

# Arquiteturas de Integração

CAPÍTULO 3. PRINCÍPIOS DE DESIGN DE SERVIÇOS

PROF. DIOVANI LUIZ MERLO

# Arquiteturas de Integração

---

AULA 3.1. INTRODUÇÃO AOS PRINCÍPIOS DE DESIGN DE SERVIÇOS

PROF. DIOVANI LUIZ MERLO

# Nesta aula

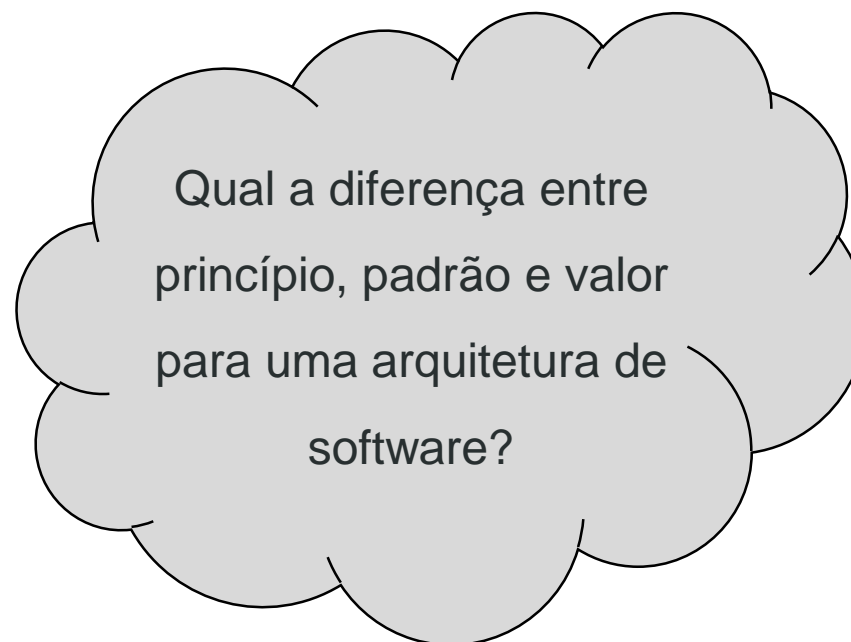
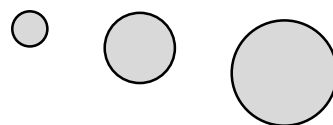


- ☐ Introdução a princípios de design de serviços.
- ☐ Tipos de princípios a serem considerados.

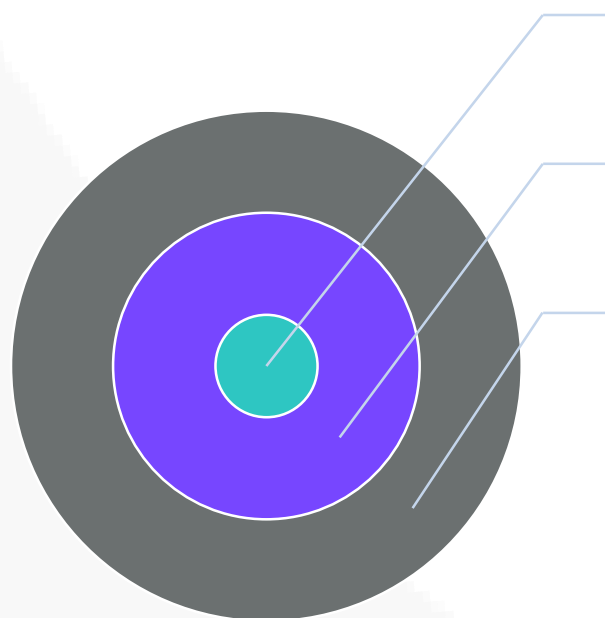
# Introdução

Princípios de design são práticas que definem diretrizes aceitas pelo mercado. Essas diretrizes são comparadas ao que também chamamos de melhores práticas (*blueprint*).

Em SOA, a construção de soluções orientadas a serviços de negócio que possam ser reutilizados, gerando rapidez e redução de custos de TI, é uma tarefa complexa para empresas que ainda não possuem um nível de maturidade adequado.



# Princípio vs Padrão vs Valor

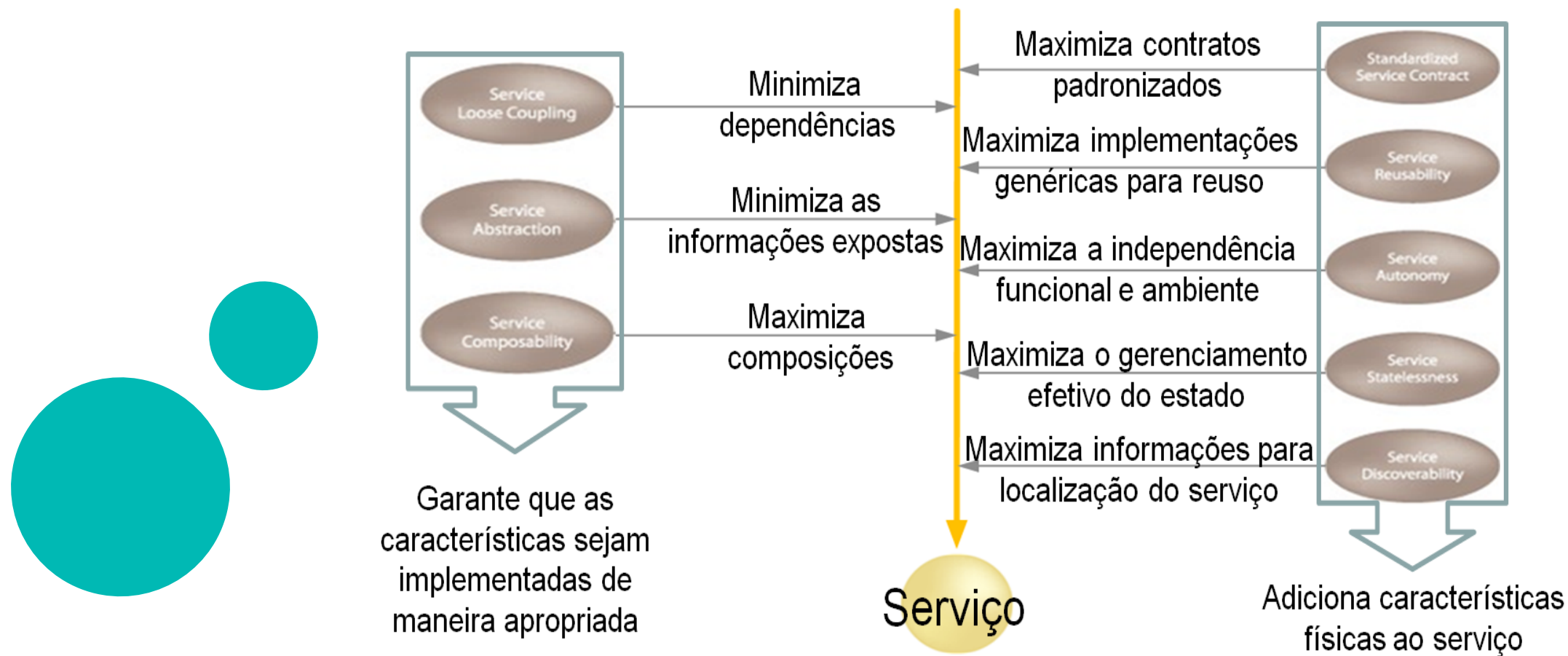


Tipo	Definição
Padrão	Soluções para problemas recorrentes. Aplicável em contextos específicos.
Princípio	Guia geral de boas práticas. Associado a paradigmas de programação.
Valor	Uso universal para definir se uma solução traz resultados.



