Centro de Informática
UFPE

Integração e Evolução de Sistemas de Informação

Relembrando Arquiteturas de TI

David J. M. Cavalcanti
djmc@cin.ufpe.br

Centro de Informática
UFPE

“Arquitetura” é uma palavra promíscua no contexto de Ciência da Computação

As arquiteturas servem para:

1) Definir as estruturas computacionais usadas por sistemas de informação
2) Modelar como o sistema deve ser organizado
3) Descrever a estrutura geral do sistema de software
4) Identificar os principais componentes estruturais do sistema e como eles comunicam

Centro de Informática
UFPE

Arquiteturas Computacionais

Arquiteturas Computacionais

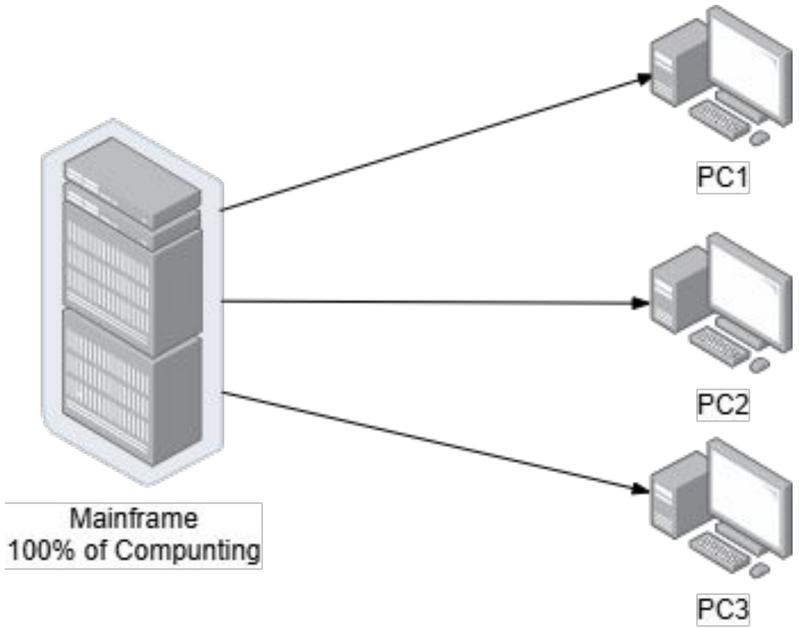
Centro de Informática
UFPE

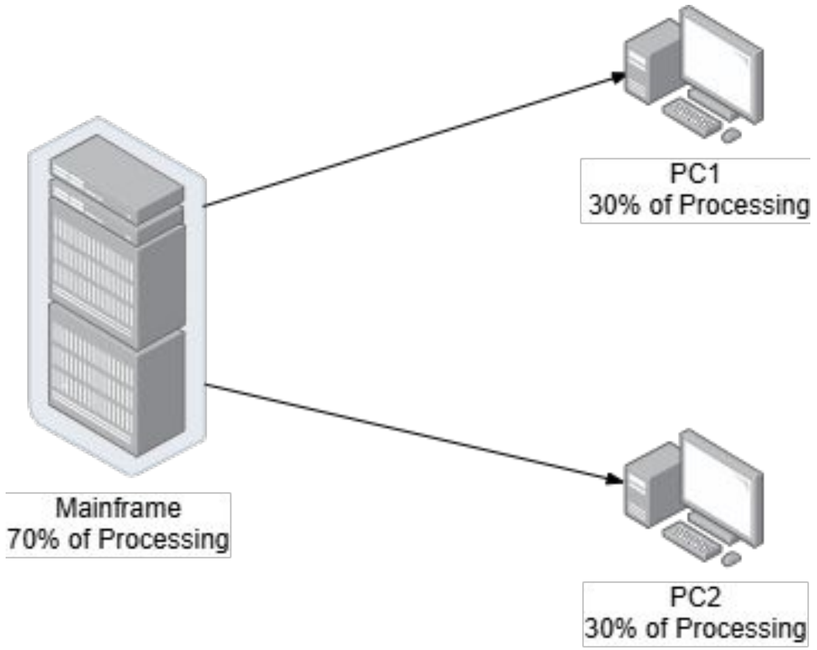
- ➔ Mainframe (1959+)
- ➔ Computadores pessoais (1981+)
- ➔ Cliente/Servidor (1983+)
- ➔ Computação Corporativa (1992+)
- ➔ Peer-to-Peer (P2P) (1999+)
- ➔ Computação em Nuvem [SOA/Móvel/Microservices] (2000+)

Arquiteturas Computacionais

Centro de Informática UFPE

- ➔ Mainframe (1959+)
- ➔ Computadores pessoais (1981+)

