

# Sistemas Distribuídos

## Comunicação, Falhas e Coordenação

março de 2019



- troca de mensagens é sempre a primitiva básica

- sobre essa primitiva podemos construir outras visões da comunicação



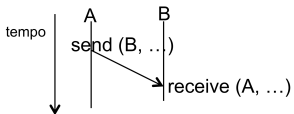
```
send (destino, &mensagem)
```

```
receive (origem, &mensagem)
```

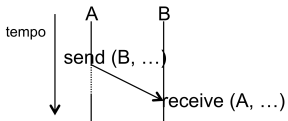
- questões
  - semântica de operações
  - especificação de origem e destino
  - formato da mensagem



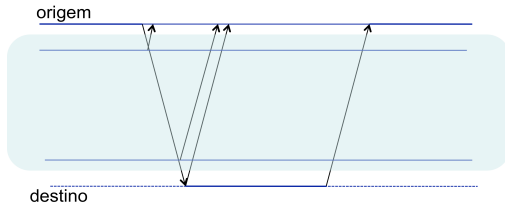
- envio assíncrono: execução procede imediatamente
  - bufferização



- envio síncrono: execução procede quando destinatário recebe mensagem
  - determinismo



# Variantes síncronas e assíncronas



algumas bibliotecas oferecem primitivas com várias diferentes semânticas de garantia de envio



- recebimento síncrono é o convencional
  - execução procede quando há algo a tratar
  - alternativa de recebimento com timeout

## bloqueio!

suspensão da linha de execução corrente até chegada da mensagem



- espera por mensagens de tipos específicos?
  - podem haver outros tipos de mensagens pendentes
- espera por qualquer tipo de mensagem
  - código se torna um enorme emaranhado





- orientação a eventos: chegada de mensagem é encarada como evento a ser tratado
  - recebimento fica implícito



- endereçamento direto (acoplamento)
  - endereços e volatilidade
  - serviços de nomes
- caixas de correios e canais (desacoplamento)
  - endereços bem conhecidos
  - comunicação com par “qualquer”



- no nível mais baixo: sequências de bytes
  - interpretação por conta do programa



- apoio de bibliotecas ou linguagens a empacotamento e desempacotamento
  - marshalling e unmarshalling
- abstrações em vários níveis