Por que deveria me  
preocupar com o uso  
de princípios de  
design de serviços?

# Tipos de Princípios para Serviços



Fonte: ERL (2015)



# Conclusão



- ✓ Conhecer e utilizar princípios de design de serviços minimiza a complexidade em se desenvolver arquiteturas orientadas a serviços que possam ser evoluídas e que agregam valor ao longo do seu ciclo de vida.

# Próxima aula



01.

Princípio de design de serviço:  
padronização de contratos.

02.

Princípio de design de serviço:  
abstração.

03.

Princípio de design de serviço:  
baixo acoplamento.

04.

Princípio de design de serviço:  
reúso.

# Arquiteturas de Integração

---

AULA 3.2. PRINCÍPIOS: PADRONIZAÇÃO, BAIXO ACOPLAMENTO, ABSTRAÇÃO E REÚSO

PROF. DIOVANI LUIZ MERLO

# Nesta aula

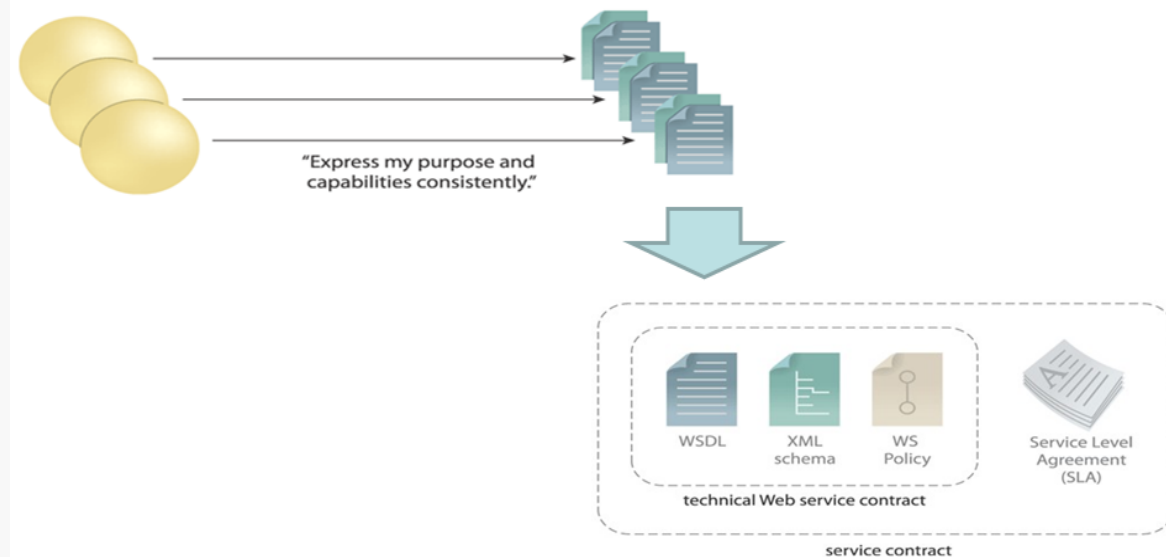


- ☐ Princípio de design de serviço.
- ☐ Padronização.
- ☐ Baixo acoplamento.
- ☐ Abstração .
- ☐ Reúso.

# Padronização de contratos



Em soluções orientadas a serviços, o contrato tem papel fundamental para que os objetivos de uma arquitetura SOA sejam seguidos, pois será esse contrato que definirá para seus clientes tudo o que o serviço fará.



Fonte: ERL (2015)



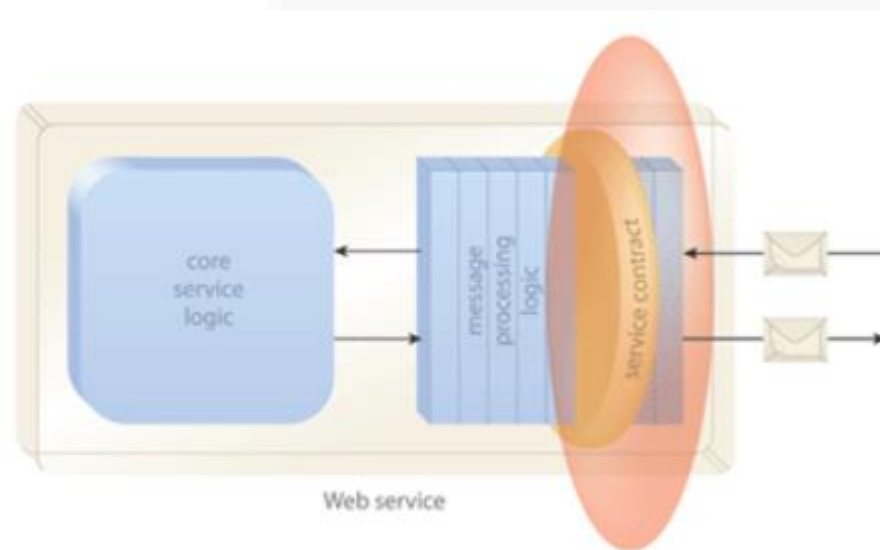
Qual o papel de um  
contrato bem definido  
para um serviço?

# Padronização de contratos



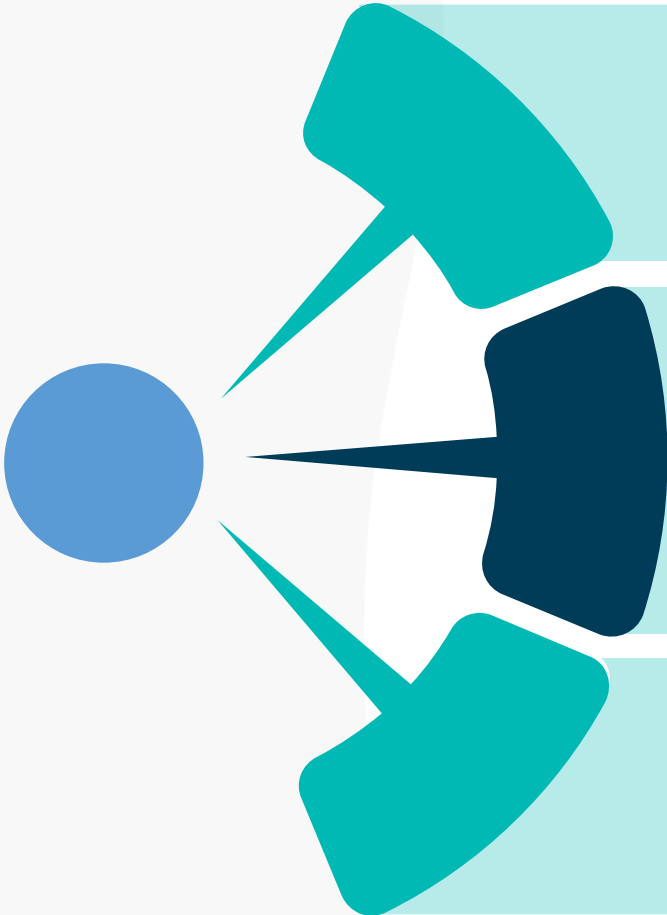
O papel fundamental deste princípio é garantir que os objetivos e capacidades dos serviços sejam definidos de forma consistente.

Os contratos dos serviços, dentro do contexto SOA, devem ser compostos de elementos técnicos e também elementos não técnicos.



Fonte: ERL (2015)

# Padronização: o que abordar



Informações técnicas de entrada de um serviço como operações e seus atributos/tipos.

