

UNIVERSIDADE CATÓLICA DE PELOTAS  
ENGENHARIA DE COMPUTAÇÃO  
SISTEMAS DISTRIBUÍDOS

# Sistemas Publicar-Assinar

MATHEUS GONÇALVES STIGGER

---

1º/07/2022



- Os *sistemas publicar-assinar*, também são referidos como *sistemas baseados em eventos distribuídos*. Essa é a mais amplamente utilizada de todas as técnicas de comunicação indireta discutidas no capítulo 6 do nosso livro-texto.
- *Publicadores* divulgam eventos para um serviço de evento.
- *Assinantes* expressam interesse em eventos específicos por meio de *assinaturas*.



- *Heterogeneidade:*

Fazer com que componentes de um sistema distribuído que não foram projetados para operação conjunta funcionem juntos.

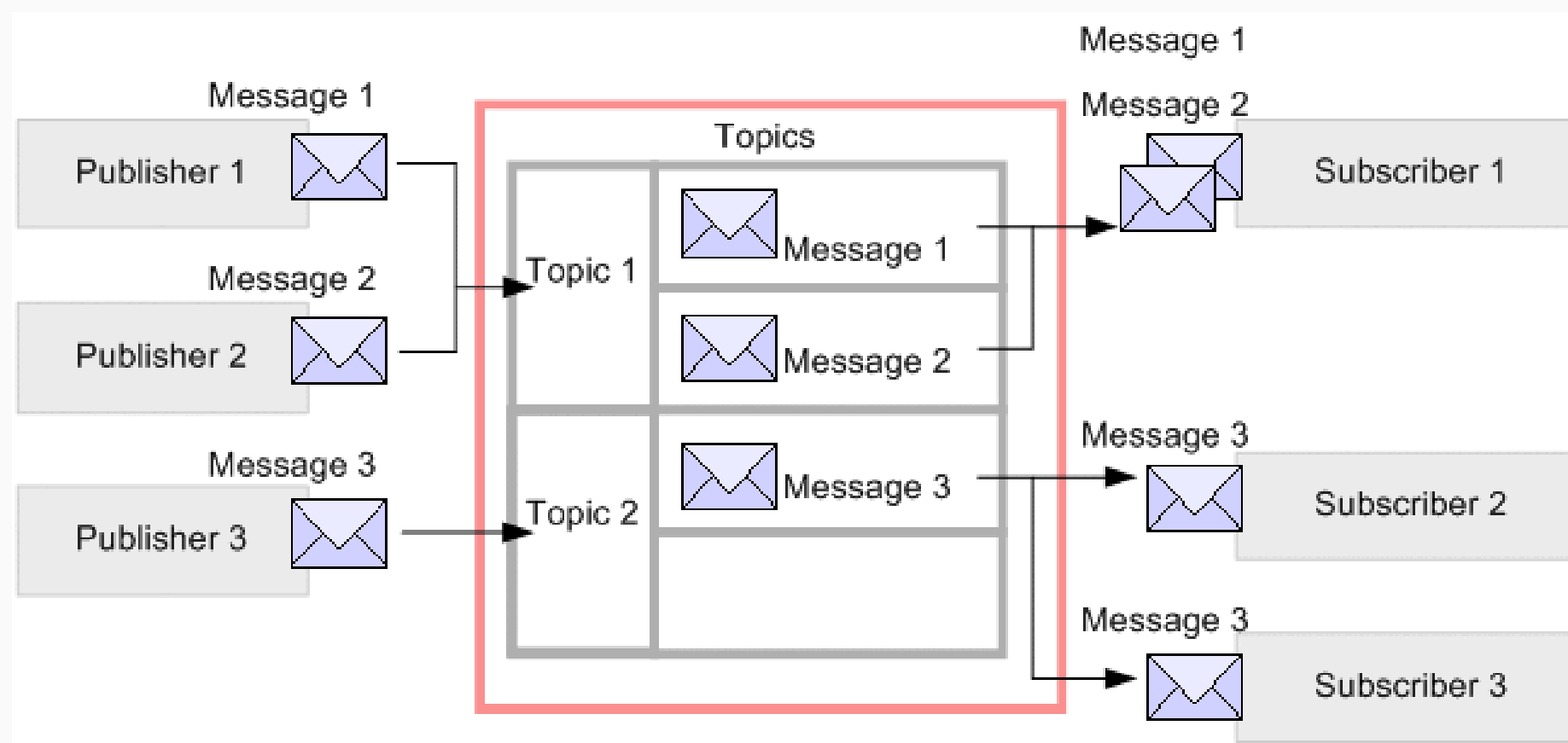
- *Assíncronos:*

Evitar que os publicadores precisem estar sincronizados com os assinantes.

