

# Design and Evaluation of a Support Service for Mobile, Wireless Publish/Subscribe Applications

Pedro Miguel Martins Nunes

**Tecnologias de Middleware 06/07**  
Curso de Especialização em Informática  
Departamento de Informática  
Faculdade de Ciências da Universidade de Lisboa

15.12.2006

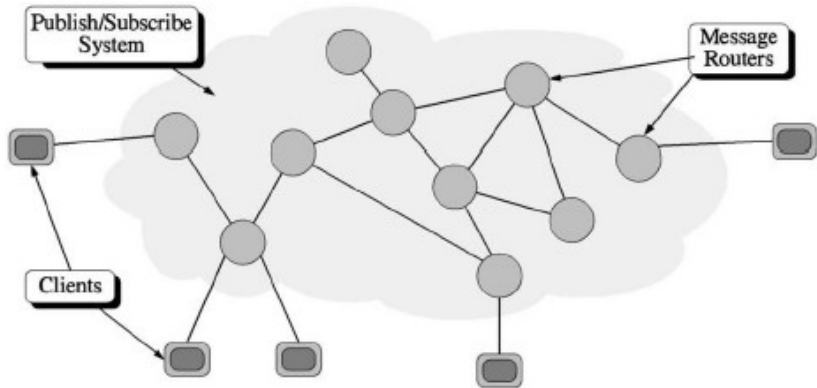
- 1 Introdução
- 2 Serviço de suporte a mobilidade
  - Descrição genérica
  - Funcionamento
  - Opções de sincronização
- 3 Avaliação do desempenho
  - Rede com fios
  - Rede sem fios
- 4 Conclusões
- 5 Referências

# Sistema *Publish/Subscribe*

- **Publish/Subscribe**: serviço de comunicação em que...
  - **Publishers**: publicam mensagens no *middleware*
  - **Subscribers**: registam interesse no *middleware* em determinadas mensagens
  - Serviço é responsável pela entrega de **mensagens publicadas** aos respectivos **subscriptores**
- Formas de implementação:
  - **Servidor centralizado**
  - **Rede de routers de mensagens**

# Sistema *Publish/Subscribe* distribuído

- Clientes estão ligados a vários **pontos de acesso distribuídos**, conectados entre si por *routers* de mensagens



# Serviço de suporte a mobilidade

- **Serviço de suporte a mobilidade** pretende fornecer suporte a aplicações de clientes móveis<sup>1</sup>
- Enquanto a aplicação móvel muda de ponto de acesso, o serviço deve efectuar de forma transparente:
  - gestão de subscrições activas
  - gestão de mensagens
- A inacessibilidade das aplicações móveis ao sistema de *Publish/Subscribe* pode ocorrer devido a:
  - movimento do dispositivo móvel<sup>2</sup>, que pode originar o acesso noutra localização no sistema
  - desconexão do dispositivo móvel

---

<sup>1</sup>Aplicações que se movimentam de um ponto de acesso para outro durante a sua execução

<sup>2</sup>Computador portátil, PDA, telemóvel...

## Objectivos do serviço

- Gestão das subscrições<sup>3</sup>:
  - enquanto o cliente se encontra desconectado
  - durante a fase de mudança de ponto de acesso
- Minimização de:
  - mensagens duplicadas
  - mensagens perdidas
- Portabilidade:
  - diferentes plataformas  $P/S$ <sup>4</sup>
- Adaptabilidade:
  - características da implementação do sistema  $P/S$
  - características da rede de comunicação subjacente

---

<sup>3</sup>O serviço não suporta a gestão de publicações, ficando estas a cargo dos publicadores móveis

<sup>4</sup>*Publish/Subscribe*

## Desenho do serviço

- O desenho do serviço de suporte à mobilidade é baseado num proxy cliente:
  - actua como *interface* ao sistema *P/S*, enquanto o cliente se encontra desconectado
  - desvia subscrições e mensagens de um ponto de acesso para outro, quando o cliente se reconecta à rede
- São feitas poucas *assumpções* sobre o sistema *P/S*:
  - Considerada API com funções *publish* e *subscribe* genéricas
  - É requerido que as subscrições e publicações possam ser serializadas<sup>5</sup> para armazenamento
  - Não são feitas *assumpções* sobre modelo de dados, linguagem de filtragem, algoritmo de encaminhamento, etc.