Em serviços REST adicionar links para as URIs.

Informações de negócio sobre o propósito do serviço, descrevendo suas operações e atributos.

Informações sobre níveis de acordo (SLA).

Responsável pelo serviço e suas informações de contato.



# Padronização de contratos



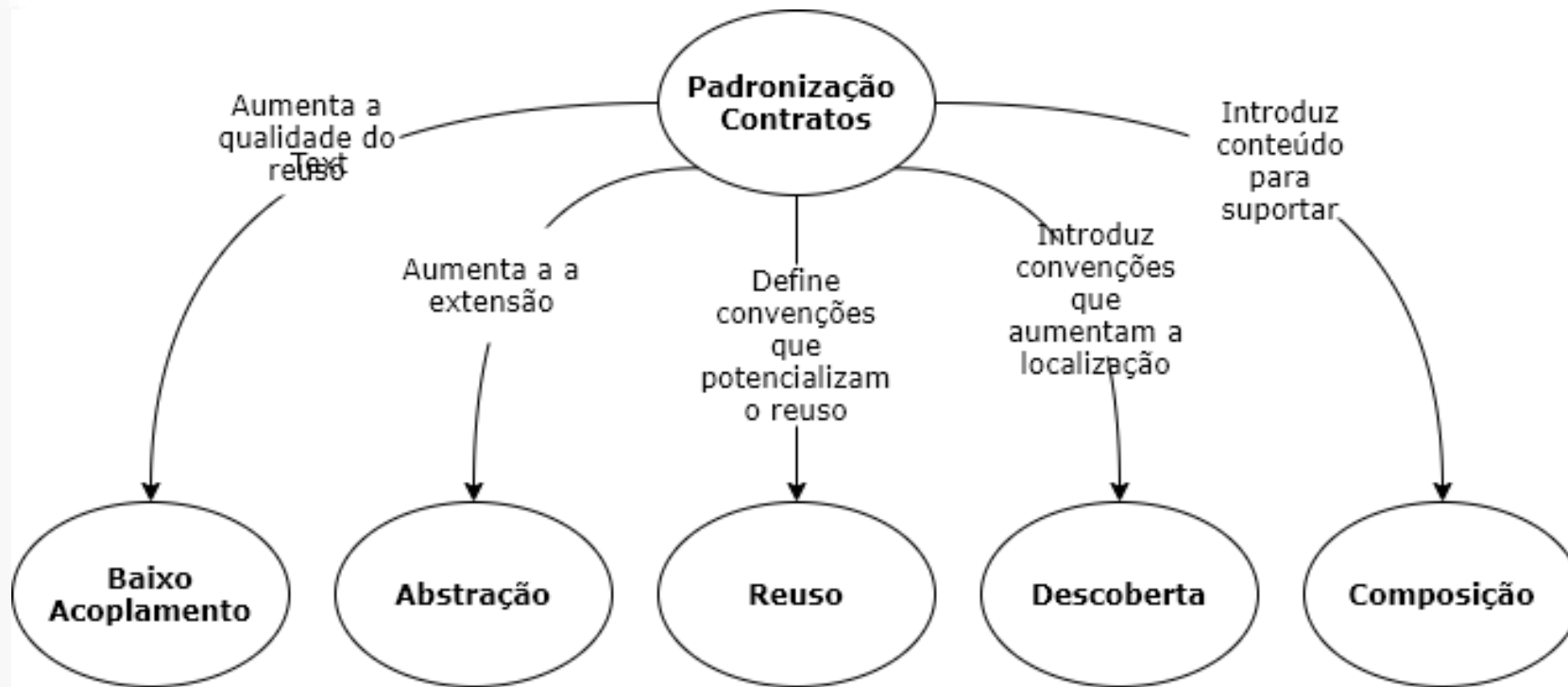
Em serviços REST, uma boa prática é adicionar links para as URIs dos serviços.

Esse modelo é baseado no HATEOAS (*Hypermedia as the Engine of Application State*).

Fonte: RICHARDSON (2018)

```
{
  "links": {
    "self": { "href": "http://api.com/items" },
    "item": [
      { "href": "http://api.com/items/1" },
      { "href": "http://api.com/items/2" }
    ]
  },
  "data": [
    { "itemName": "a" },
    { "itemName": "b" }
  ]
}
```

# Como o padrão afeta outros

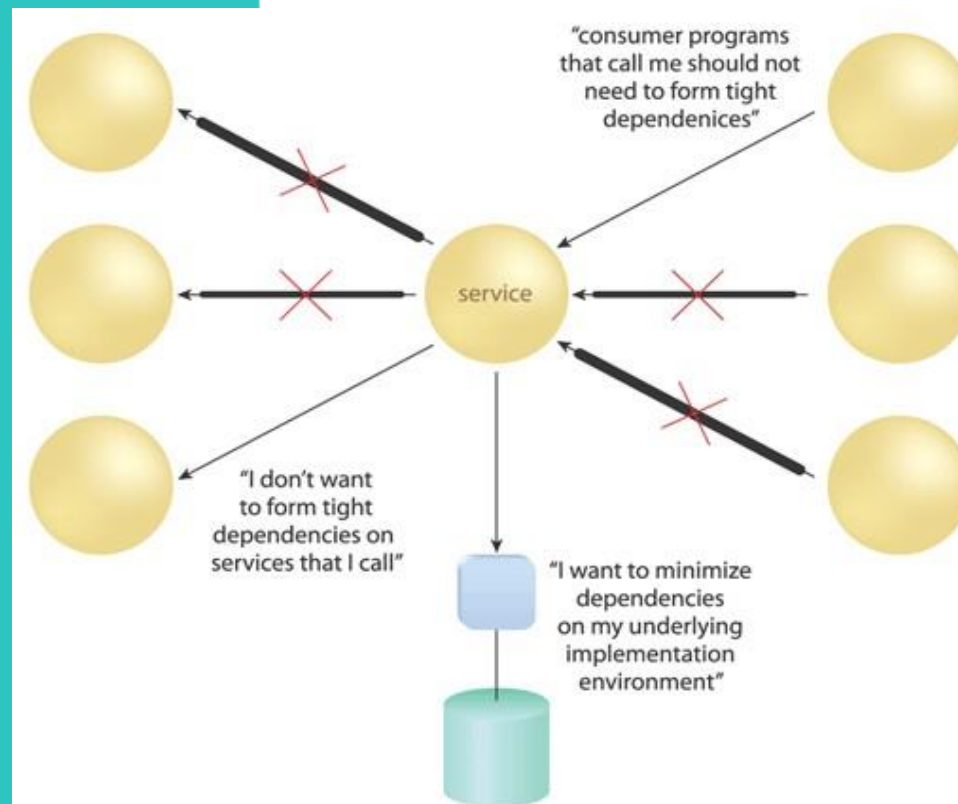


Fonte: Figura adaptada (ERL, 2015)

