

# Como Avaliar o Desempenho de um Middleware?

Nelson Rosa – nsr@cin.ufpe.br

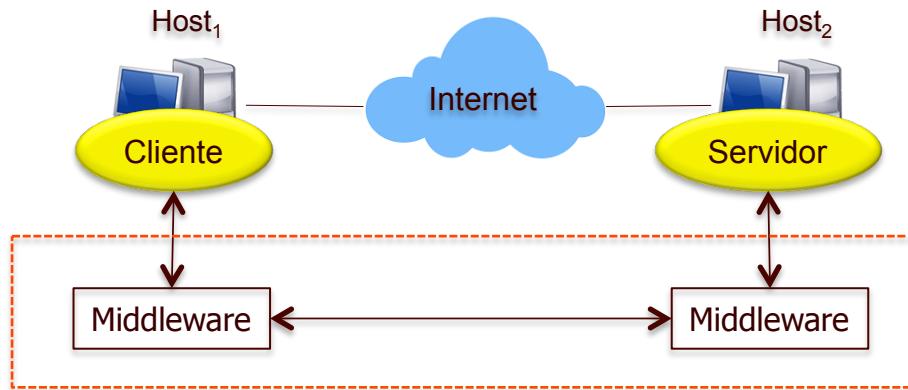
# Objetivos

---

- **Introduzir conceitos básicos de avaliação de desempenho**
- **Apresentar o passo-a-passo para avaliação de desempenho de um middleware**
- **Apresentar um exemplo de avaliação**

# Conceitos Básicos

Avaliação de Desempenho de Middleware [simplificada, tradicional]:  
mede-se o tempo de atendimento de solicitações, o tempo mais curto  
é uma indicação de que o middleware adotado é o mais rápido!



# Conceitos Básicos:: Avaliação de Desempenho

---

- **[O que é] aplicar técnicas a sistemas existentes/ou novos para obtenção de medidas de desempenho**
  
- **[Por que] avaliação de desempenho**
  - Encontrar gargalos nos sistemas
  - Planejamento de capacidade
  - Comparação de sistemas
  
- **[Como] medição, simulação, modelagem**

# Conceitos Básicos:: Técnicas para Avaliação de Desempenho

---

## ■ Simulação

- Modelagem das características que se deseja avaliar
- Predizer o desempenho futuro do sistema
- Uso de *benchmarks*