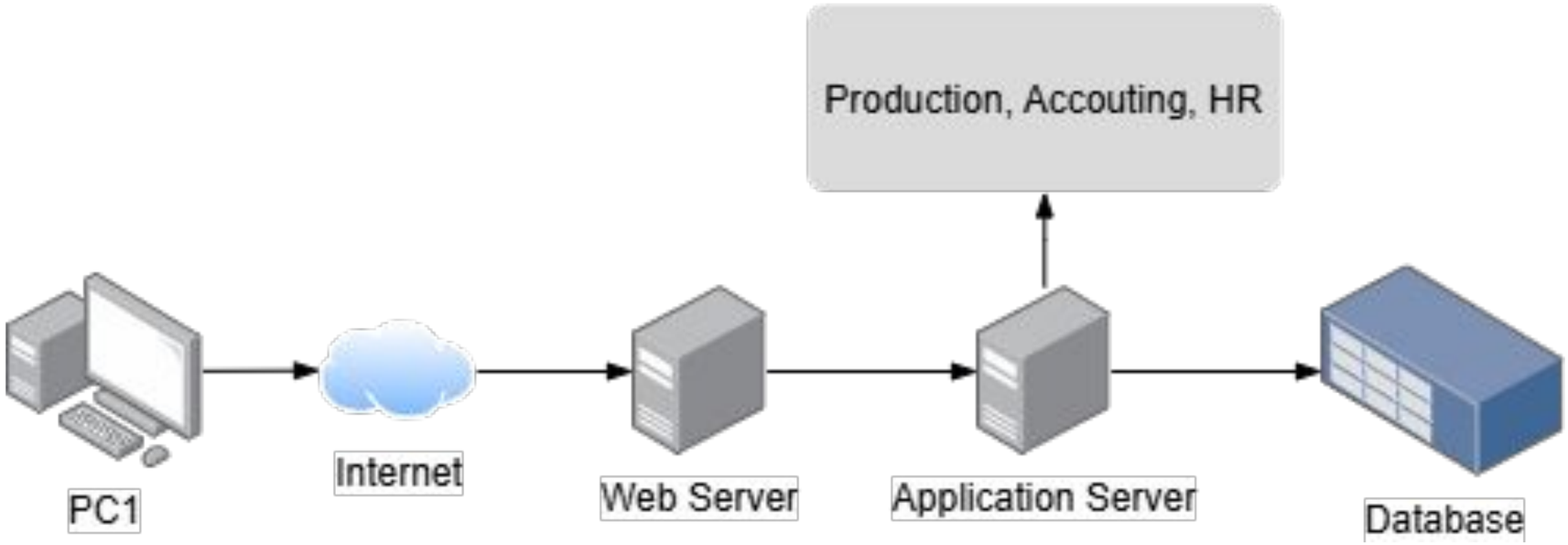




Arquiteturas Computacionais

Centro de Informática
UFPE

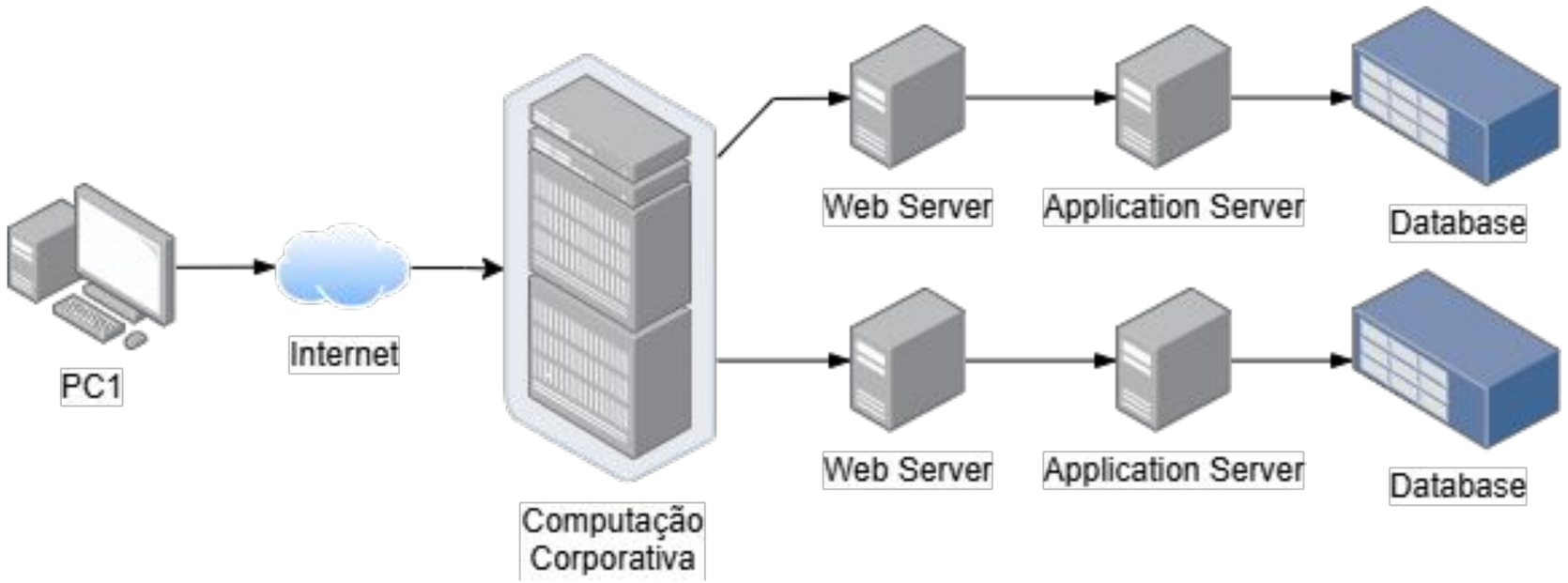
➔ Cliente/Servidor (1983+)



Arquiteturas Computacionais

Centro de Informática
UFPE

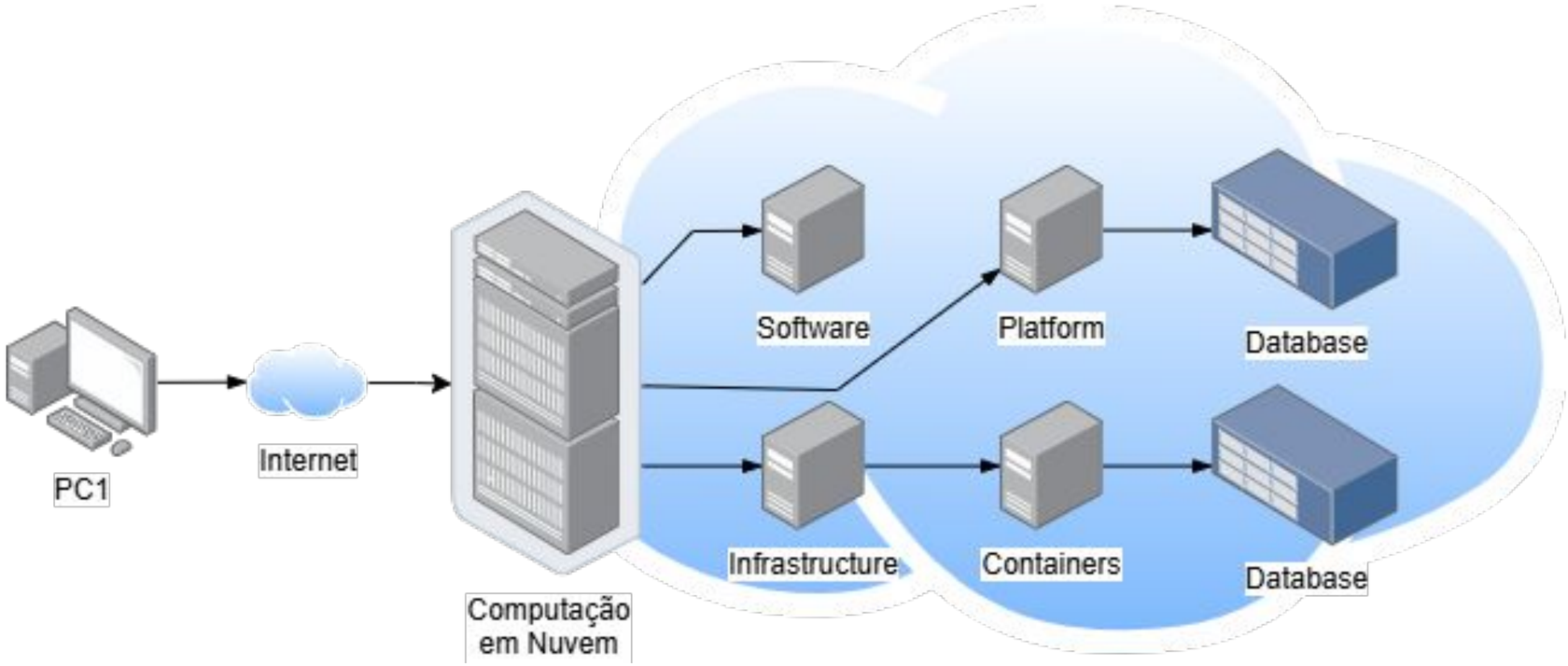
➔ Computação Corporativa (1992+)



Arquiteturas Computacionais

Centro de Informática
UFPE

➔ Computação em Nuvem [SOA/Móvel] (2000+)



Computação em Nuvem

Centro de Informática
UFPE

- ➔ IaaS (Infrastructure as a Service)
  - ◆ VMs, Redes e Armazenamento
- ➔ PaaS (Platform as a Service)
  - ◆ SOs, Linguagem de Programação, Banco de Dados
- ➔ SaaS (Software as a Service)
  - ◆ E-mail, CRM e ERP

Centro de Informática
UFPE

Arquiteturas de Software

Centro de Informática
UFPE

O que é a arquitetura de software?

Centro de Informática
UFPE

Representação abstrata ou de organização da estrutura de software como uma coleção de componentes computacionais — ou simplesmente componentes — juntamente com uma descrição das interações entre esses componentes — os conectores.

Garlan & Shaw (1993). “An Introduction to Software Architecture”

Para que serve um projeto de arquitetura de software?