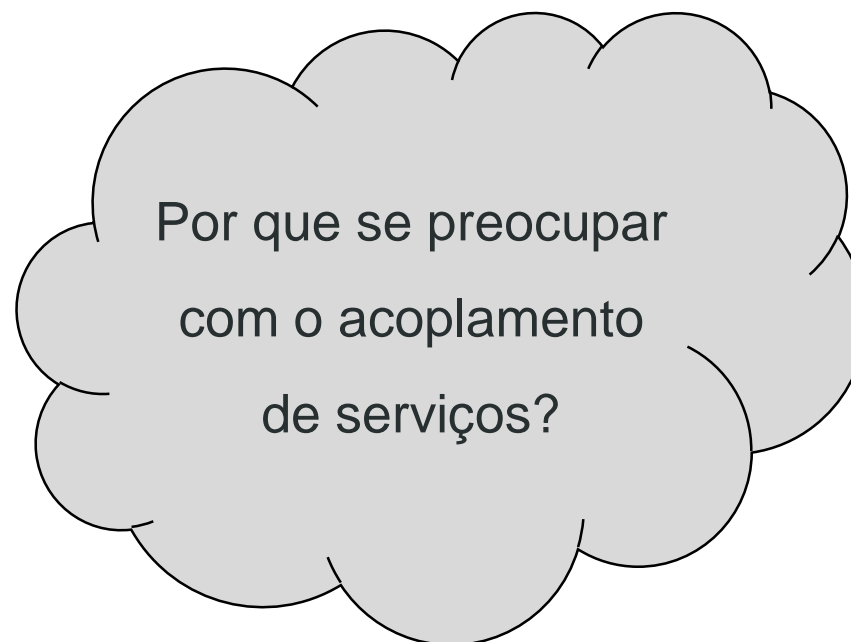
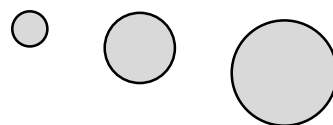
# Baixo acoplamento

Este princípio dá ênfase à redução do acoplamento entre os serviços, ou seja, busca minimizar o relacionamento dos serviços desde o seu contrato até a sua lógica.

Um exemplo de acoplamento tradicional são as aplicações monolíticas que executam todos os comandos/transações em bancos de dados a partir de suas lógicas de negócio centralizadas.



Fonte: ERL (2015)



# Tipos de acoplamento

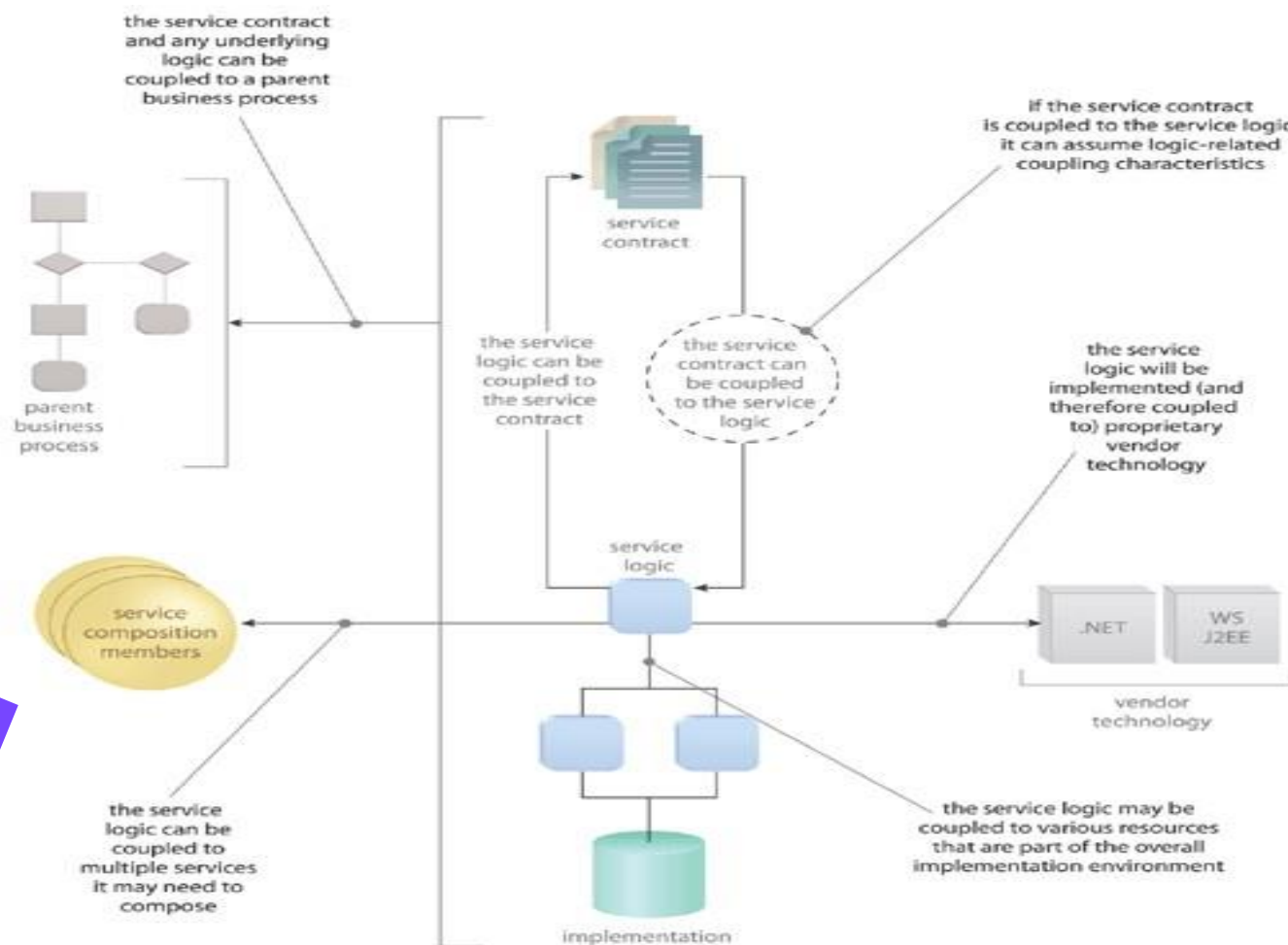
IGTI

Acoplamento de processo

Acoplamento de contrato

Acoplamento de outros serviços

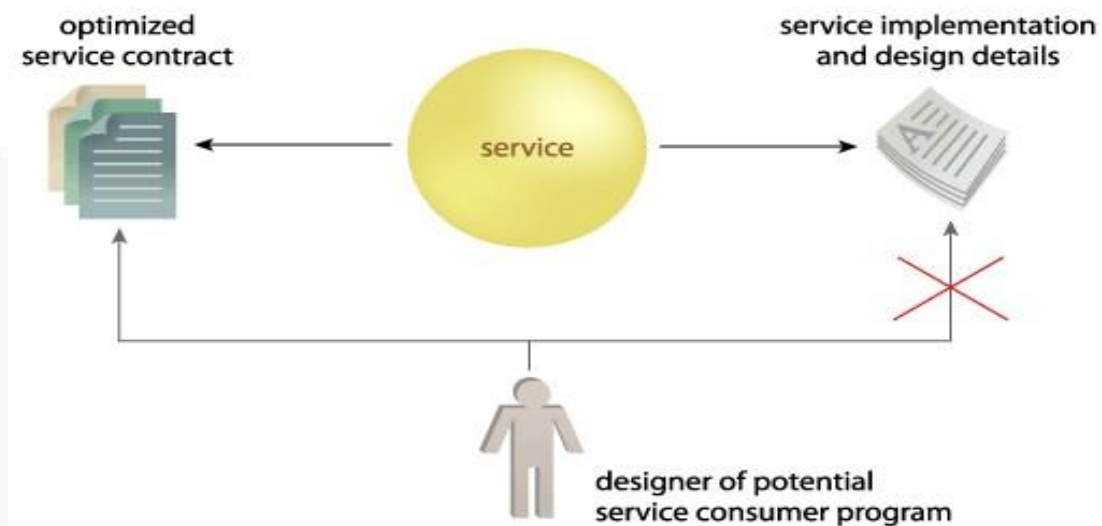
Acoplamento fornecedor tecnologia



Fonte: ERL (2015).