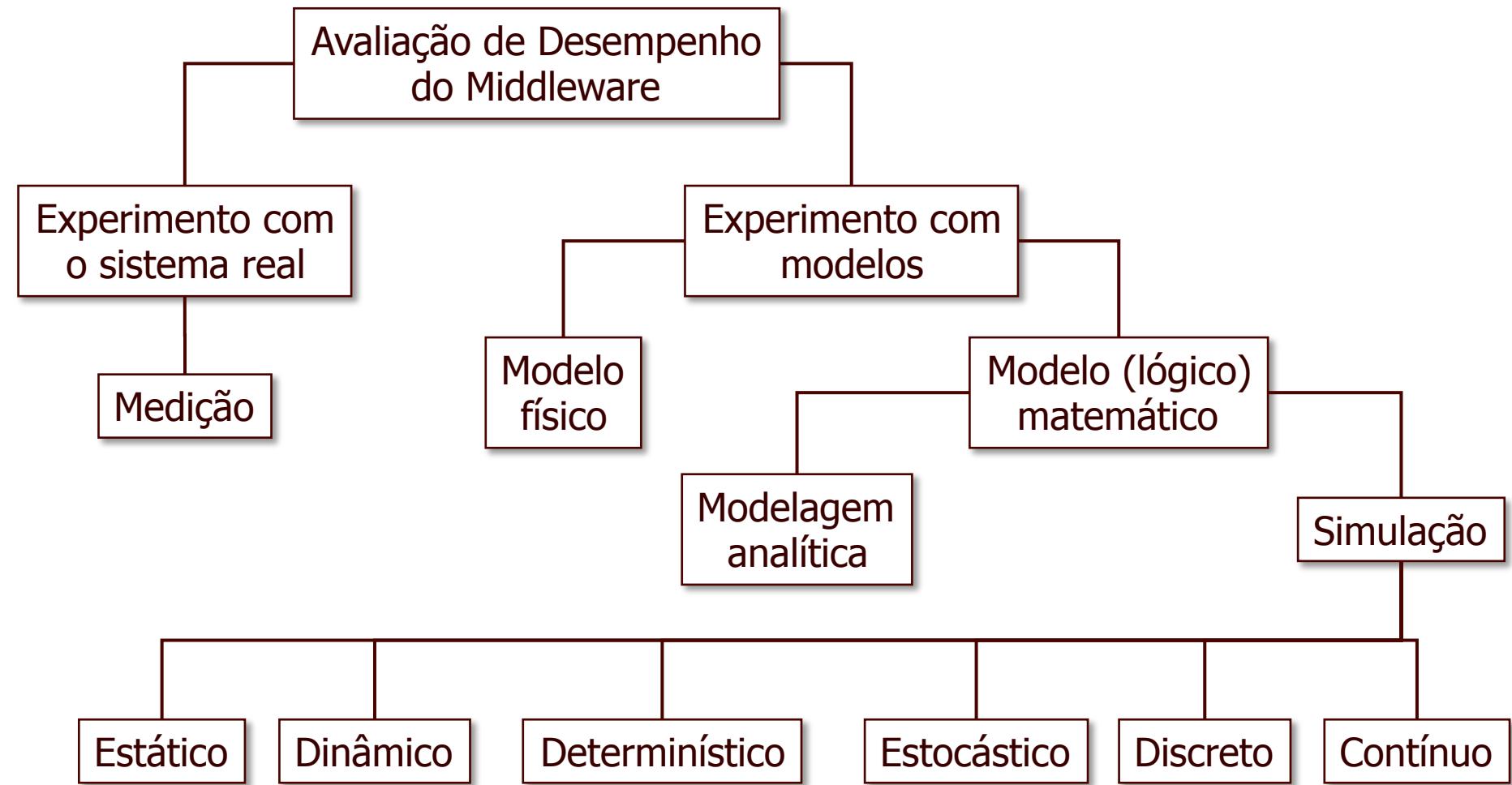
## ■ Medição

- Coleta de informações de um sistema real
- Uso de *benchmarks*

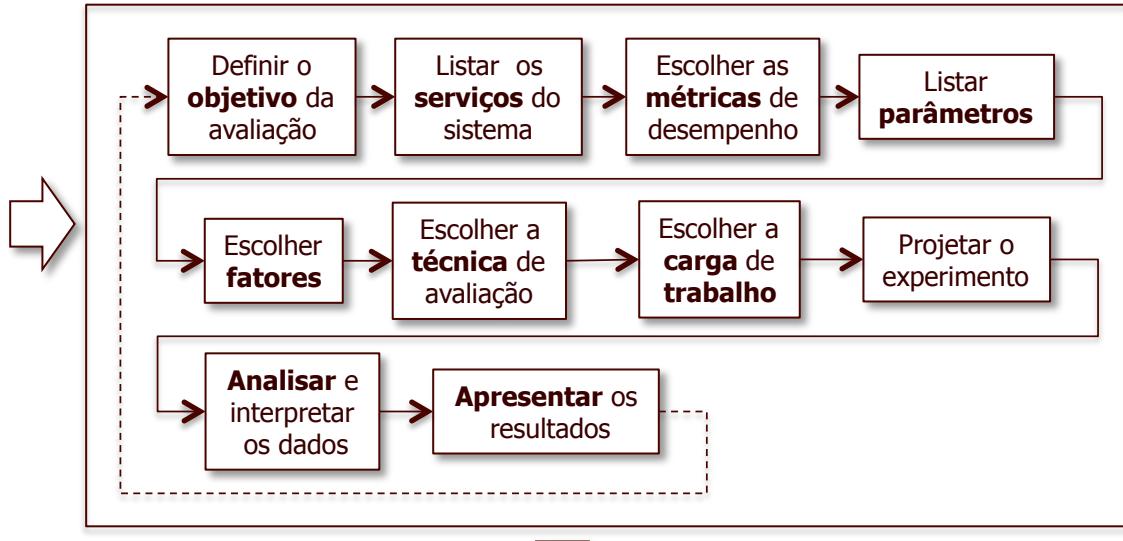
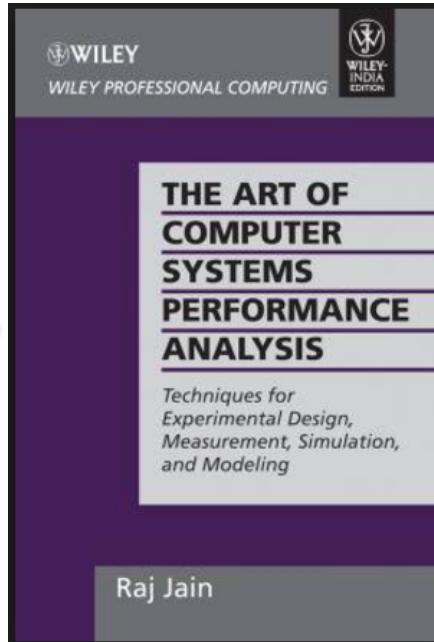
## ■ Modelagem analítica