Centro de Informática
UFPE

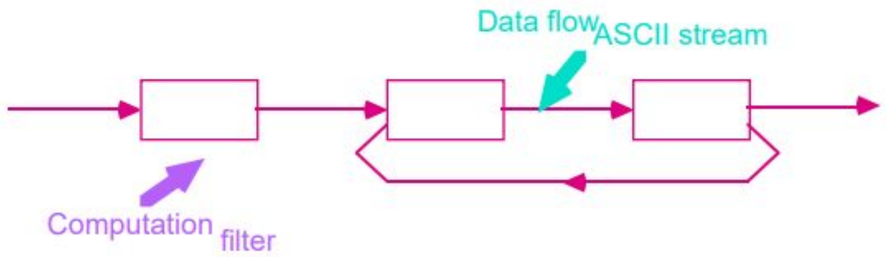
- ➔ Apoio à comunicação de stakeholders
- ➔ Apoio na análise de sistemas
- ➔ Apoio ao reuso de componentes

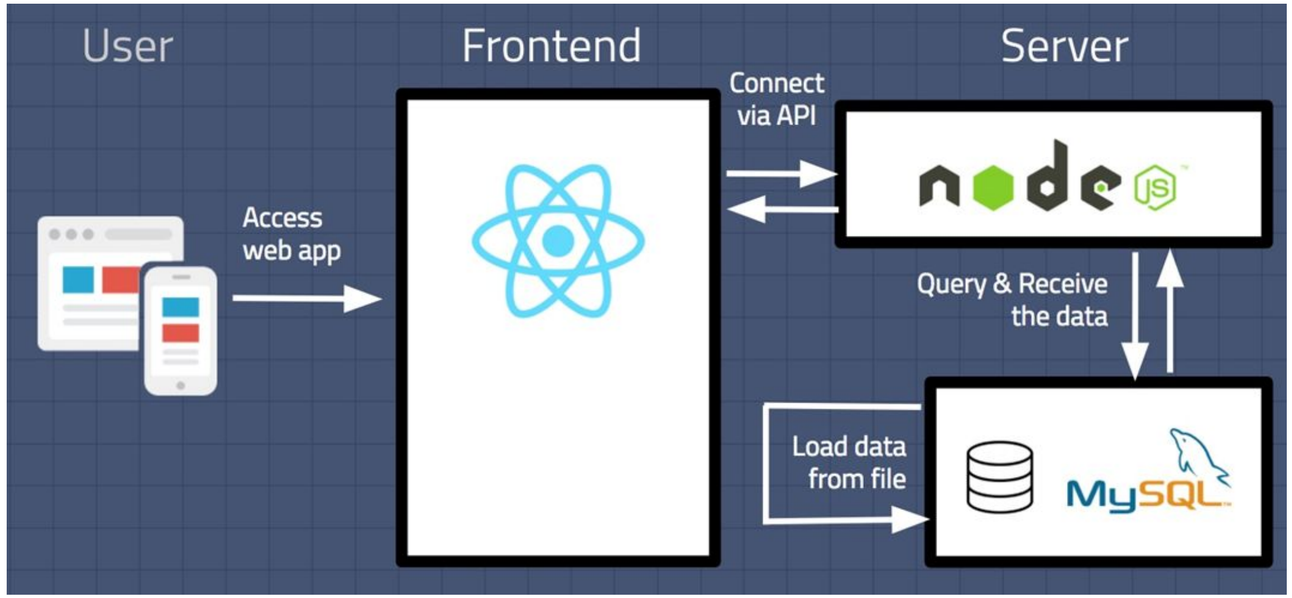
Projeto de Arquitetura

Centro de Informática UFPE

➔ Como o sistema deve ser organizado:

- Descreve estrutura geral do sistema
- Identifica os principais componentes estruturais do sistema e os relacionamentos entre eles





Elementos da Arquitetura de Software

Centro de Informática
UFPE

- Components
  - Módulos computacionais: cliente, servidor, ou uma aplicação
- Conectores
  - Interação entre os módulos: uma chamada de função, uma consulta em um banco de dados, etc.
- Configurações
  - Como os componentes são conectados, possíveis restrições, etc.

Linguagem de Descrição de Arquitetura

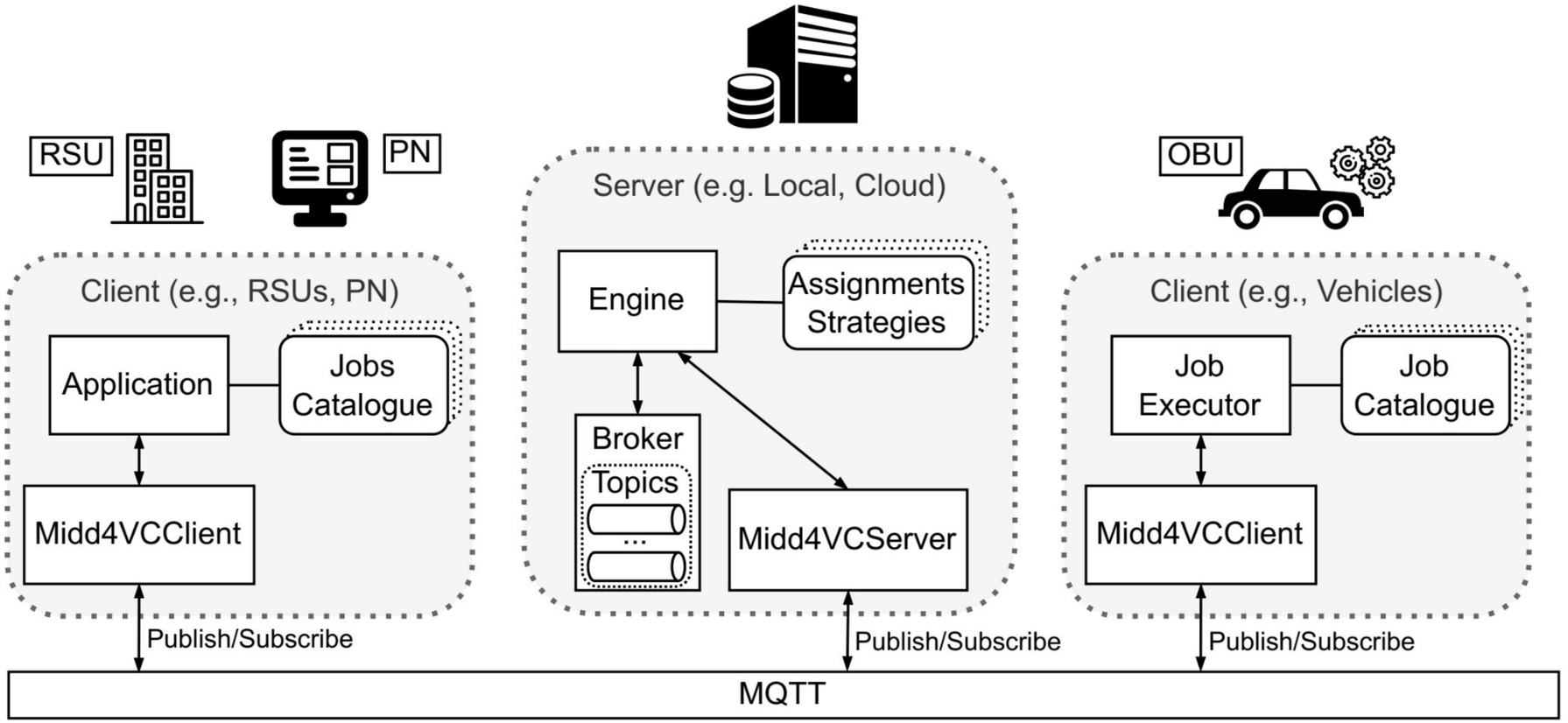
Centro de Informática
UFPE

➔ ADL é uma linguagem especializada para descrever arquiteturas de software:

- Representação formal e precisa da arquitetura de software
- Permite validar e analisar antes da implementação
- Facilita a explicação para os stakeholders e a manutenção
- Pode ser processada computacionalmente

Exemplo: Arquitetura

Centro de Informática
UFPE



Exemplo: ADL + Arquitetura

Centro de Informática UFPE

```
1. adl = {
2.    'components': {
3.       'QueueProxy': 'queueproxy',
4.       'Marshaller': 'marshaller',
5.       'ClientRequestHandler': 'crh'
6.    },
7.    'attachments': {
8.       'queueproxy': 'marshaller',
9.       'marshaller': 'crh'
10.   },
11.   'adaptability': {
12.      'type': 'evolutive',
13.      'timeout': 1200
14.   },
15.   'configuration': {
16.      'start': 'queueproxy',
17.      'otherConfigs': {}
18.   }
19.}
```

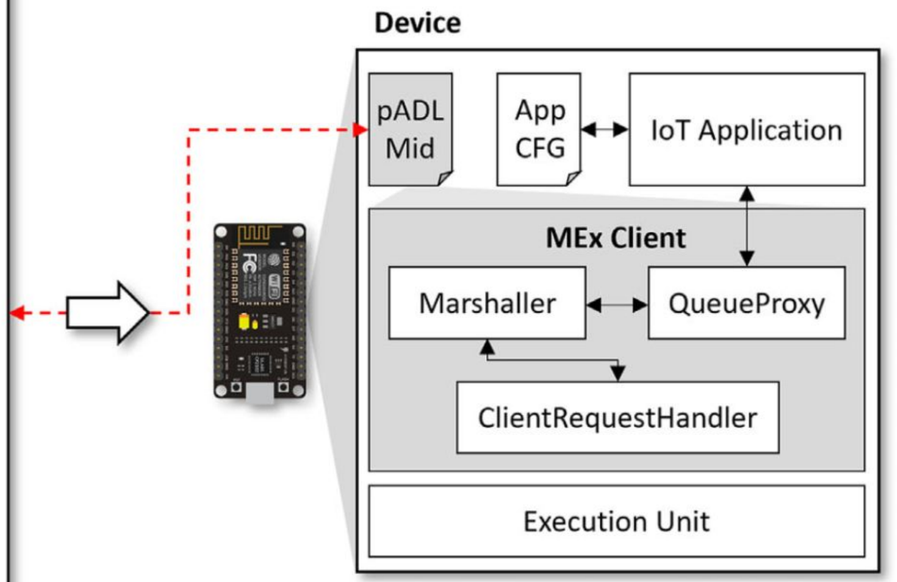


FIGURE 2 An architecture of a *M*iddleware *Ex*tendify (MEx) client in Python-based architecture description language (*p*ADL).

Cavalcanti & Rosa (2023). “Customizable and adaptable middleware of things”

Centro de Informática
UFPE

Estilos Arquiteturais

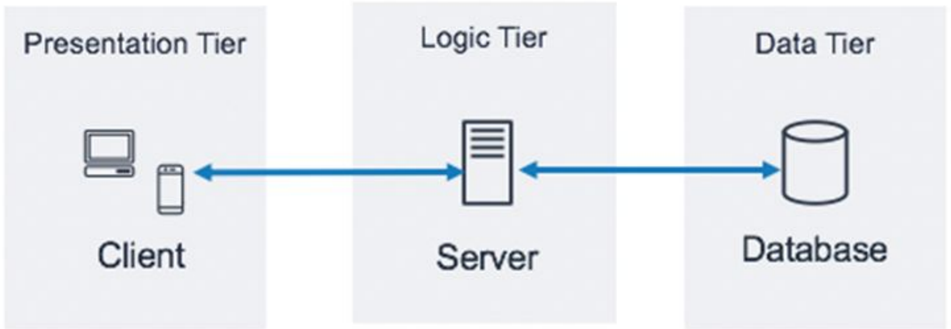
O que é o estilo arquitetural?

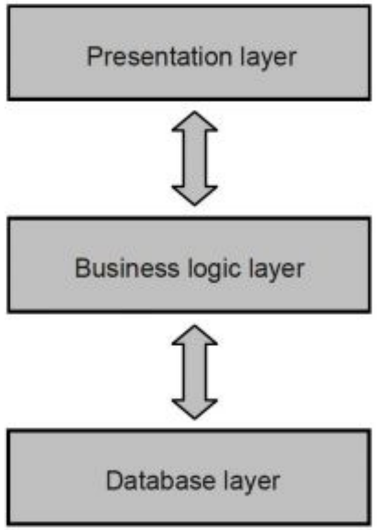
- Especifica como os componentes devem ser organizados e como eles devem se comunicar

Arquitetura em Camadas

Centro de Informática
UFPE

- Arquitetura “n-tier”
- Sistema organizado em camadas
- Separação de responsabilidade claras
- Comunicação normalmente entre camadas adjacentes