---

<sup>5</sup>Serialização é o processo de persistir o estado de um objecto, para posterior reutilização

# Componentes do serviço

- **Mobility service proxies**

- componentes independentes e estacionários, que correm nos pontos de acesso do sistema *P/S*
- largamente independente do sistema *P/S*

- **Mobility service client library**

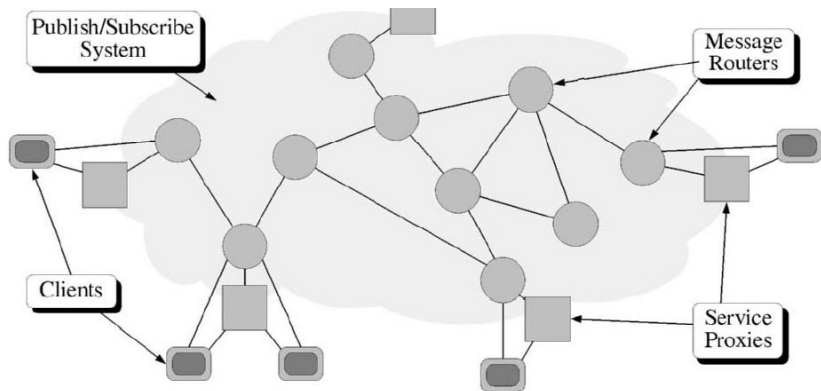
- componente ligado à aplicação cliente
- medeia alguns dos pedidos efectuados ao sistema *P/S*
- interage com *mobility proxies* durante as funções *move-out*<sup>6</sup> e *move-in*<sup>7</sup>
- largamente interligado à API do sistema *P/S*

---

<sup>6</sup>Função invocada pelo cliente para se desconectar da rede

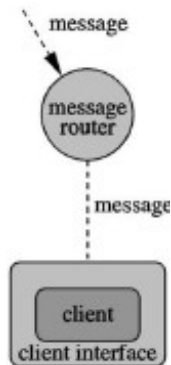
<sup>7</sup>Função invocada pelo cliente para se reconectar à rede

# Mobility service proxies



## Funcionamento básico do serviço de mobilidade

- Durante operações conectadas, o cliente publica e subscreve directamente do sistema *P/S*
- Operação de subscrição mediada pela *client library*, que mantém cópia local das subscrições



## Funcionamento básico do serviço de mobilidade

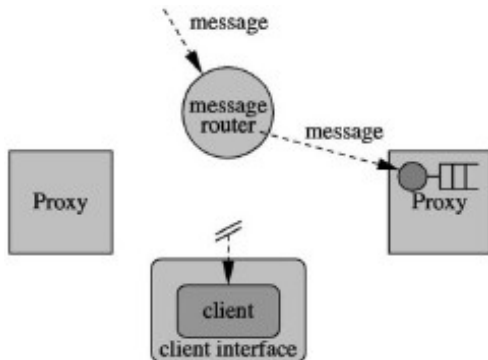
- 1 Antes de se desconectar, cliente invoca função *move-out*
- 2 Função *move-out* indica à *client library* para transferir as subscrições armazenadas para o seu *mobility proxy*
- 3 *Proxy* inicia processo de subscrição, armazenando as mensagens correspondentes num *buffer* dedicado
- 4 Quando o cliente chega ao destino, utiliza a função *move-in* para contactar um *proxy* local<sup>8</sup>, passando-lhe o endereço do *proxy* remoto<sup>9</sup> do qual se encontra desligado
- 5 *Proxies* executam protocolo que resulta na transferência de todas as subscrições e mensagens para o *proxy* local e posteriormente para o *buffer* mantido pela *client library*

---

<sup>8</sup> *Move-in proxy*

<sup>9</sup> *Move-out proxy*

# Funcionamento básico do serviço de mobilidade

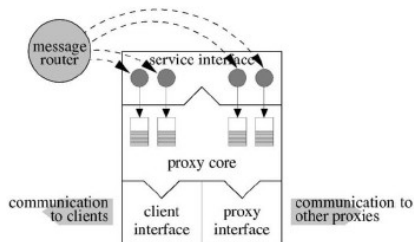


## Arquitectura interna

- Os *mobility service proxies* podem ser considerados como estações de serviço locais para clientes móveis
- Um *proxy* pode servir múltiplos clientes
- Os *proxies* são independentes entre si, não estabelecendo conexões permanentes
  - As únicas conexões estabelecidas são consequência da execução da função *move-in*