- Capturar o comportamento do sistema através de uma fórmula matemática

# Conceitos Básicos:: Técnicas para Avaliação de Desempenho

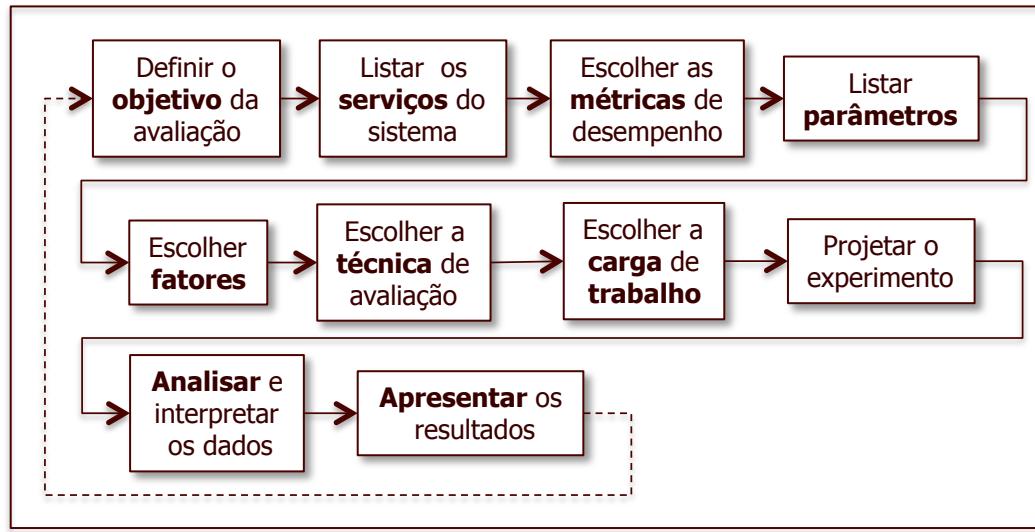


# Passo-a-Passo [Raj Jain]



# Passo-a-Passo:: Passo 1

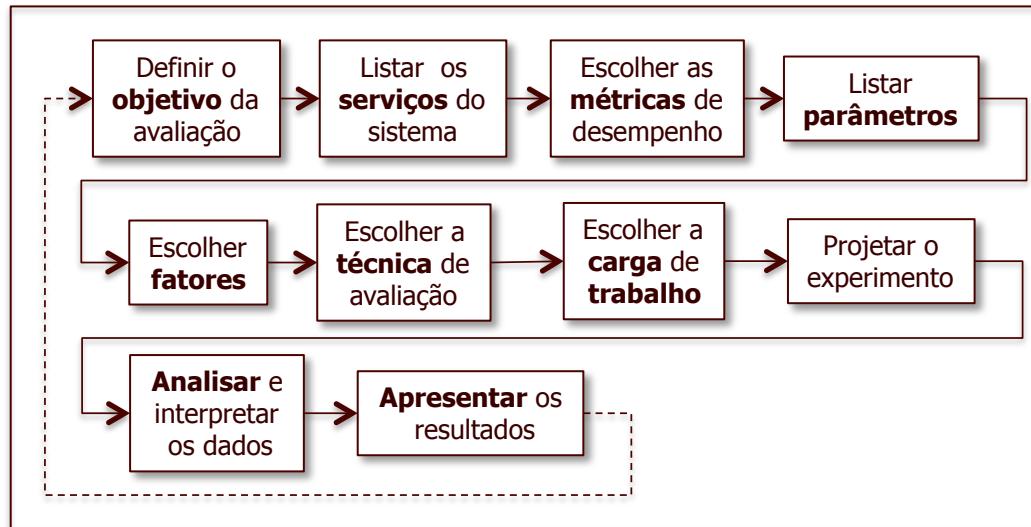
---



1. Definir o objetivo da avaliação, e.g., “comparar o desempenho da solução A com a Solução B”
2. Definir os “limites” do que você está sendo avaliado (**sistema**’), e.g., uma camada, um componente, um sistema inteiro

# Passo-a-Passo:: Passo 2

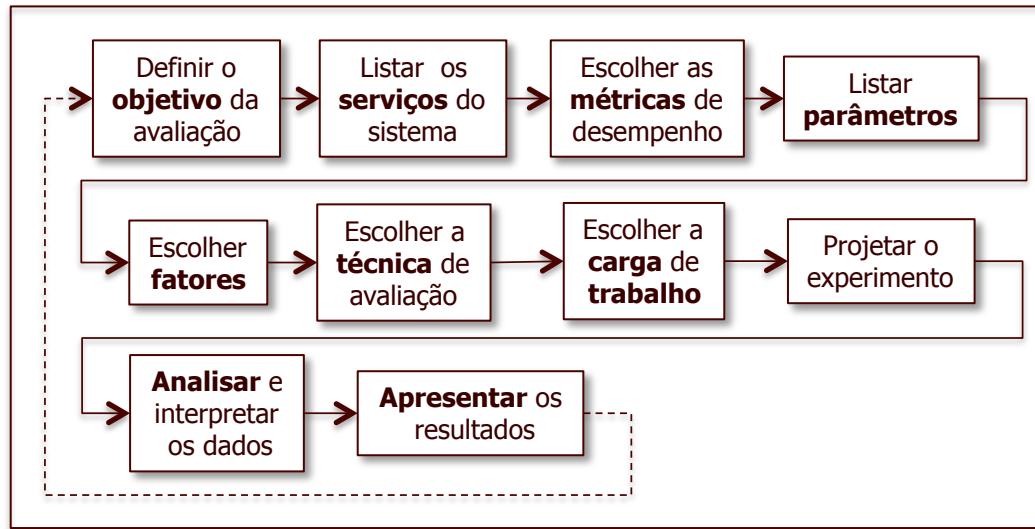
---



1. Definir os serviços (ou funcionalidades) providos pelo **sistema**, e.g., assumindo que o sistema é o *Marshaller*, os serviços a serem avaliados serão a “serialização” e “desserialização”.
2. Definir os “possíveis resultados” da requisição do serviço, e.g., “serialização realizada com sucesso”, “houve um erro na serialização”.

