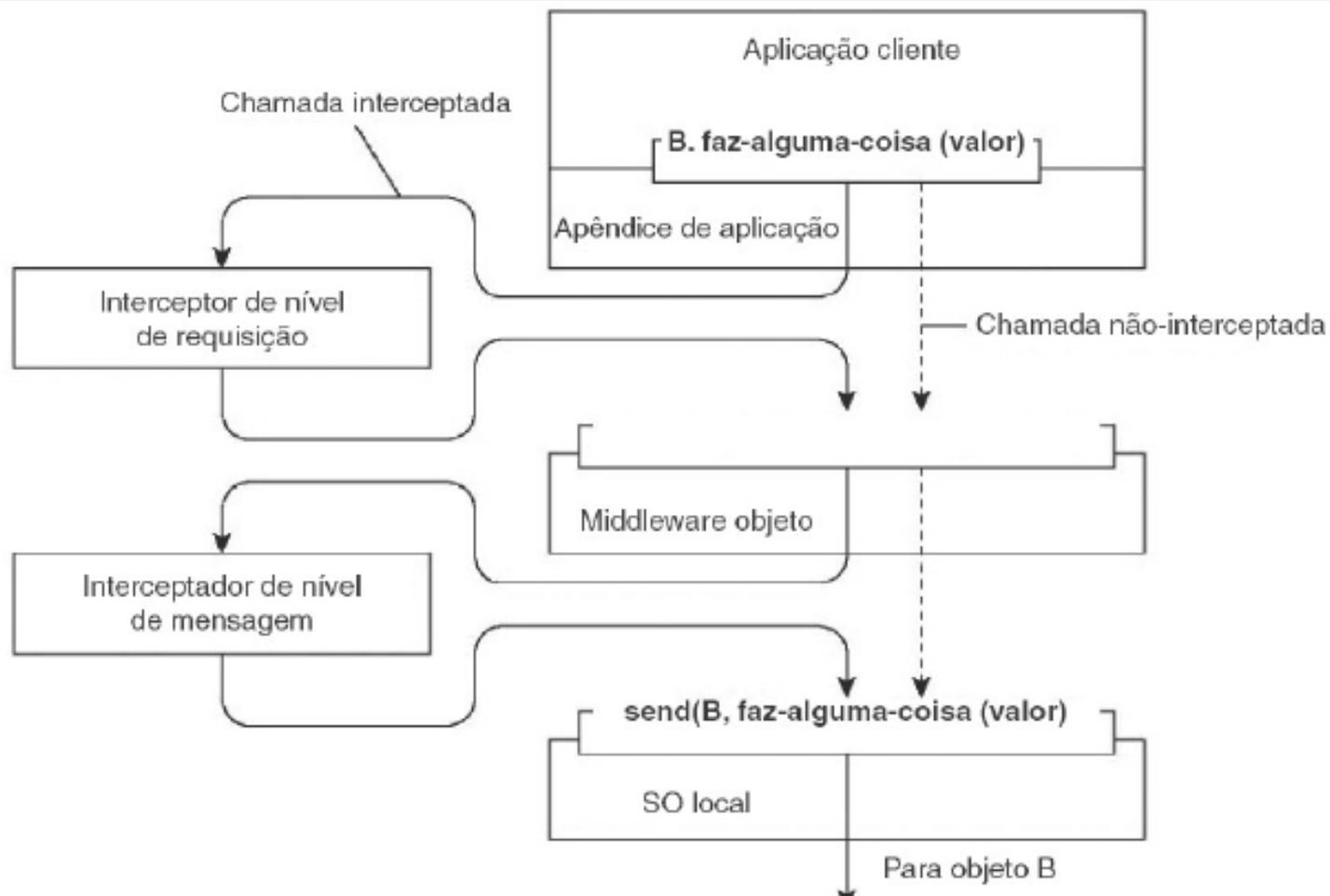
- A melhor solução é fazer sistemas de middleware de modo que sejam simples de configurar, adaptar e personalizar conforme a necessidade da aplicação;
- Alguns mecanismos para realização dessa tarefa são apresentados a seguir;



Interceptadores

- Um interceptador nada mais é do que um **constructo de software que interromperá o fluxo de controle usual e** permitirá que seja executado um outro código (específico da aplicação).





Software adaptativos

- O ambiente na qual as aplicações distribuídas são executadas está sempre mudando, seja por causa da mobilidade dos componentes, seja por causa do sistema de comunicação ou de problemas de hardware;
- Essas características mostram a necessidade de adaptação e em vez de fazer com que as aplicações sejam responsáveis por agir à mudanças, essa tarefa é colocada no middleware;



Técnicas para adaptação de Software

- Separação de Interesses;
- Reflexão Computacional;
- Projeto Baseado em Componentes;



Separação de Interesses

- Está relacionada com o modo tradicional de modularizar sistemas: separar as partes que implementam funcionalidades das que cuidam de outras coisa (funcionalidades extras)
 - Confiabilidade;
 - Desempenho;
 - Segurança.