# Arquitectura interna

- *Proxy core*
  - armazenamento em *buffer* de mensagens recebidas
  - serviços de sincronização
- *Client Interface*
  - gestão da interacção com o cliente
- *Proxy Interface*
  - gestão da interacção com outro proxy
- *Service Interface*
  - gestão da interacção com o sistema *P/S*



## Função *move-out*

- 1 Na invocação, o interface de mobilidade do cliente passa um **identificador do cliente** e uma **lista de subscrições** ao *proxy* de mobilidade<sup>10</sup>
- 2 O *proxy* de mobilidade executa a função de *move-out*:
  - criando um *client handler* com o identificador fornecido
  - subscrevendo a lista de subscrições enviada pela *client library*, para o *handler* criado
- 3 Após o processo de subscrição terminar, é enviado uma notificação de sucesso para a *client library*
- 4 Após recepção da notificação, a *client library* desconecta-se do sistema *P/S* e retorna da função de *move-out*

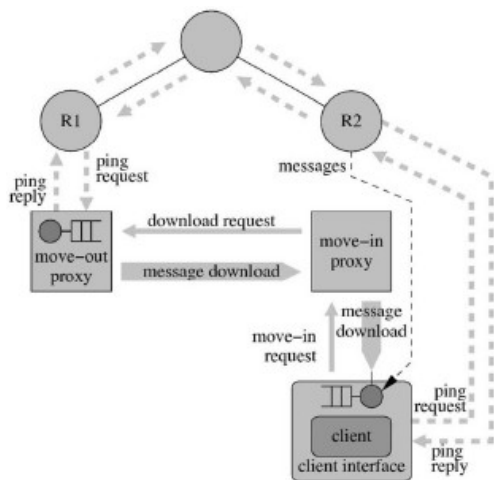
---

<sup>10</sup>Parâmetro opcional de Qualidade de Serviço (QoS) pode ser passado, incluindo opção de sincronização

## Função *move-in*

- 1 A *client library* activa uma fila local para recepção de mensagens (condizentes com as subscrições do cliente) e inicia um processo para gestão da mesma (*receiver process*)
- 2 A *client library* envia um *message download request* ao *move-in proxy*, especificando o endereço do *move-out proxy* e o identificador do cliente.
- 3 O *move-in proxy* contacta o *move-out proxy* e realiza o *download* de todas as mensagens existentes relativas ao identificador fornecido; depois o *client handler* é eliminado
- 4 O *move-in proxy* envia todas as mensagens recebidas para a *client library* que as armazena na fila de mensagens
- 5 Após recepção de todas as mensagens do *move-out proxy*, a *client library* elimina o *receiver process*, após redireccionamento das subscrições para o cliente
- 6 Após recuperação das mensagens presentes na fila, estas são passadas para o cliente e a fila é eliminada

# Função *move-in*



## Opções de sincronização

- **Duplicação de mensagens** pode ocorrer quando:
  - durante a função *move-out*, entre o fim do processo de subscrição pelo *proxy* e a desconexão do cliente  
a mesma mensagem pode ser entregue ao **cliente** e ao **proxy**
  - durante a função *move-in*, entre a activação do *receiver process* e a desactivação do *client handler*  
a mesma mensagem pode ser entregue ao **proxy** e ao **receiver process**
- **Perda de mensagens** pode ocorrer quando:
  - durante a função *move-out*, depois do cliente se ter desconectado, enquanto as subscrições efectuadas pelo *proxy* ainda não estão activas
  - durante a função *move-in*, depois do *move-out proxy* ter destruído o *client handler*, enquanto as subscrições no *receiver process* ainda não estão activas

## Perda de mensagens - mecanismos preventivos

- Assegurar que, quando movendo mensagens de A para B, as subscrições em B estão activas, antes de remover subscrições de A
- 1º mecanismo: **ping messages**
  - 1 durante a função *move-in*, o *move-out proxy* A subscreve uma *ping request*
  - 2 a *client library* B, subscreve uma *ping response*
  - 3 a *client library* B publica uma série de *ping request*, a intervalos regulares
  - 4 após recepção de uma *ping request* de B, A publica um conjunto de *ping response*
  - 5 após recepção de uma *ping response* de A, a *client library* B solicita o *download request* ao *move-out proxy* A
  - 6 A pode remover as suas subscrições
- 2º mecanismo: **atraso configurável**: esperar um tempo pré-definido antes de remover subscrições de A