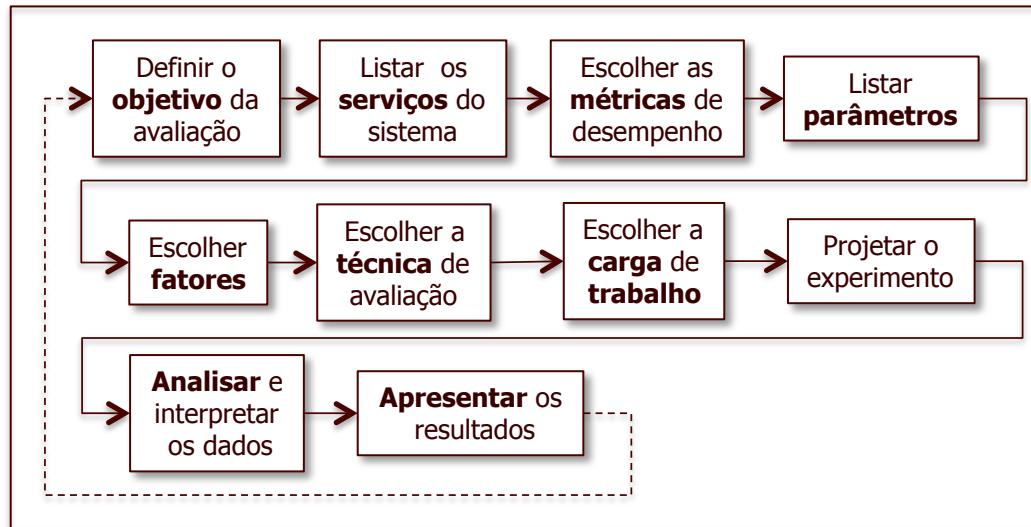
# Passo-a-Passo:: Passo 3

---



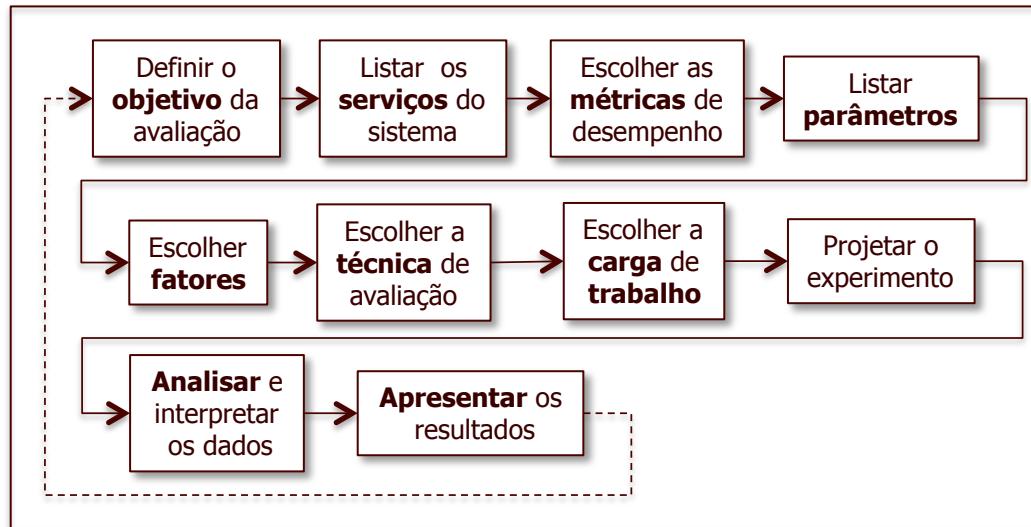
1. Definir critérios (métricas) para avaliar o desempenho do **sistema**, e.g., “tempo para executar a serialização”.
2. Muitas destas métricas estão relacionadas a velocidade, precisão e disponibilidade do sistema.

# Passo-a-Passo:: Passo 4



1. Definir tudo aquilo ("parâmetros") que afeta o desempenho do **sistema**
  1. "parâmetros do sistema" [em geral] são características de software e de hardware onde o sistema executa, e.g., velocidade da CPU, quantidade de memória, sistema operacional, etc.
  2. "parâmetros da carga de trabalho" são características das requisições feitas ao sistema, e.g., "tempo entre chamadas para serialização", "tamanho do dado que precisa ser serializado"
  3. Observação: Esta lista pode ser extensa.

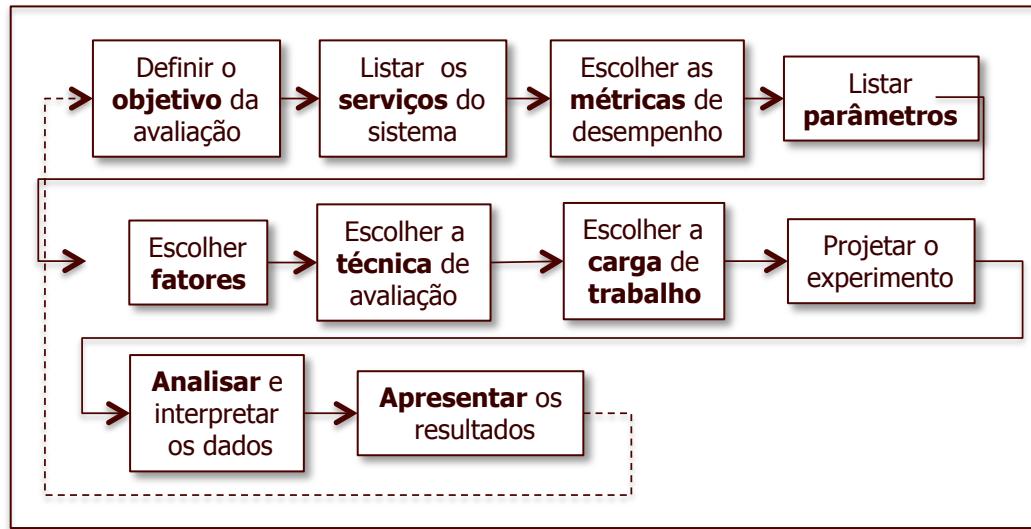
# Passo-a-Passo:: Passo 5



1. A lista de parâmetros pode ser dividida em dois grupos: aqueles que **variam** e aqueles que **não variam** durante a avaliação.
  - e.g., Processador (não varia)
  - e.g., tamanho do dado a ser serializado
2. Os parâmetros que **variam** são chamados de **fatores**.
3. As variações dos fatores são os **níveis**.
4. Aqueles parâmetros com alto impacto sobre o desempenho devem ser escolhidos como fatores.
5. É interessante começar com uma lista pequena de **fatores** e **níveis** e, se necessário, ir expandindo esta lista.