# Abstração

Tem por objetivo diminuir a exposição de informações desnecessárias do serviço para os seus clientes, pois quanto maior for o número de informações expostas por seu contrato, maior será o acoplamento com os clientes deste serviço.



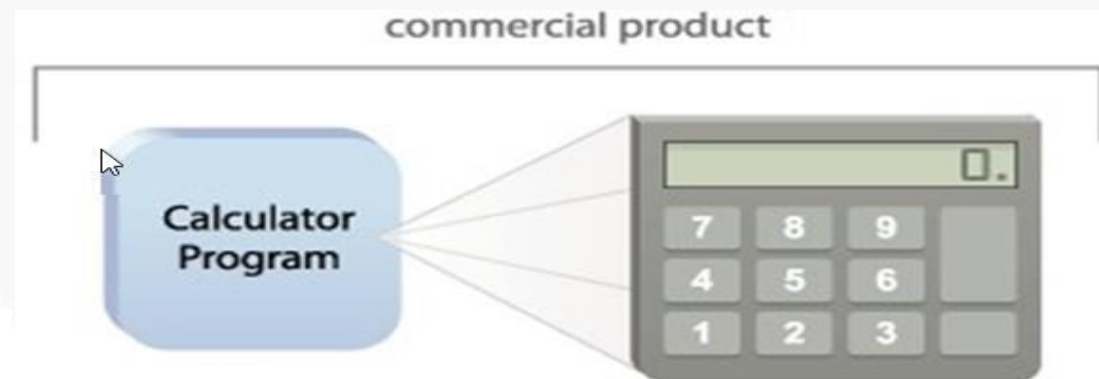
Fonte: ERL (2015)

# Abstração



As pessoas não têm necessidade de saber qual foi a linguagem de programação, como foi desenhado e sua qualidade de serviço é estabelecida em seu manual.

- Forte abstração de tecnologia.
- Forte abstração funcional.
- Completa abstração da lógica.
- Forte abstração da qualidade de serviço.



Fonte: ERL (2015)

# Abstração



Por exemplo, um serviço que expõe no contrato a necessidade que o cliente que o consuma tenha que dominar um domínio discreto para um determinado atributo, relativo a categoria do cliente (0 – Ouro, 1 – Prata, 2 – Bronze etc.):

```
User:
  type: "object"
  properties:
    id:
      type: "integer"
      format: "int64"
    category:
      type: "integer"
      format: "int64"
      description: "OURO = 0, PRATA = 1, BRONZE = 2"
    username:
      type: "string"
    firstName:
      type: "string"
    lastName:
      type: "string"
    email:
      type: "string"
    password:
      type: "string"
    phone:
      type: "string"
    userStatus:
      type: "integer"
      format: "int32"
      description: "User Status"
```

**POST** /user Create user

```
{
  "id": 0,
  "category": 0,
  "username": "string",
  "firstName": "string",
  "lastName": "string",
  "email": "string",
  "password": "string",
  "phone": "string",
  "userStatus": 0
}
```

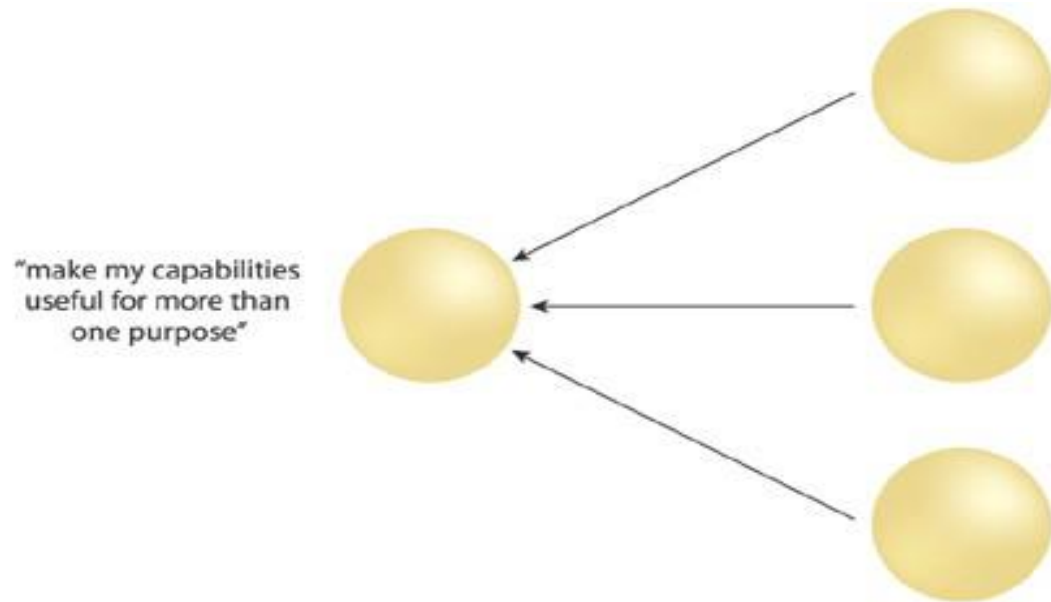


# Reúso

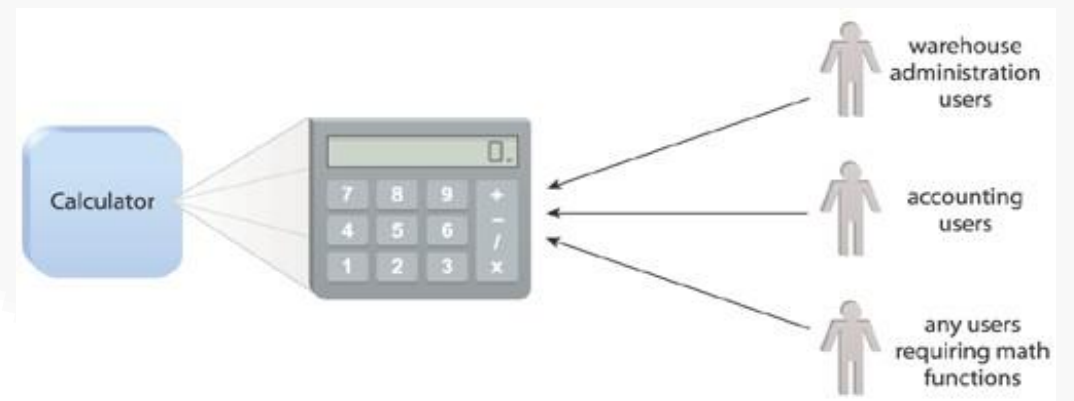
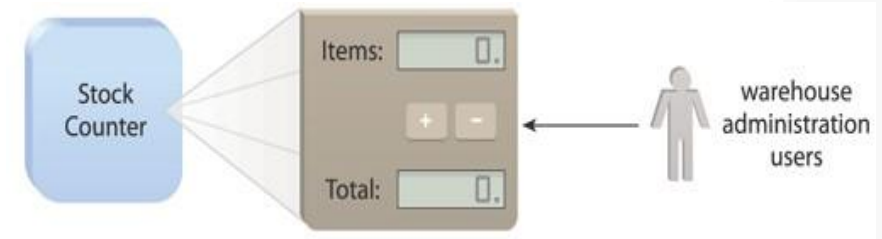
Este princípio tem como principal objetivo garantir que os serviços sejam especificados e construídos de forma a possibilitar ao máximo a sua utilização por diversos clientes, em diferentes contextos de negócio, por maior tempo possível.

