

# Appia

Moldura de composição e execução de  
protocolos

Departamento de Informática, FCUL  
*Nuno Carvalho*

# Resumo

- Modelo
- Componentes e protocolos
- Instalação e configuração
- Suporte
- Exemplo “Ecco”

# Visão geral

- Appia é composto por
  - Um núcleo usado para efectuar composições de protocolos
  - Conjunto de protocolos já existentes
    - Comunicação em grupo, ordenação de mensagens, entre outros

# Modelo de programação

- Programação orientada aos eventos
  - Eventos são recebidos num *handle*
- Significado semântico dos eventos, através de herança
  - Timer, SendableEvent, GroupSendableEvent, etc
- Um único fio de execução

# Componentes (I)

- Layer - Descreve o comportamento de um protocolo através de eventos (required, provided, accepted)
- QoS - Composição de Layers; define uma Qualidade de Serviço
- Session - Instância de um protocolo; mantém estado e trata eventos
- Channel - Composição de Sessions; instanciação de uma QoS

# Componentes (II)

- Temporizadores
  - São pedidos ao núcleo do Appia, através de eventos próprios
  - Tempo da máquina lido por interface própria, a partir do canal
- Threads
  - Pedidas em interface própria

# Eventos

- Usados para comunicação entre camadas e entre processos
- Atributos básicos
  - Canal
  - Sessão que o criou
  - Direcção

# Eventos de tempo

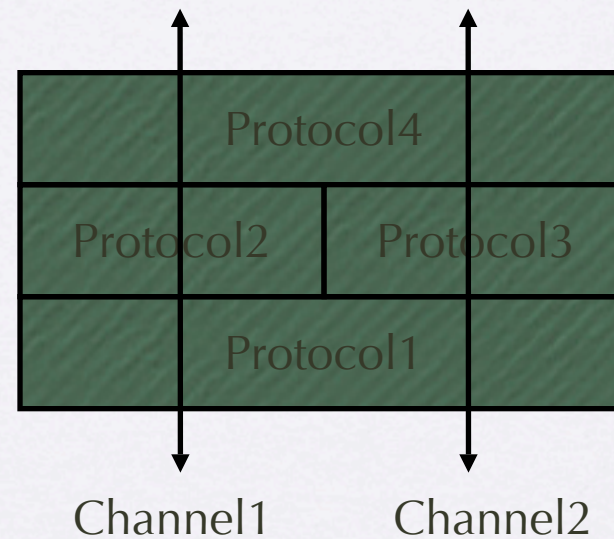
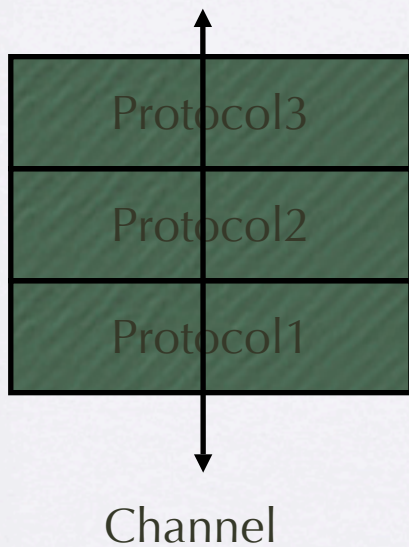
- Timer
- Periodic timer
- Atributo de notificação: EventQualifier
  - ON; OFF; NOTIFY