# Passo-a-Passo:: Passo 6

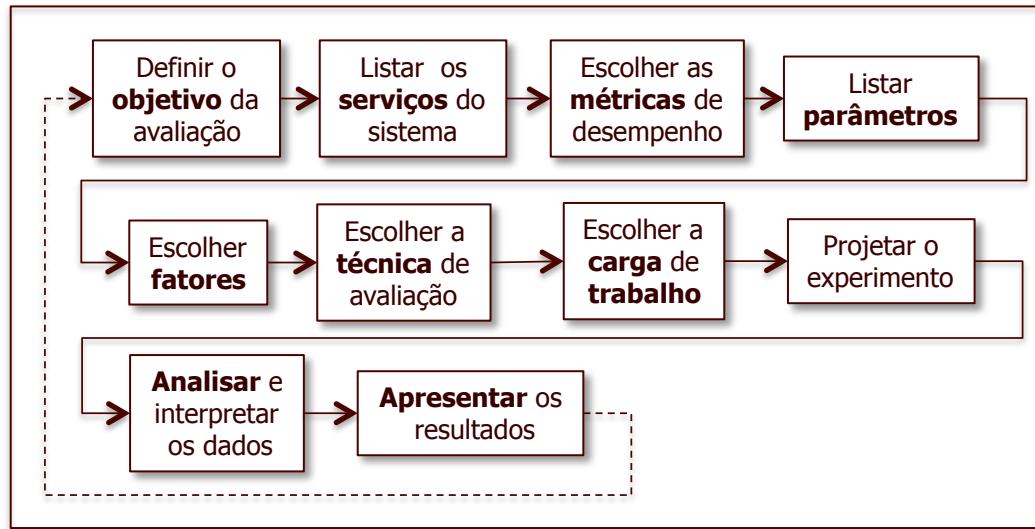
---



1. Existem três técnicas de avaliação de desempenho: medição, modelagem analítica e simulação.
2. Usaremos a **medição**.

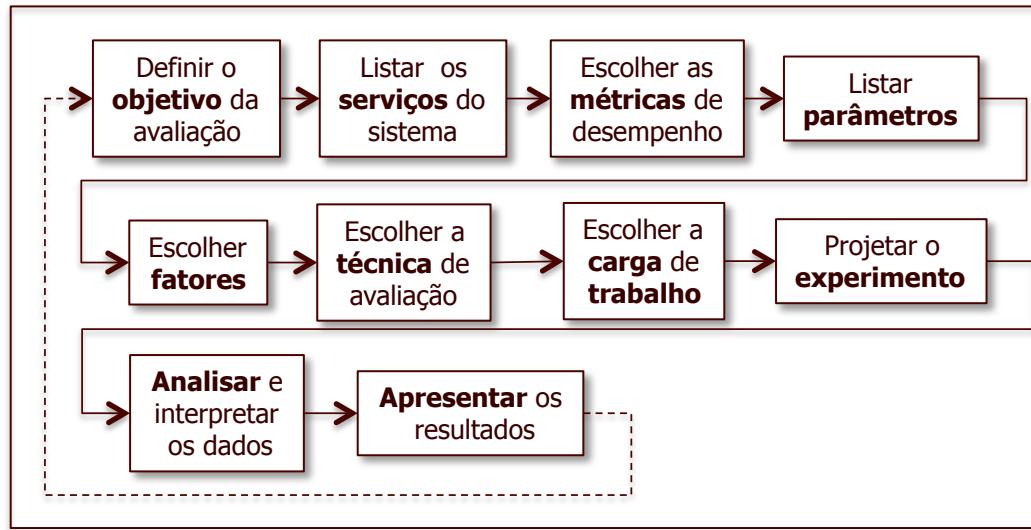
# Passo-a-Passo:: Passo 7

---



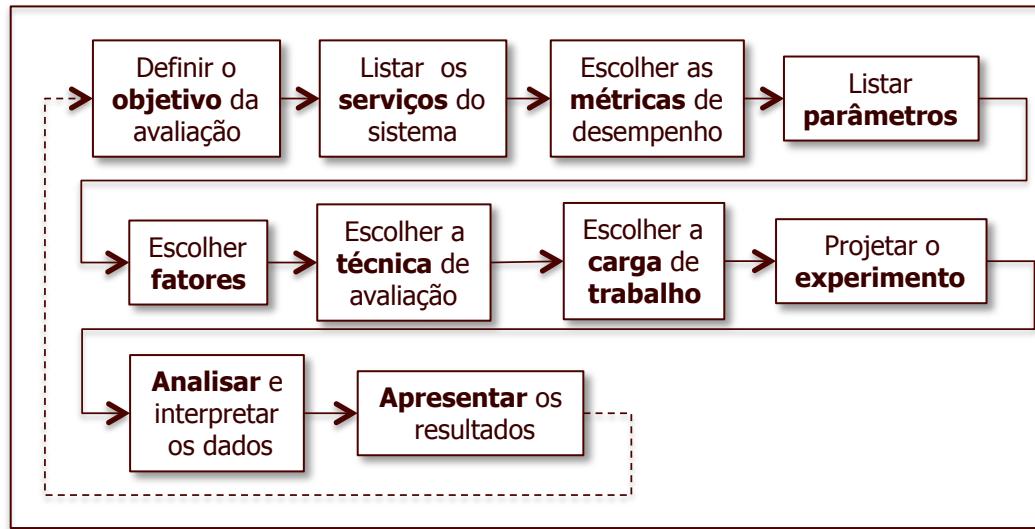
1. Escolher a carga de trabalho consiste em definir uma lista de **invocações ao sistema**, e.g., “invocar 10.000 vezes a serialização provida pelo *Marshaller*”
2. A carga de trabalho deve ser o mais próximo possível da “vida real”.