Em uma arquitetura SOA, o reúso é um dos principais motivadores para as empresas, tendo em vista que isso aumenta o retorno de investimento (ROI) das soluções.

# Reúso



IGTI



# Conclusão



- ✓ Os princípios de design são um guia muito importante para que uma arquitetura de integrações orientada a serviços sejam construídas com consistência e robustez.

# Próxima aula



01.

Princípio de design: autonomia.

02.

Princípio de design: descoberta.

03.

Princípio de design: estado.

04.

Princípio de design: composição.

# Arquiteturas de Integração

---

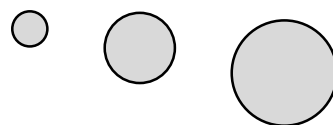
AULA 3.3. PRINCÍPIOS: AUTONOMIA, ESTADO, DESCOBERTA E COMPOSIÇÃO

PROF. DIOVANI LUIZ MERLO

# Nesta aula



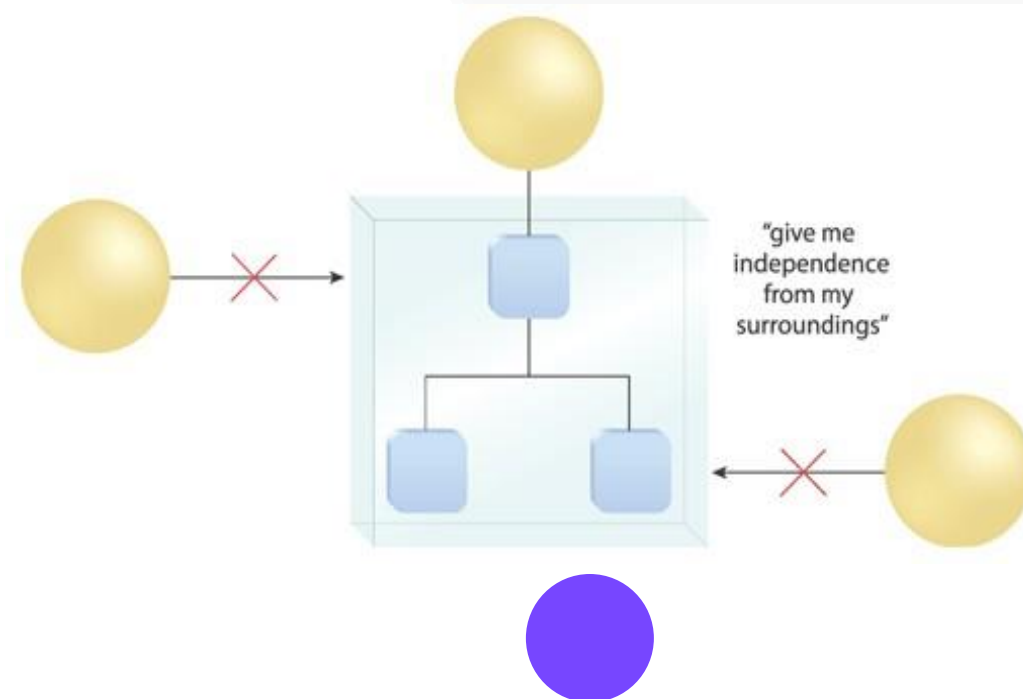
- ☐ Princípios de design de serviços.
- ☐ Autonomia.
- ☐ Estado.
- ☐ Descoberta.
- ☐ Composição.



Como a autonomia dos  
serviços pode impactar  
a arquitetura de uma  
solução?

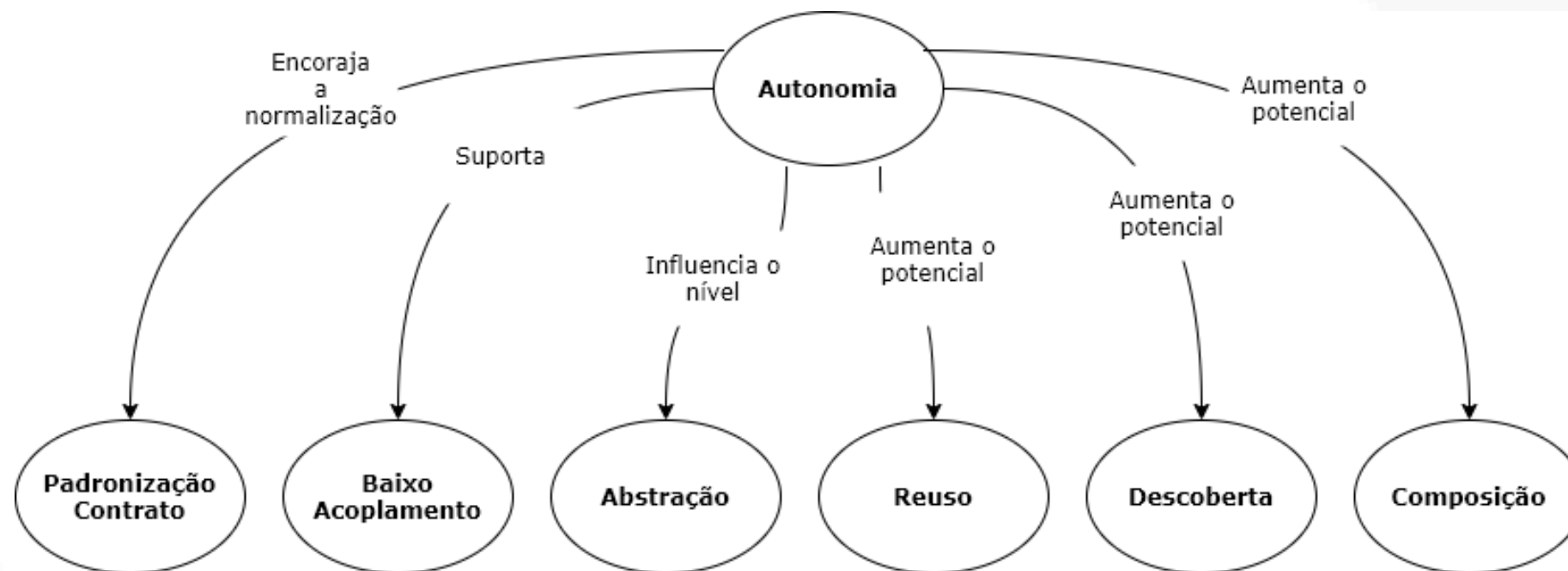
# Autonomia

Este princípio representa, para o serviço, a habilidade de independência ou influência de recursos externos para que sua lógica seja executada. Essa autonomia está ligada diretamente a questões como desempenho e interoperabilidade, ou seja, poder ser executado mesmo em plataformas distintas.