# Modelo de Programação

4

- O modelo de programação em sistemas publicar-assinar é baseado em um pequeno conjunto de operações, mostrado na figura abaixo.
- Os publicadores disseminam um evento por meio de uma operação *publish(evento)*.
- Os assinantes expressam interesse em um conjunto de eventos por meio de assinaturas: operação *subscribe(f)*, onde *f* se refere a um filtro.



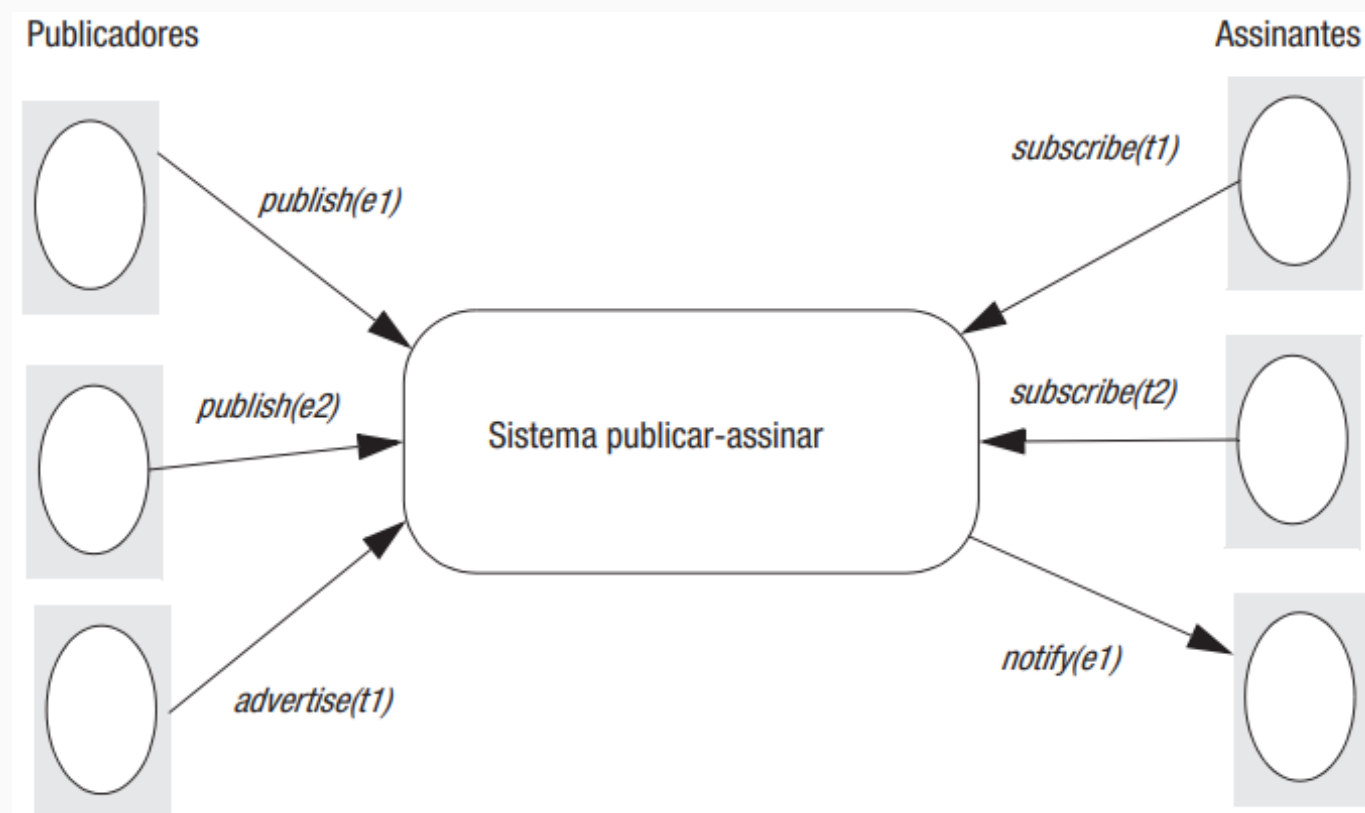
- Modelo de assinatura (filtro), com vários esquemas definidos:
  - *Baseado em canal*: os publicadores divulgam eventos para canais nomeados e, então, os assinantes se inscrevem em um deles para receber todos os eventos enviados para esse canal.
  - *Baseado em tópico*: cada notificação é expressa em termos de vários campos, com um deles denotando o tópico. Então, as assinaturas são definidas em termos do tópico de interesse.
  - *Baseado em conteúdo*: permitem a expressão de assinaturas sobre diversos campos em uma notificação de evento.
  - *Baseado em tipo*: as assinaturas são definidas em termos de tipos de eventos e a combinação é definida em termos de tipos ou subtipos do filtro dado.