# Passo-a-Passo:: Passo 8



1. Como já foi definida a lista de fatores (e os seus respectivos níveis), é preciso decidir uma “**sequência de experimentos que gere o máximo de informações com um mínimo de esforço**”.
2. Em geral, a avaliação é feita em duas fases: (1) fase com **muitos fatores** e poucos níveis para determinar o efeito relativo de cada fator, e (2) em seguida, **reduz-se o número de fatores** – considerando-se aqueles que tem mais impacto - e aumenta-se o número de níveis.

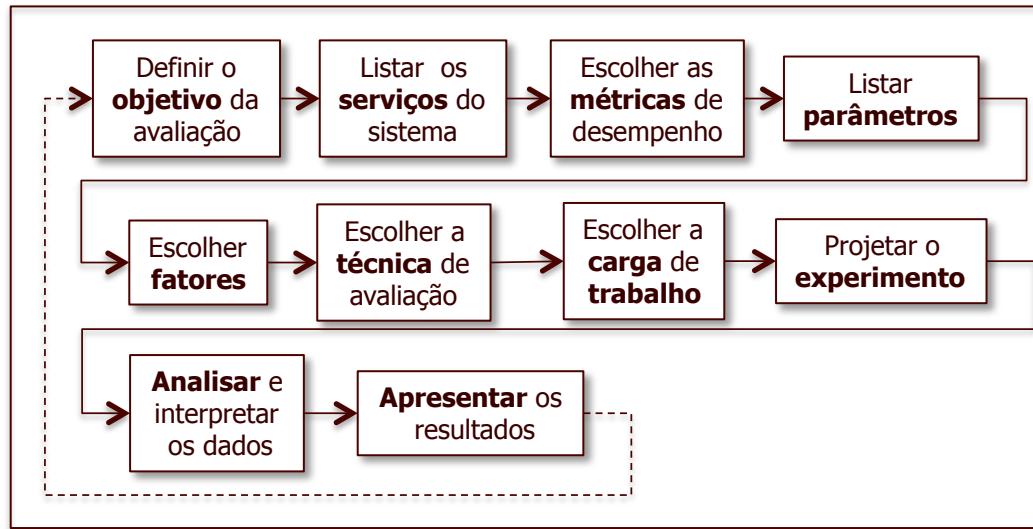
# Passo-a-Passo:: Passo 9



1. Em uma avaliação comparativa, a simples comparação da “média” pode levar a conclusões imprecisas.
2. A análise produz apenas resultados e não conclusões. Um analista é que precisa ver os resultados e chegar a conclusões.

# Passo-a-Passo:: Passo 10

---



1. Quase sempre os resultados da avaliação são apresentados na forma de gráficos.

# Avaliação de Desempenho de Middleware

---

## ■ Dificuldades

- Ausência de estado, controle e clock globais
- Heterogeneidade
- Ausência de métricas
- Grande diversidade de modelos de computação
- Grande diversidade de operações, o que leva a dificuldade em construir *benchmarks*

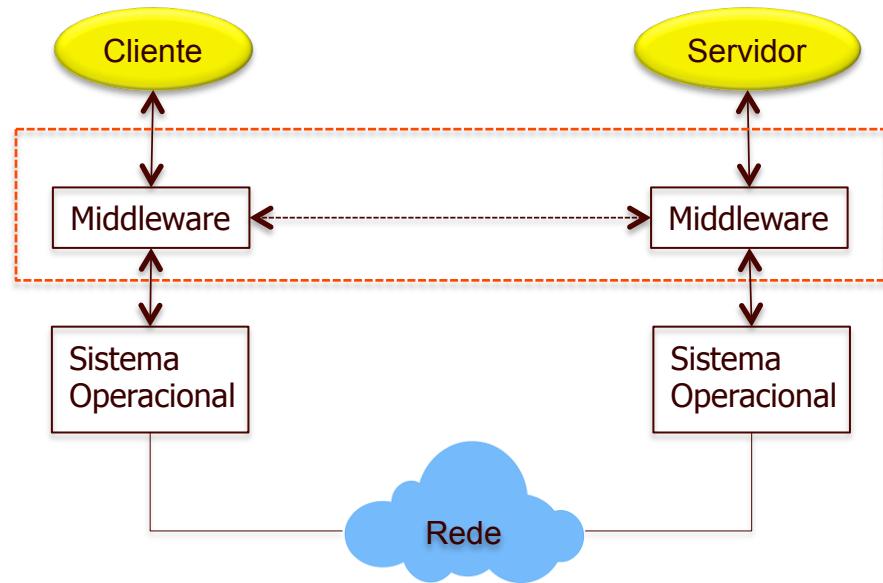
## ■ Além das dificuldades anteriores...