# Autonomia

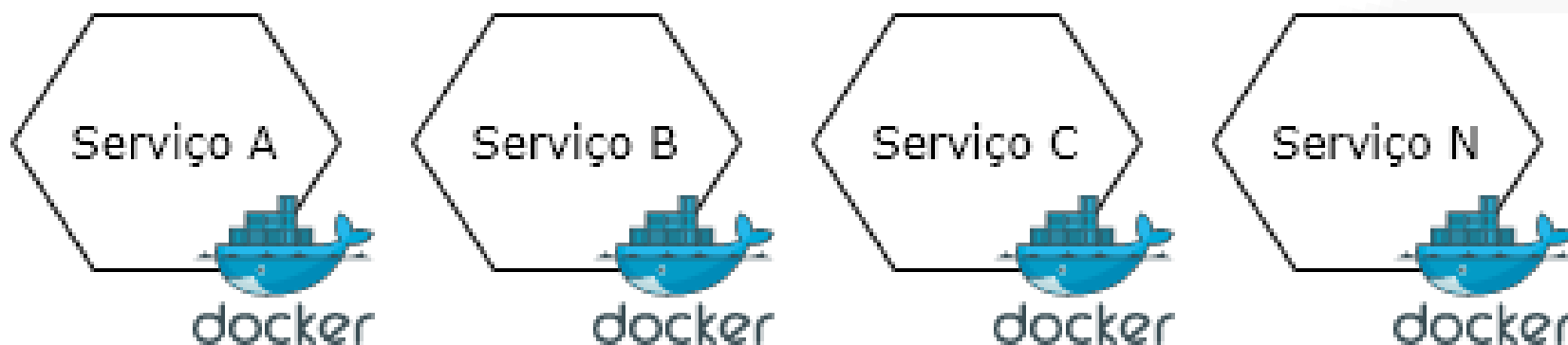


Fonte: Figura adaptada (ERL, 2015)

# Autonomia



Serviços que podem ser movidos para ambientes diferentes, mantendo suas funcionalidades.

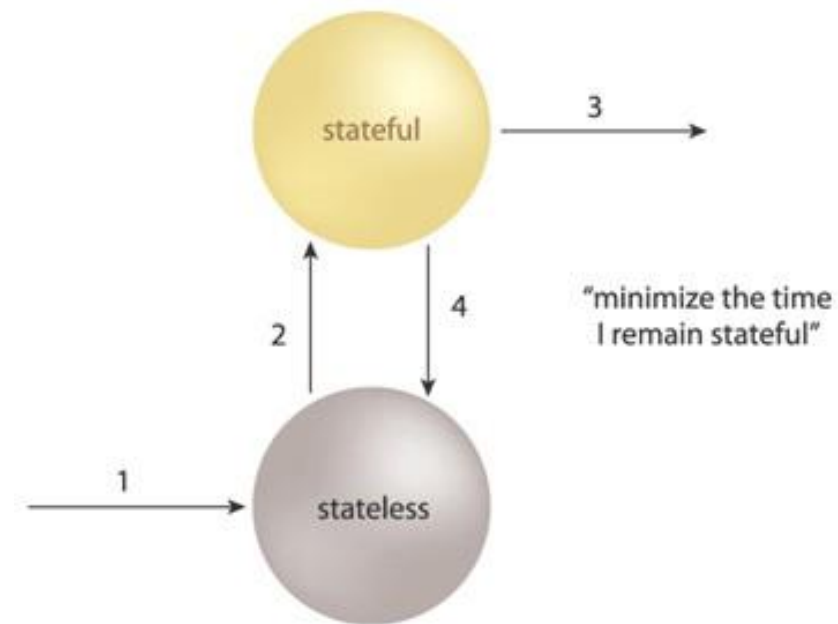


Serviços onde seu desempenho não depende de fatores externos.

# Estado

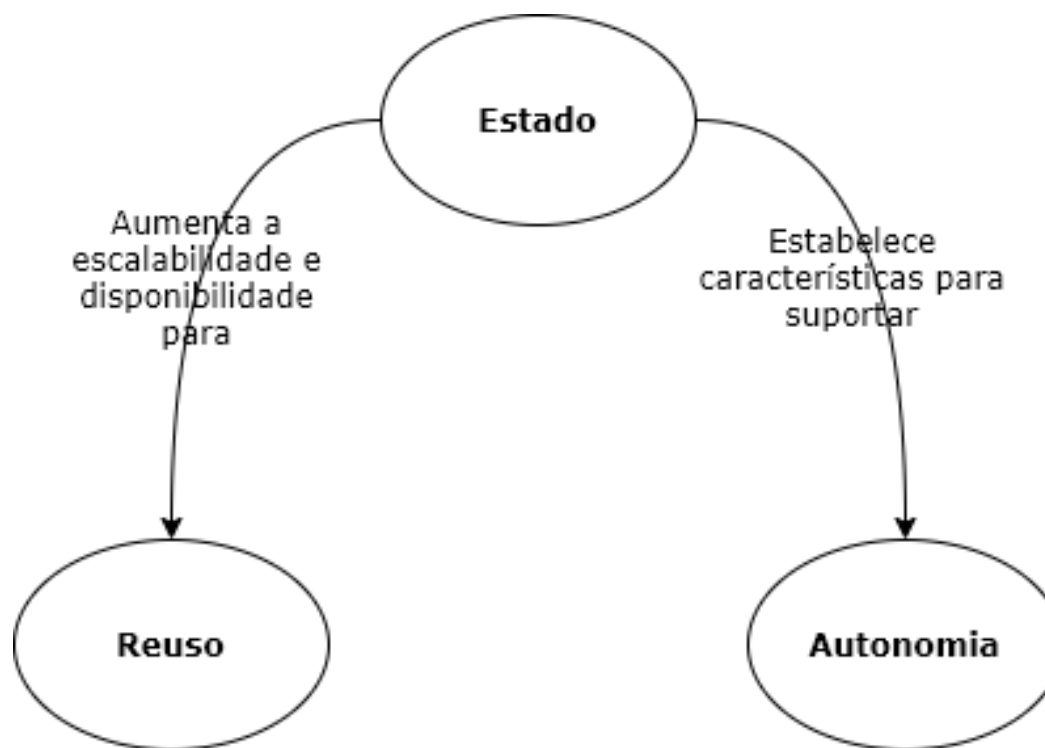
Necessidade de otimização da lógica de processamento do serviço quando se torna necessário suportar requisitos de acessos múltiplos por parte de programas clientes, buscando consumir uma quantidade mínima de recursos durante sua utilização.

Esse princípio busca otimizar ao máximo a alocação de recursos durante sua execução para permitir escalabilidade ao mesmo, favorecendo diretamente seu reúso.

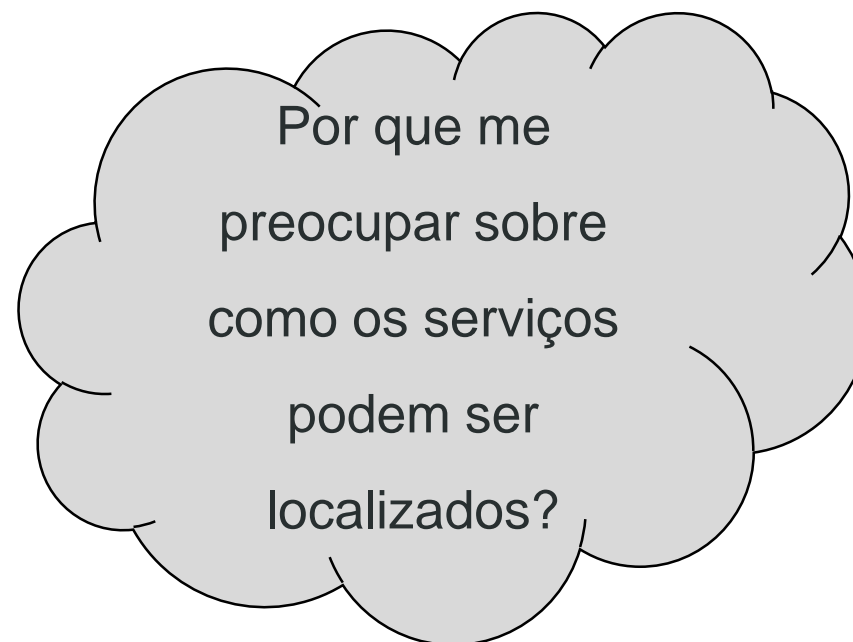
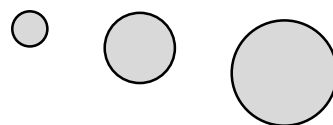