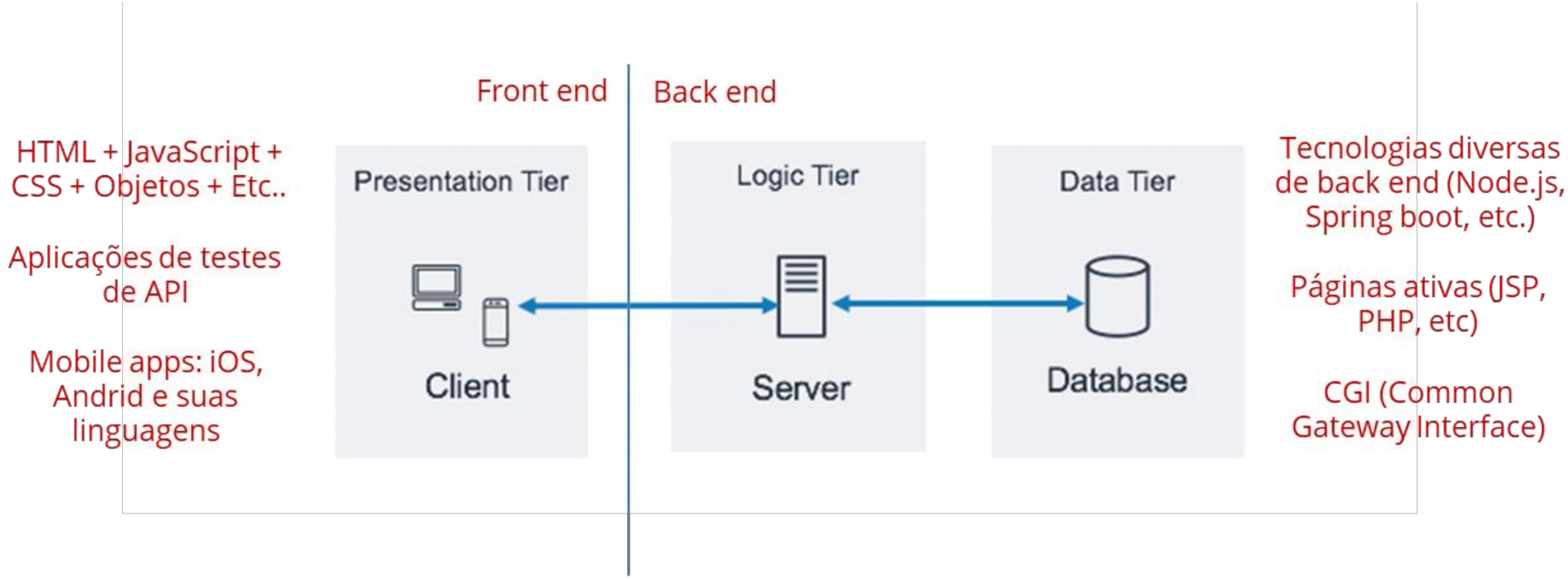




Arquitetura em Camadas

Centro de Informática
UFPE

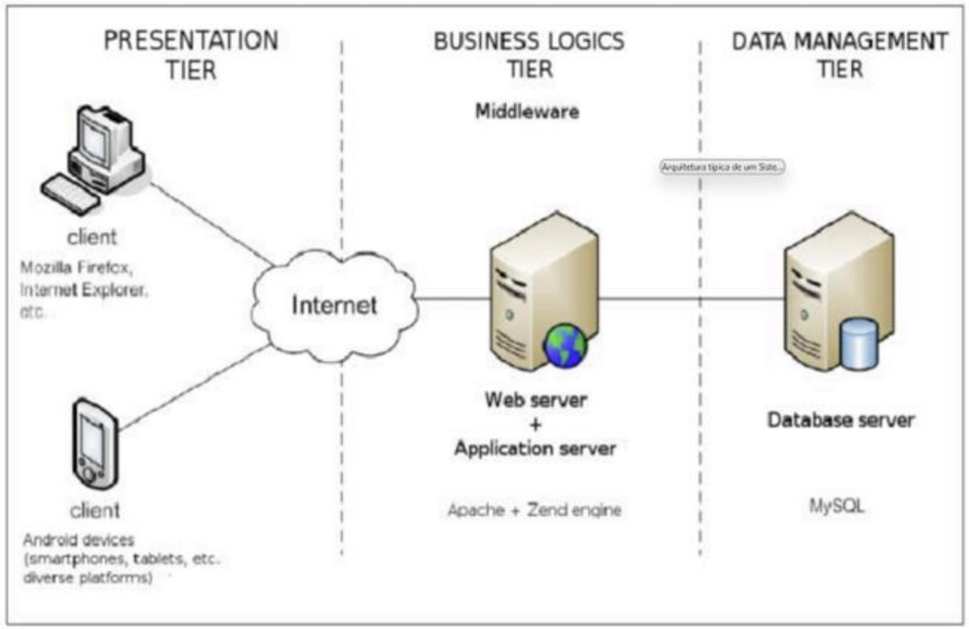
➔ Arquitetura “n-tier”