- Middleware não é aplicação, não é SO e não é rede, e.g., métricas usadas refletem este ponto, e.g., conversão de tipos (heterogeneidade)
- Métricas adotadas em redes de computadores [já amplamente estudadas] dizem respeito “apenas” à comunicação, e.g., # de pacotes transmitidos

# Métricas para Avaliação de Desempenho de Middleware

## ■ Middleware

- Tempo em cada uma das camadas
  - Distribuição no cliente, distribuição no servidor, ambos
- Tempo em cada um dos componentes, e.g., marshaller, invoker, requestor
- Tempo para demultiplexação
- Tamanho médio do pool de objetos



# **Passo-a-Passo de Avaliação de Desempenho do Middleware**



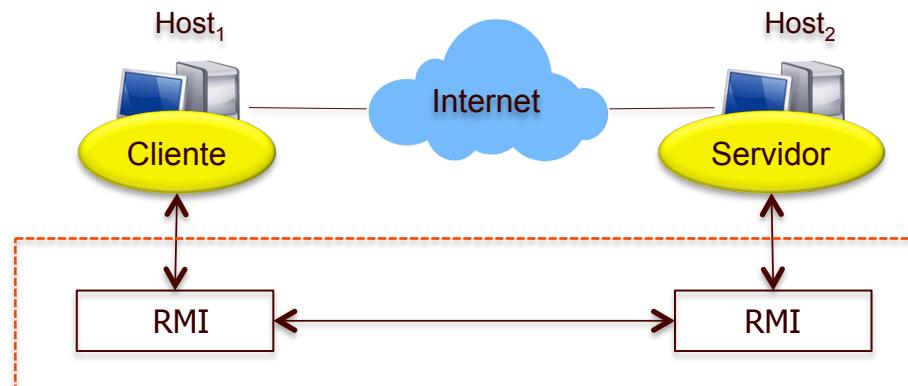
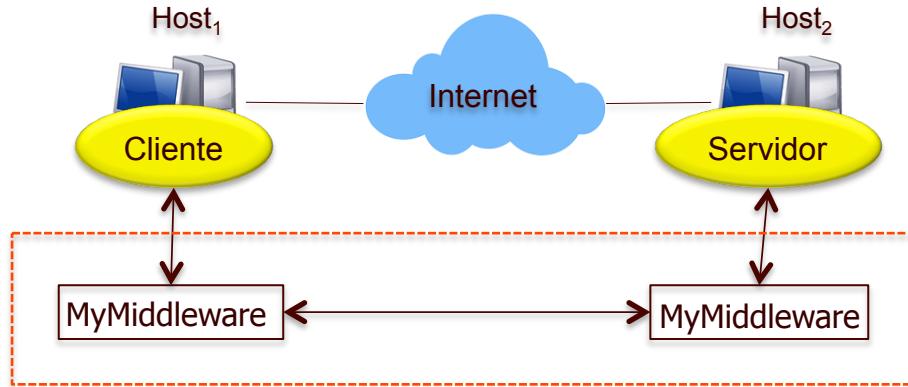
UNIVERSIDADE  
FEDERAL  
DE PERNAMBUCO

---

**CIn.ufpe.br**

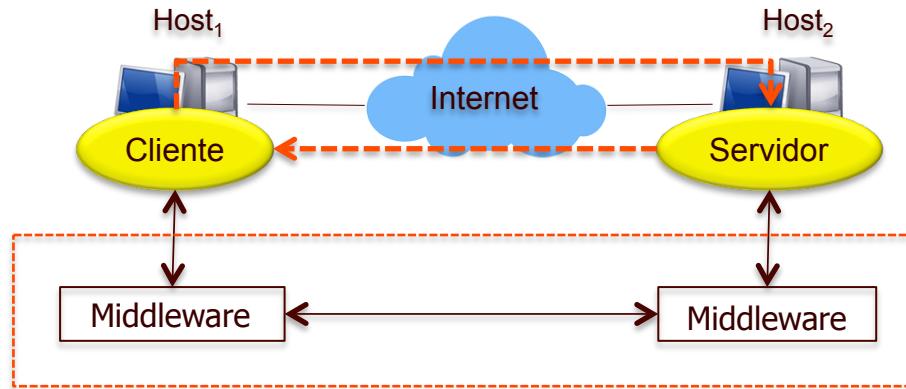
# Passo-a-Passo:: Definir o Objetivo da Avaliação

## ■ Comparar o desempenho do MyMiddleware com o RMI



# Passo-a-Passo:: Listar os serviços do sistema

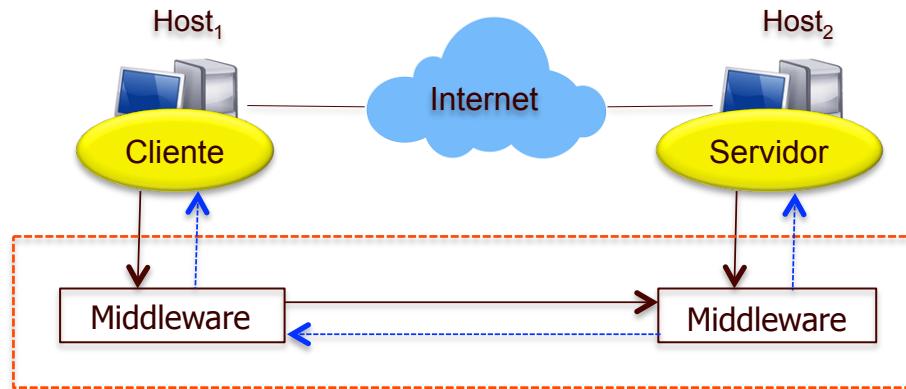
## ■ Invocação remota de método



# Passo-a-Passo:: Escolher as Métricas de Desempenho

## ■ Métrica

- “*Round trip time* de uma invocação” (medida no “Cliente”)