Fonte: ERL (2015)

# Estado



Fonte: Figura adaptada (ERL, 2015)



# Descoberta

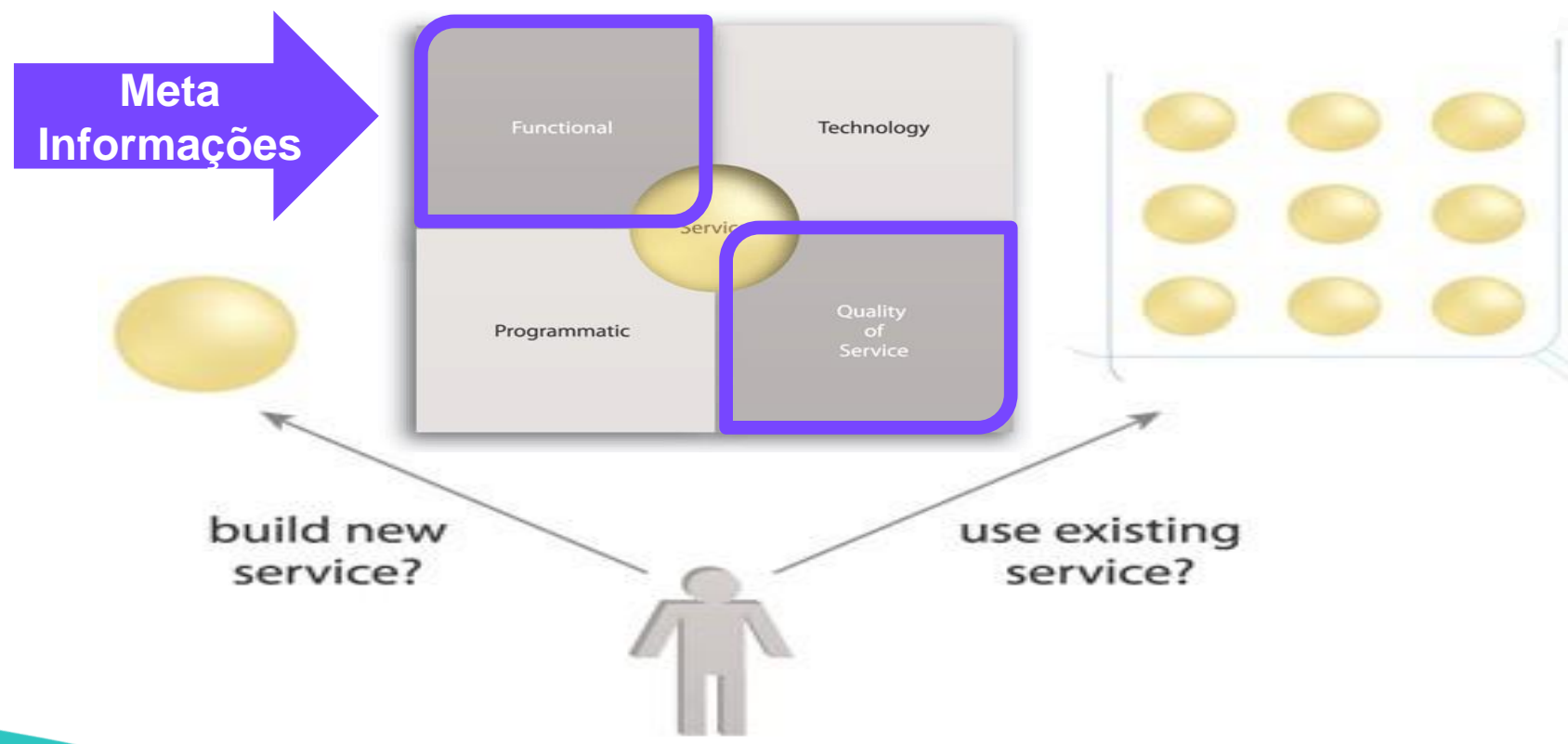
Este princípio busca trabalhar as informações associadas ao serviço e que são de grande importância na sua localização pelos clientes durante a execução de processos de negócio ou tempo de desenvolvimento.

Se as informações não são corretamente definidas, um time de desenvolvimento pode ser levado ao erro e a construir um novo serviço somente pelo fato de que não localizou o existente.

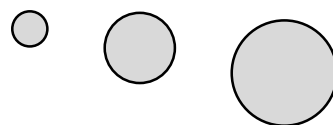


# Descoberta

iGTi



Fonte: ERL (2015)



Macro, mini,  
microserviços. Por  
que o tamanho pode  
influenciar tanto?



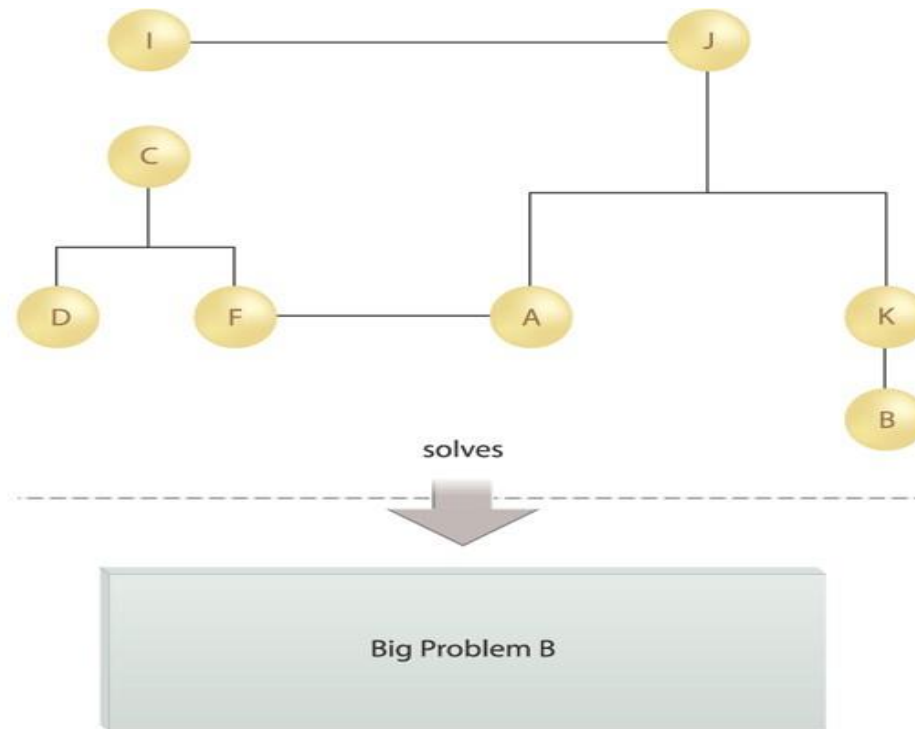
# Composição



O reúso é um dos aspectos mais importantes para uma arquitetura orientada a serviços, e o princípio de composição de serviços está diretamente alinhado com esse objetivo, pois permite que serviços sejam agrupados ou compostos para atendimento de uma determinada necessidade de negócio que não poderia ser atendida por esses de forma independente.

# Composição

Combinar diferentes serviços para atender a necessidade do negócio.



Fonte: ERL (2015)

# Conclusão



- ✓ Aplicar corretamente os princípios de design de serviços permitirá minimizar a complexidade de liberação de versões iniciais de arquiteturas de integração que sofrerão menos refatorações nas próximas versões.

# Próxima aula

01.

Introdução a microsserviços.

IGTi