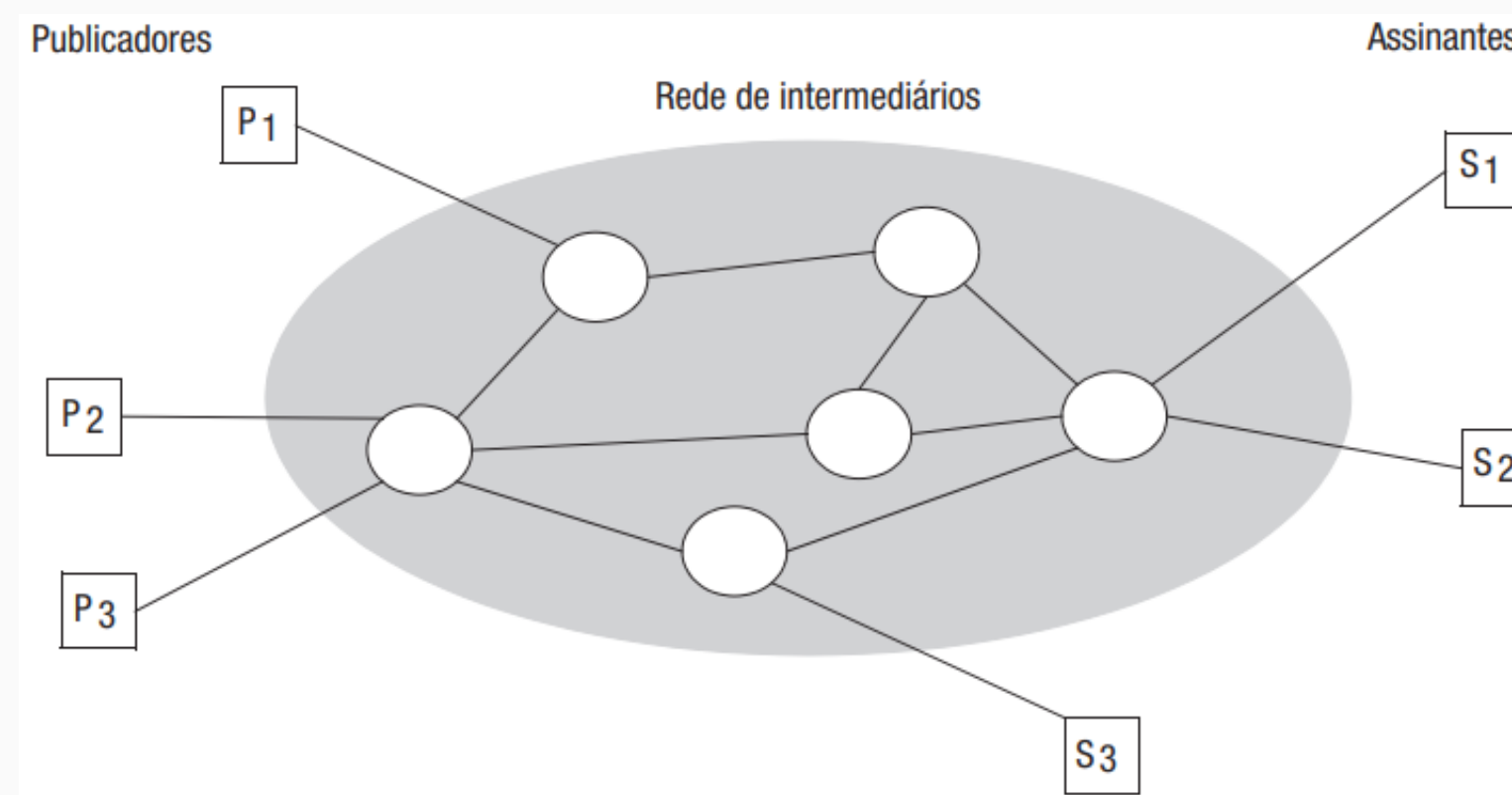
# Implementações

6

- *Centralizada*: centraliza a implementação em um único nó, com um servidor nesse nó atuando como intermediário de evento. Assim, a interação com o intermediário é por meio de uma série de mensagens ponto a ponto.
- *Distribuída*: o intermediário centralizado é substituído por uma *rede de intermediários*, que coopera para oferecer a funcionalidade desejada. Essa estratégia têm o potencial de sobreviver à falha do nó.



Centralizada



Distribuída

- *Inundação*
  - Enviar uma notificação de evento para todos os nós da rede e, então, realizar a correspondência apropriada na extremidade assinante.
- *Filtragem*
  - Os intermediários encaminham as notificações pela rede somente onde há um caminho para um assinante válido.

- Sistemas de informação financeira;
- Outras áreas com divulgação ao vivo de dados em tempo real (incluindo *feeds* RSS);
- Suporte para computação ubíqua, incluindo o gerenciamento de eventos provenientes de infraestrutura ubíqua (por exemplo, eventos de localização);
- Um grande conjunto de aplicativos de monitoramento.





# Questão

9

1 Q48311 Arquitetura de Software > Padrões de projeto (Design Patterns)

Ano: 2009 Banca: CESPE / CEBRASPE Órgão: INMETRO Prova: CESPE - 2009 - INMETRO - Analista Executivo em Metrologia e Qualidade - Desenvolvimento de Sistemas - Parte II

Texto associado ⊖