# Passo-a-Passo:: Listar Parâmetros

---

Parâmetros do Sistema	Valor
Processador	2,9 GHz Intel Core i7
Memória	8 GB 1600 MHz DDR3
WiFi	Desligada
Sistema Operacional	macOS Sierra Versão 10.12.6

## Parâmetros da Carga de Trabalho

Operação remota

Tempo entre invocações sucessivas

Número de invocações

# Passo-a-Passo:: Escolher Fatores

---

Parâmetros do Sistema	Valor
Processador	2,9 GHz Intel Core i7
Memória	8 GB 1600 MHz DDR3
WiFi	Desligada
Sistema Operacional	macOS Sierra Versão 10.12.6

Parâmetros da Carga de Trabalho	Níveis
Operação remota	add
Tempo entre invocações sucessivas	Distribuição normal (média = 10 ms, desvio padrão = 0.01 ms)
Número de invocações	1000, 2500, 5000, 7500, 10000

# **Passo-a-Passo:: Escolher a Técnica de Avaliação**

---

## **■ Medição**

# **Passo-a-Passo:: Escolher a Carga de Trabalho**

---

- Um cliente irá gerar invocações sucessivas ao objeto remoto

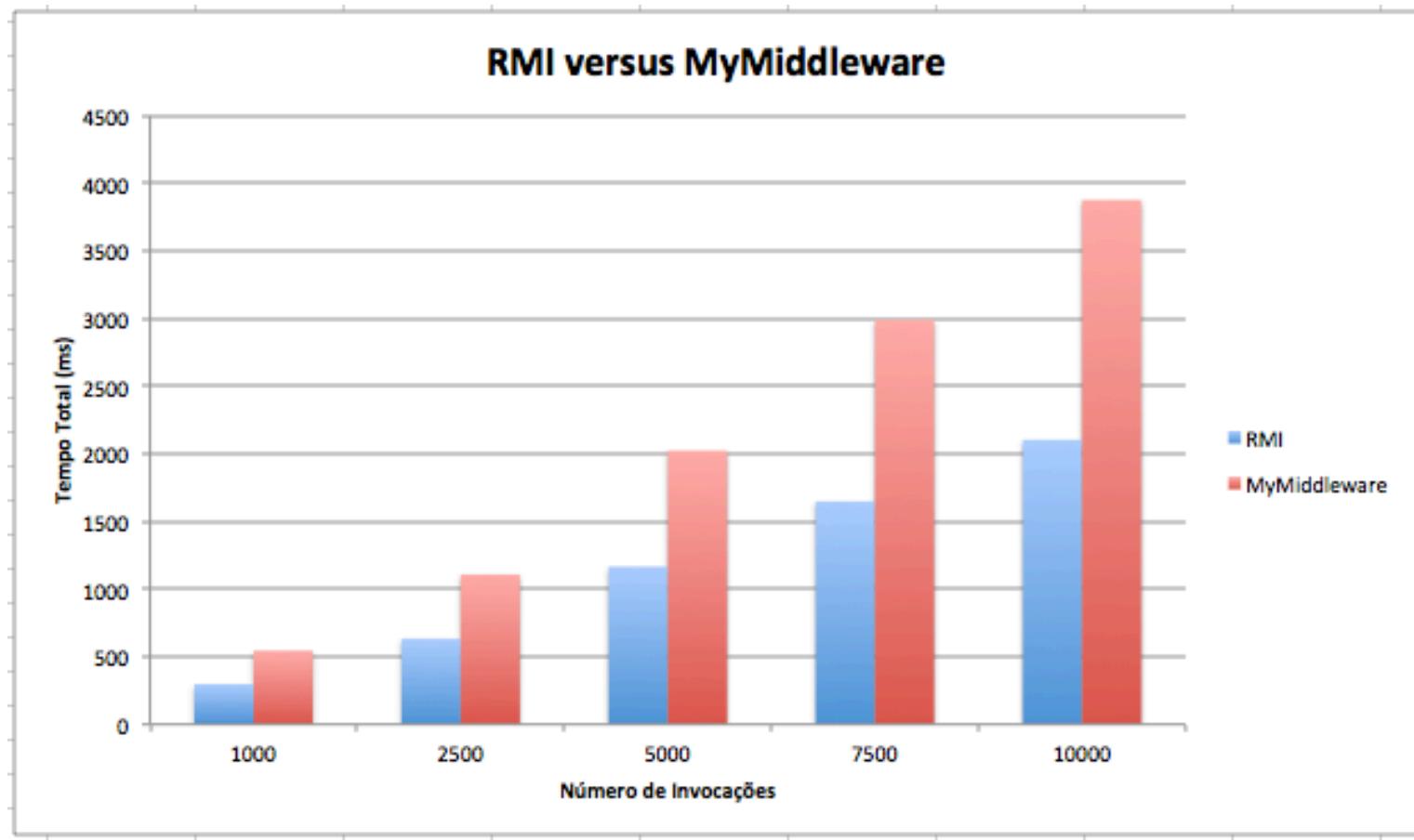
# **Passo-a-Passo:: Projetar o Experimento**

---

- Os experimentos serão realizados com a operação (add), fixando o tempo entre invocações e variando o número de invocações.

# Passo-a-Passo:: Analisar e interpretar os resultados

---



---

# **Fim dos Slides**