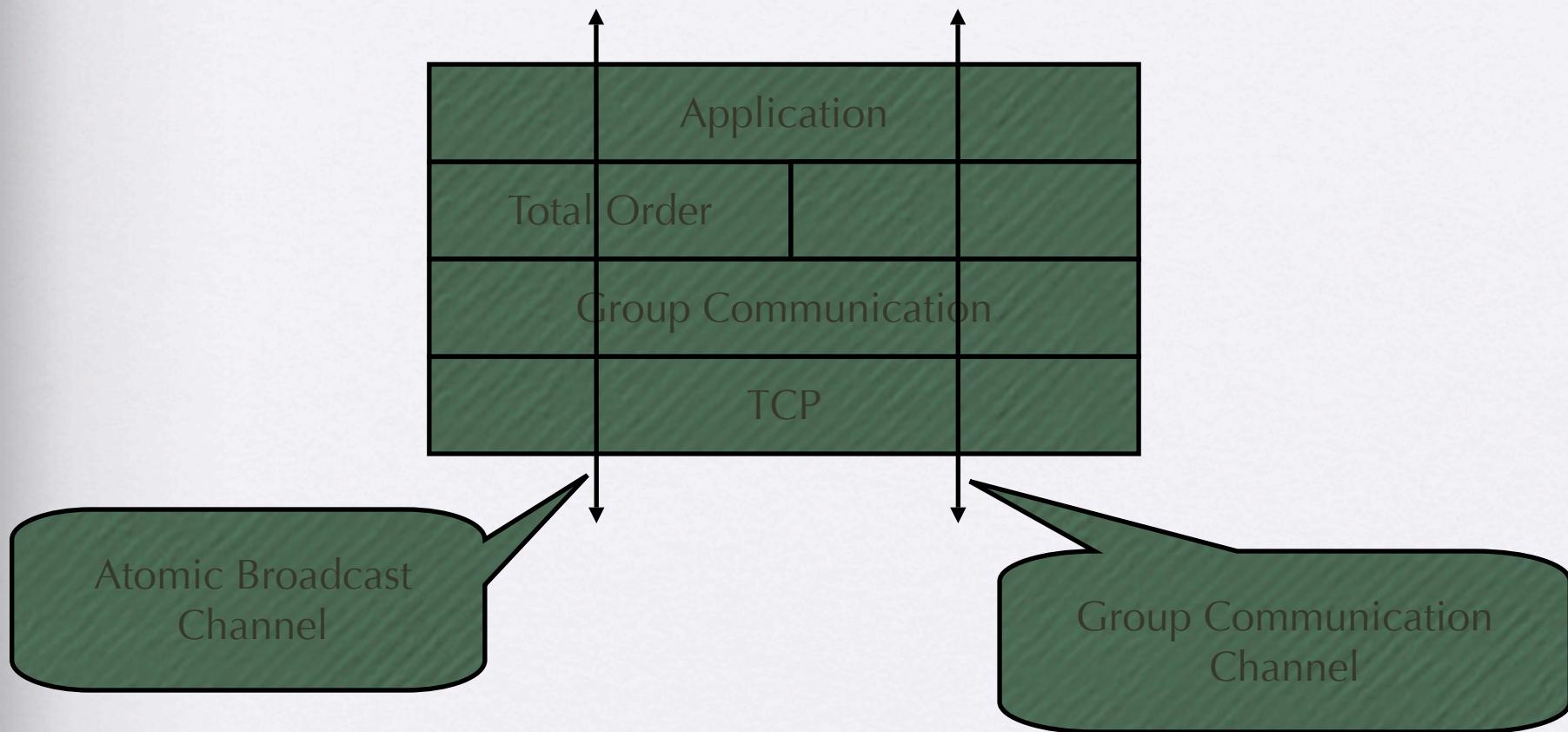
# Eventos a enviar pela rede

- Descendem de SendableEvent
- Só é enviado o conteúdo da classe Message
- Classe Message segue uma semântica de pilha
  - push - insere na mensagem
  - pop - retira da mensagem
  - peek - lê da mensagem sem retirar

# Composição de protocolos (I)



# Composição de protocolos (II)



# Protocolos

- Comunicação com a rede
  - UDP; TCP; TCP+SSL
- Comunicação em grupo
  - Conjunto de protocolos que fornecem sincronia virtual
- Ordenação de mensagens
  - Total (sequenciador e token)
  - Total Optimista
  - Causal
  - Total-Causal

# Instalação

- Usar como uma biblioteca convencional de java (.jar)
- Compilar com Apache Ant
- Colocar ficheiros na CLASSPATH
  - appia.jar
  - log4j.jar
- Importar nas classes em desenvolvimento

# Configuração

- Configuração de canais
  - XML - define templates e canais
- Configuração de logging
  - log4j.properties - Define vários níveis de logging
    - INFO, DEBUG, etc
    - Não usado por todos os protocolos...

# Configuração de canais por XML

- Definição de templates
- Definição de canais a partir de templates
  - Passagem de parâmetros por XML obriga as camadas a implementar `InitializableSession`
- Mais detalhes de seguida...

# Suporte

- Web: <http://appia.di.fc.ul.pt>
- Mailing list e JIRA: <http://appia.continuent.org>
- Documentação
  - Javadoc (na web do Appia e disponível para download)
  - Protocol development manual (pdf)
  - Tutorials



Exemplo  
"Ecco"