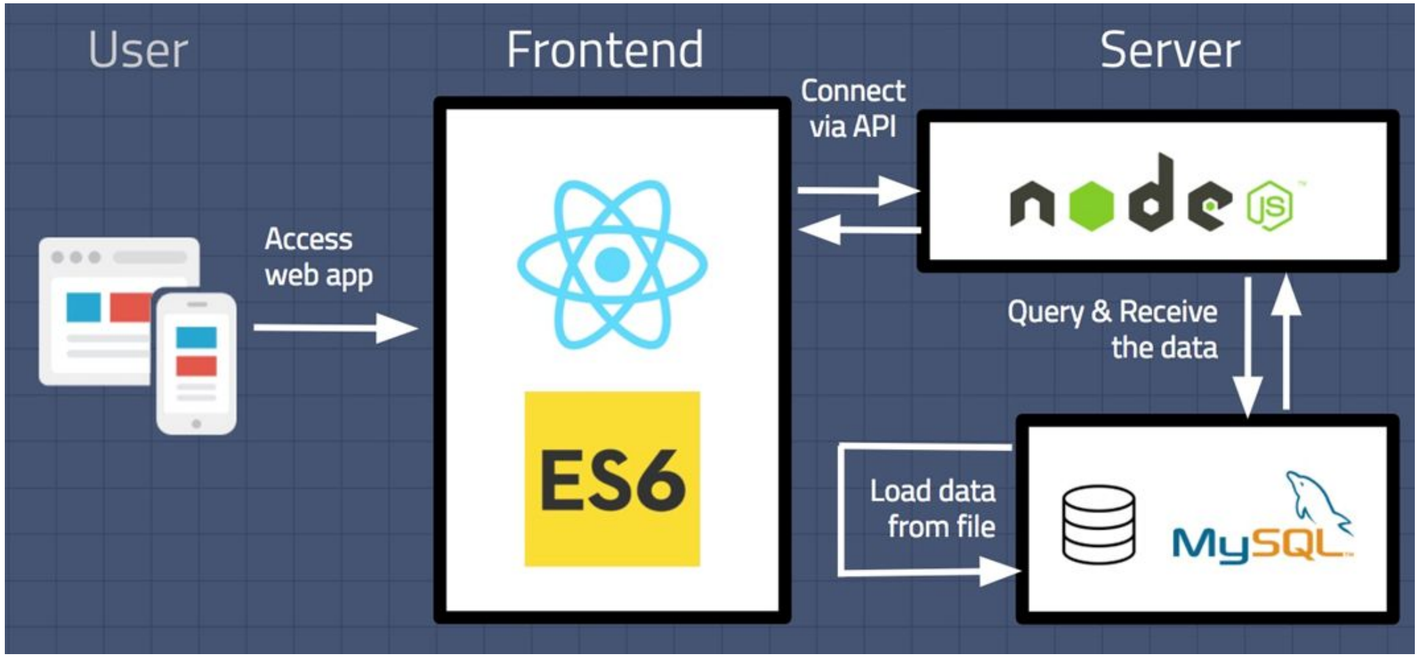


Arquitetura em Camadas

Centro de Informática
UFPE

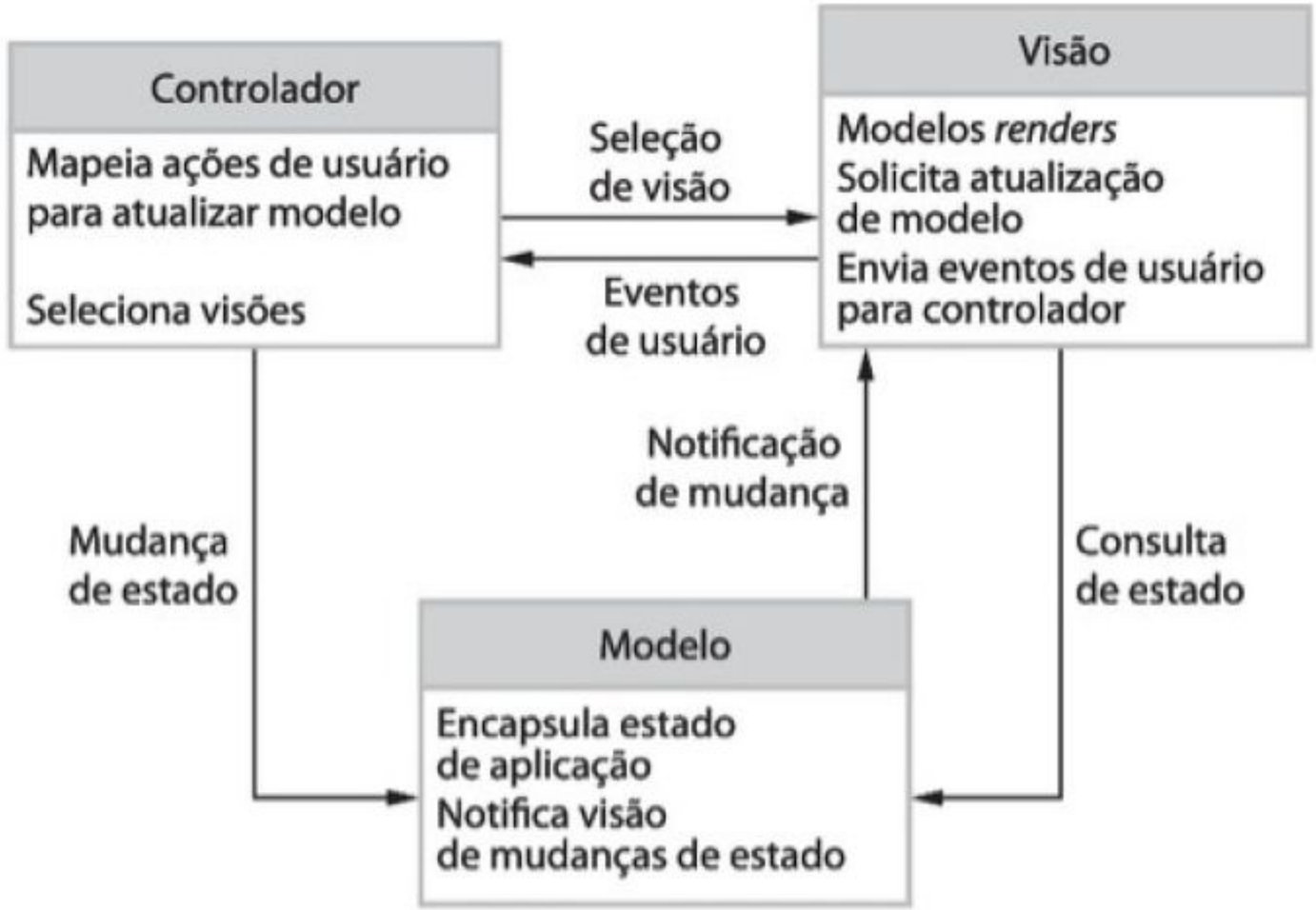
➔ Exemplificando o uso de tecnologias





O Clássico MVC

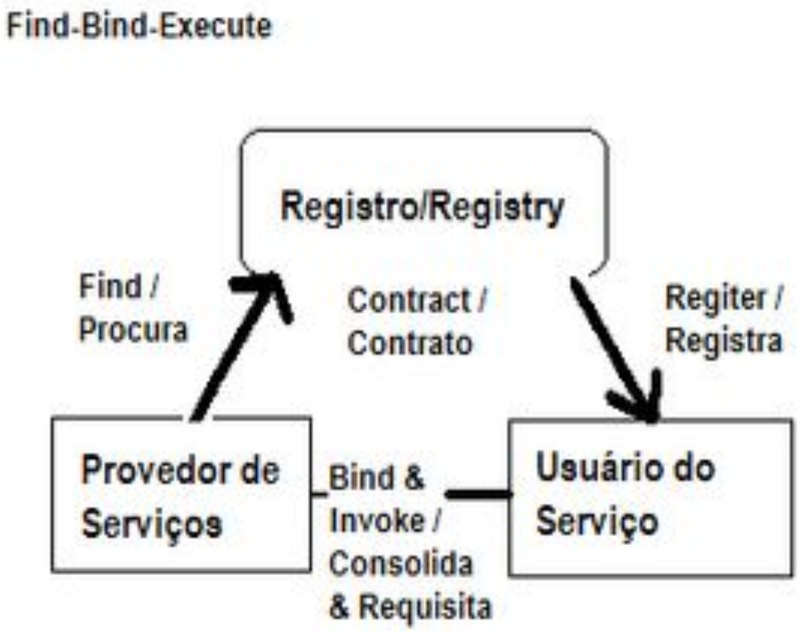
➔ Model-View-Controller



Arquitetura Orientada a Serviços

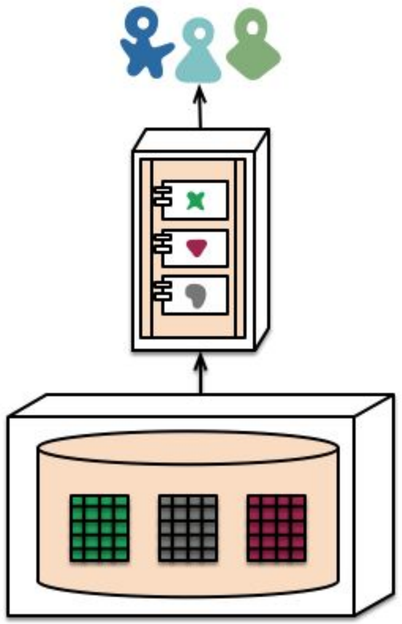
Centro de Informática
UFPE

- ➔ Integrar sistemas com serviços reutilizáveis
- ➔ Grandes sistemas corporativos como serviços
- ➔ Comunicação: SOAP, WSDL, XML