Acerca de linhas de produtos e de componentes de *software*, julgue os itens subsequentes.

Na arquitetura de sistemas de mensageria, um dos padrões bastante utilizados é o *Publish-Subscribe*, que possui maior similaridade com o padrão Observer que com o padrão *Iterator*.

☐ Certo

☐ Errado

Responder

Q48311 - Questões de Concursos | Qconcursos.com - <https://www.qconcursos.com/questoes-de-concursos/questoes/b71672ae-68>



# Questão

IO

1

Q48311

Arquitetura de Software ▶ Padrões de projeto (Design Patterns)

Ano: 2009   Banca: CESPE / CEBRASPE   Órgão: INMETRO   Prova: CESPE - 2009 - INMETRO - Analista Executivo em Metrologia e Qualidade - Desenvolvimento de Sistemas - Parte II

Texto associado 

Acerca de linhas de produtos e de componentes de *software*, julgue os itens subsequentes.

Na arquitetura de sistemas de mensageria, um dos padrões bastante utilizados é o *Publish-Subscribe*, que possui maior similaridade com o padrão Observer que com o padrão *Iterator*.

☒ Certo

☐ Errado

Responder

Parabéns! Você acertou!



# Referências Bibliográficas

---

II

- George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair. Sistemas Distribuídos: Conceitos e Projeto. 5ª Edição. Porto Alegre: Bookman, 2013.



I2

# Obrigado!

---



UNIVERSIDADE CATÓLICA DE PELOTAS  
ENGENHARIA DE COMPUTAÇÃO  
SISTEMAS DISTRIBUÍDOS

# Sistemas Publicar-Assinar

MATHEUS GONÇALVES STIGGER

---

1º/07/2022