Reflexão Computacional

- Se refere à capacidade de um programa inspecionar a si mesmo e, se necessário, adaptar seu comportamento;
- A reflexão foi embutida em algumas linguagens de programação, entre elas Java.



Projeto Baseado em Componentes

- Suporta adaptação por meio de composição.
- Um sistema pode ser configurado estaticamente durante a elaboração do projeto ou dinamicamente em tempo de execução.



Autogerenciamento

- Sistemas distribuídos – e em especial seu middleware associado – precisam fornecer soluções gerais de blindagem de modo que possam suportar o maior número possível de aplicações, mas com suporte as soluções específicas e por isso necessitam ser adaptativos;

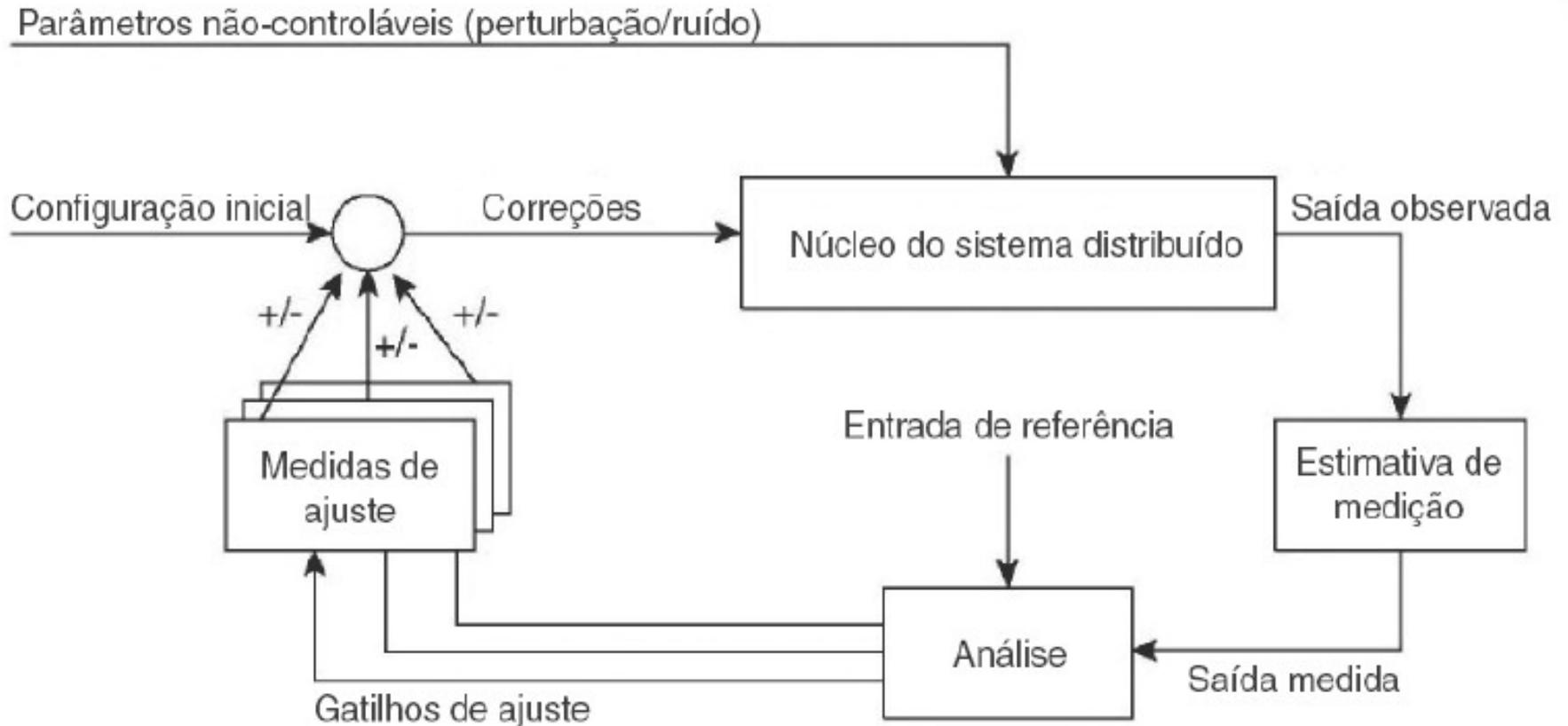


Autogerenciamento

- O autogerenciamento consiste em ter a organização de sistemas distribuídos como sistemas de realimentação de controle de alto nível que permite adaptação automática a mudanças;



Autogerenciamento



Referência

TANENBAUM, Andrew S.; STEEN, Maarten Van.,
Sistemas Distribuídos: Princípios e Paradigmas. São Paulo: Pearson Prentice Hall, 2007. 2ed.

LIMA, Rommel Wladimir de. *Arquiteturas de sistemas distribuídos.* 01 aug. 2013, 01 dec. 2103. Notas de Aula.