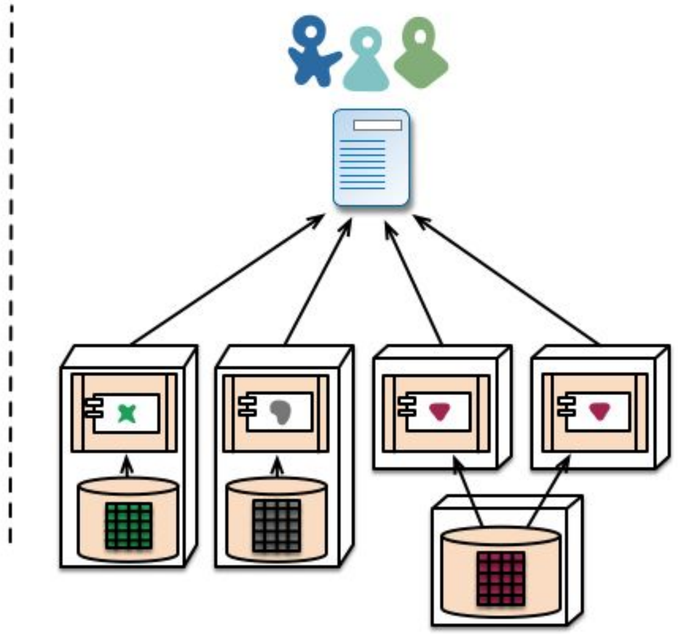


Microserviços

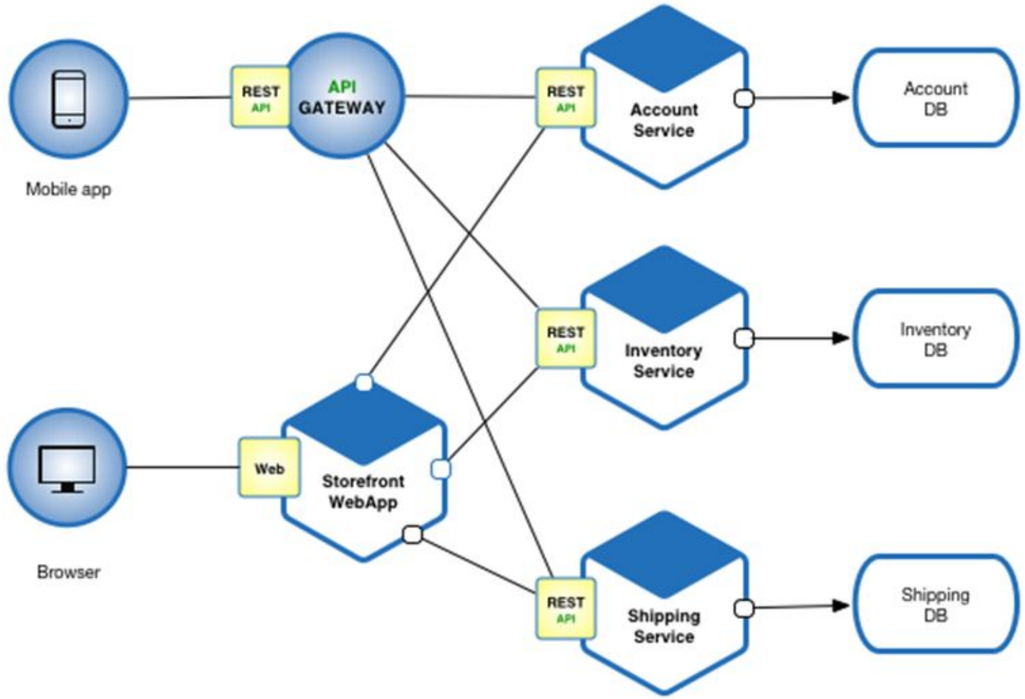
- ➔ Serviços pequenos, independentes e focados
- ➔ Organização interna de aplicações modernas
- ➔ Comunicação: HTTP/REST, gRPC, Eventos e Mensagens (JSON)



monolith - single database



microservices - application databases



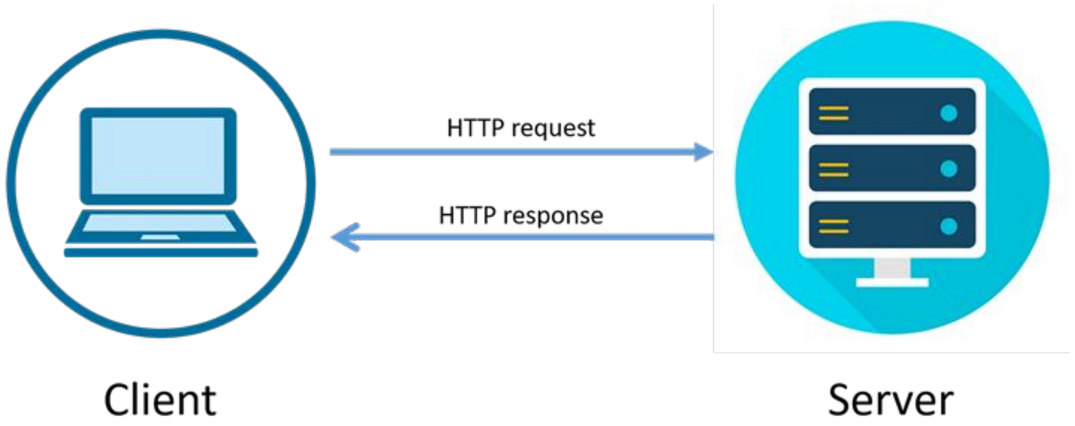
Centro de Informática
UFPE

Modelo/Paradigma de Comunicação

Cliente-Servidor

Centro de Informática UFPE

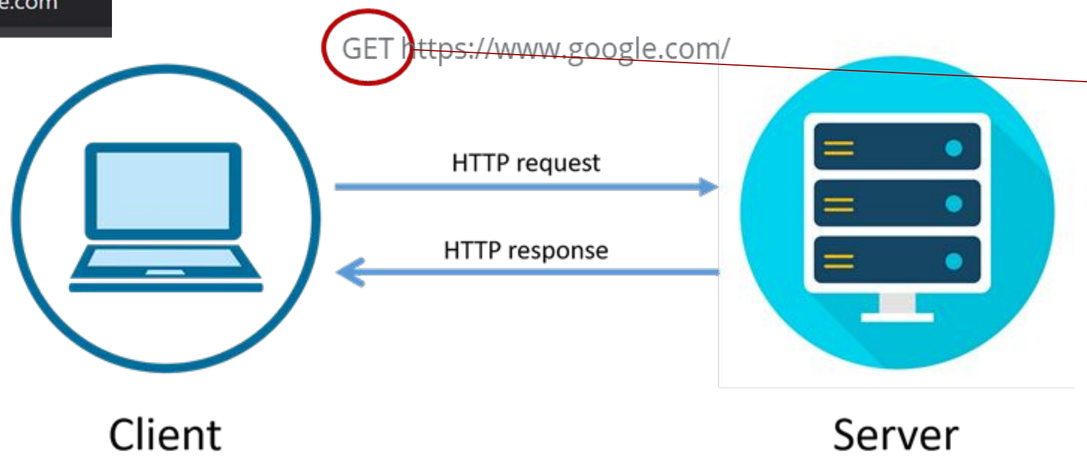
- ➔ Modelo de comunicação distribuída, base de muitas arquiteturas
- ➔ É a base tanto para aplicações monolíticas, quanto para microserviços
- ➔ Paradigma requisição/resposta (request/response)



Cliente-Servidor

Centro de Informática UFPE

- ➔ Modelo de comunicação distribuída, base de muitas arquiteturas
- ➔ É a base tanto para aplicações monolíticas, quanto para microserviços
- ➔ Paradigma requisição/resposta (request/response)



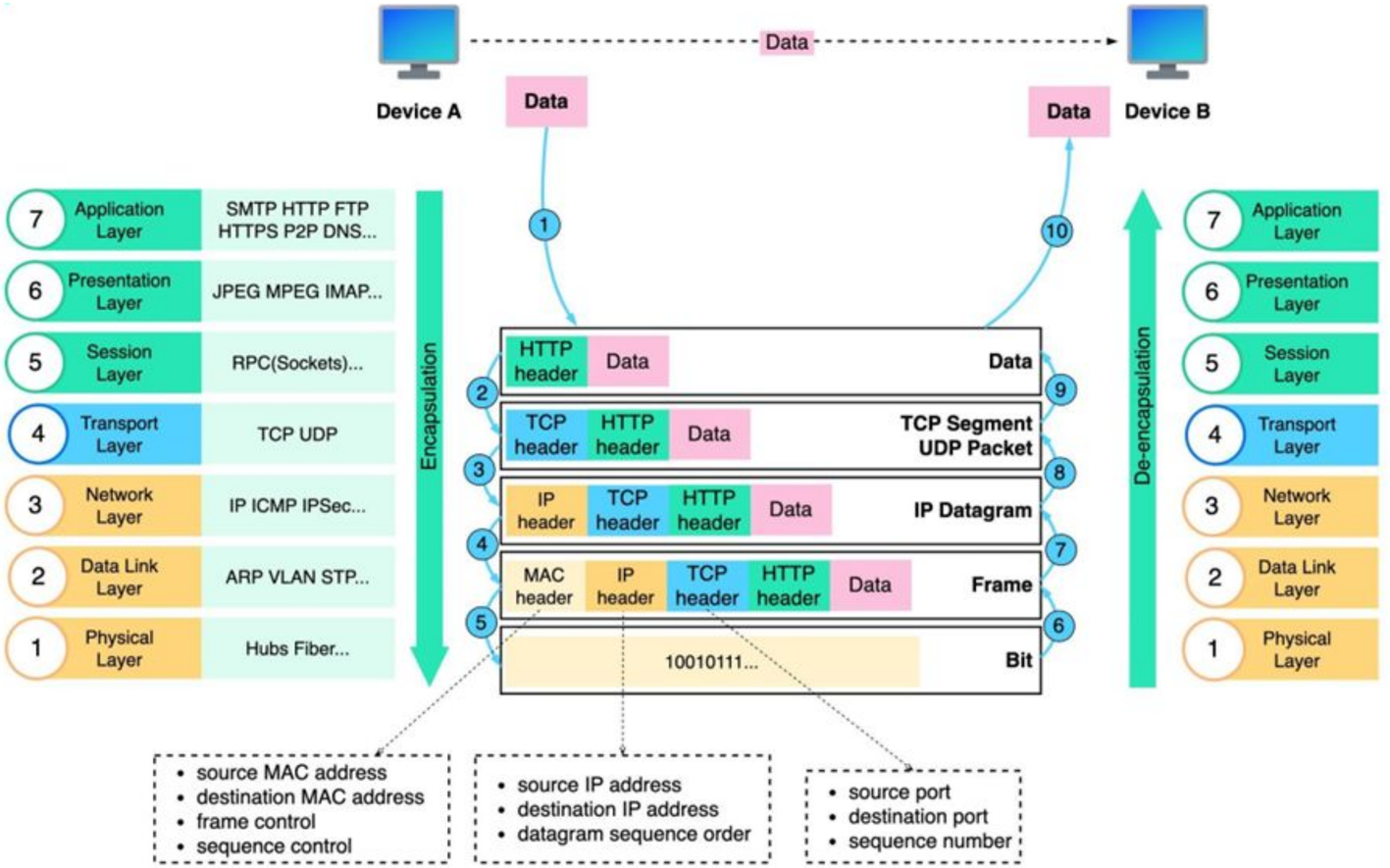
O HTTP possui o conceito de método (GET, POST, DELETE, etc.) que são equivalentes a funções que você pode chamar no servidor

O browser faz isso sozinho sem você se preocupar

Redes de Computadores em 5’

Centro de Informática
UFPE

➔ O encapsulamento do HTTP



Arquitetura Orientada a Mensagens

Centro de Informática UFPE

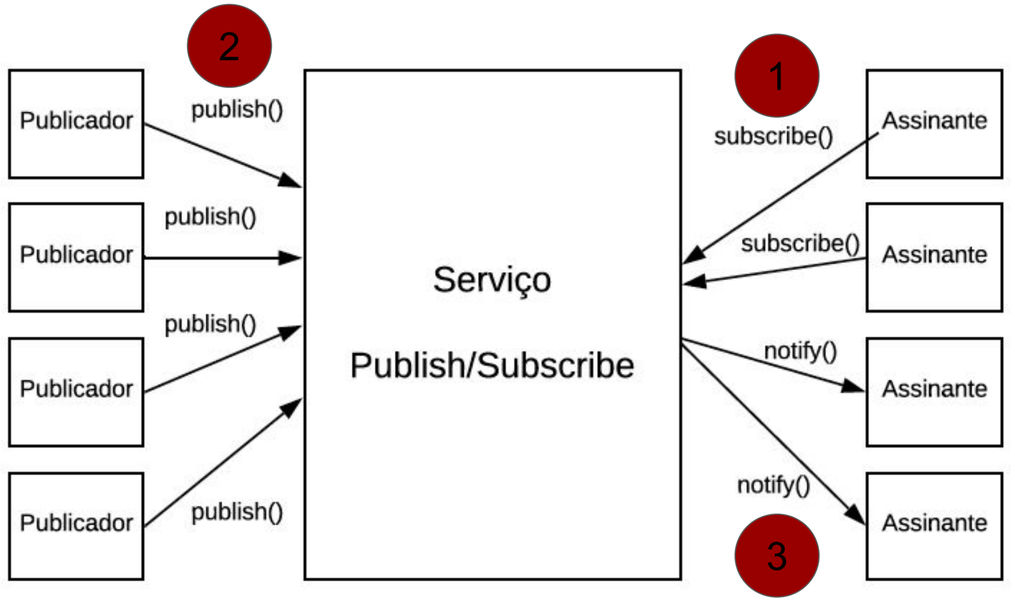
- Modelo de comunicação para aplicações distribuídas
- Clientes não se comunicam com o servidor, mas com um nó intermediário: fila de mensagens (ou Broker)
  - Tolerância a falhas
  - Escalabilidade
  - Comunicação Assíncrona
  - Desacoplamento no espaço e no tempo

Exemplo: RabbitMQ

Arquitetura Publish/Subscribe

Centro de Informática UFPE

- “Aperfeiçoamento” de fila de mensagens
- As mensagens agora são eventos
- Sistema podem: (1) assinar eventos; (2) publicar eventos; (3) serem notificados sobre a ocorrência de eventos



Exemplo: Kafka

Observações e Conclusões

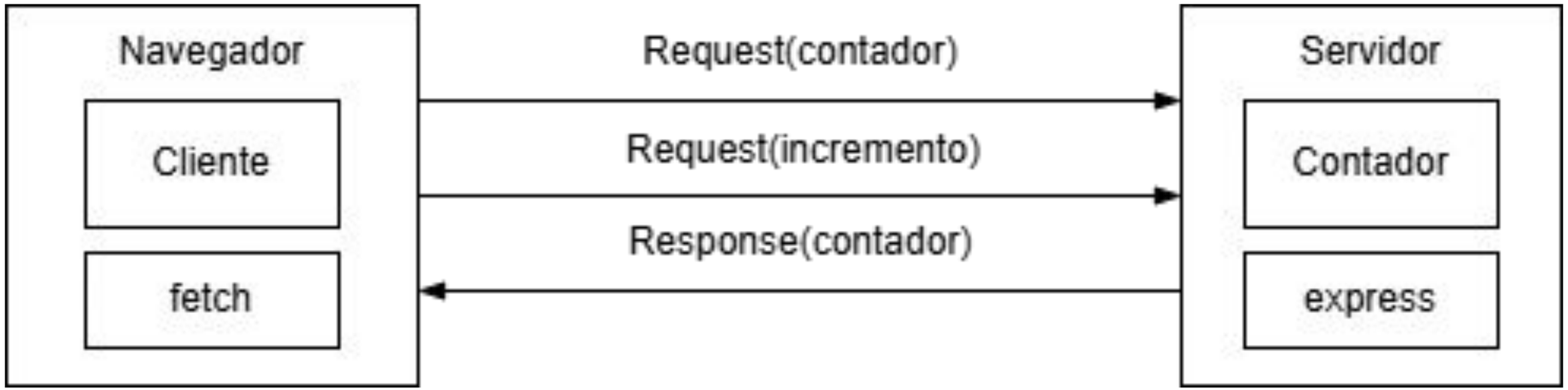
Centro de Informática
UFPE

- ➔ Agora entendemos o conceito de arquitetura de software
- ➔ Sabemos que existem diferentes estilos arquiteturais que podem ser utilizados para projetar a arquitetura do software
  - ◆ É possível utilizar soluções híbridas
- ➔ MVC é um padrão de arquitetura antigo, mas muito utilizado
- ➔ O estilo cliente-servidor também é bastante usado
- ➔ O estilo de microserviços está na crista da onda

Exemplo na prática

Centro de Informática
UFPE

- ➔ Arquitetura cliente-servidor + “camadas”
- ➔ Aplicação simples baseada “incrementador de número”, baseada na singles page application (SPA) do Marco Túlio:
  - https://engsoftmoderna.info/cap7.html



Centro de Informática
UFPE

Fim de Slides