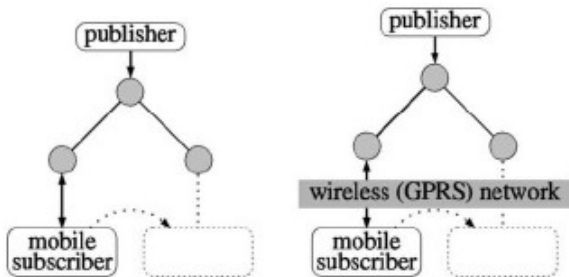
## Objectivos da avaliação de desempenho

- Verificar os benefícios resultantes do uso do serviço de suporte a mobilidade
  - desempenho da aplicação cliente, sob vários níveis de carga, usando ou não o serviço
- Verificar a portabilidade da arquitectura e implementação do serviço
  - avaliar a facilidade em transpor a implementação para outro sistema *P/S*
- Verificar a adaptabilidade do serviço, e particularmente, dos mecanismos de sincronização
  - análise comparativa do desempenho dos mecanismos descritos para os sistemas *P/S* considerados
- Confirmar a validade de desenho do serviço, sobre arquitecturas de rede diversas
  - avaliação de desempenho em redes heterógeneas: LAN com fios vs GPRS

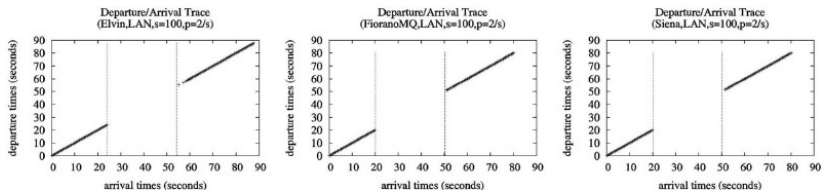
## Setup experimental

- Sistemas  $P/S$ : **Elvin, FioranoMQ, Siena**
- Rede constituída por **três nós**, um **cliente publicador estacionário** e um **cliente subscritor móvel**; sistema  $P/S$  e publicador residem numa rede com fios, de baixa latência e elevada largura de banda
- Infra-estrutura de rede A
  - Subscritor conecta-se ao sistema  $P/S$  por **ligação com fios**
- Infra-estrutura de rede B
  - Subscritor conecta-se ao sistema  $P/S$  por uma **rede GPRS simulada**
- Subscrição de  **$s=1,2,5,10,20,50,100$**  subscrições
- Publicação a uma razão constante de  **$p=1,2,5,10$**  por segundo

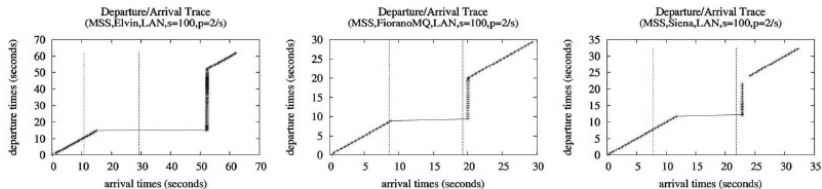
# Setup experimental



# Rede com fios com/sem serviço de suporte à mobilidade

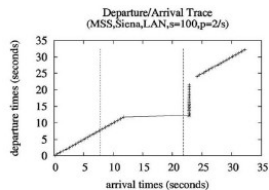
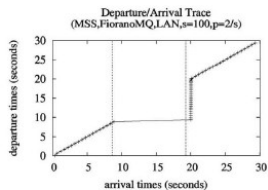
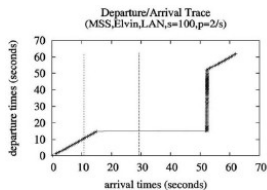


(a)

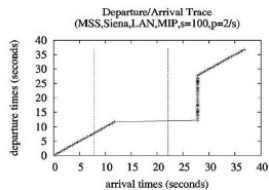
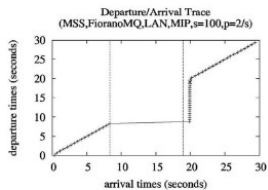
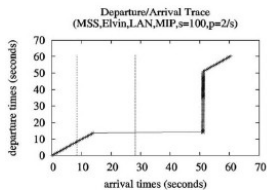


(b)

# Rede com fios com/sem opção de sincronização *ping*

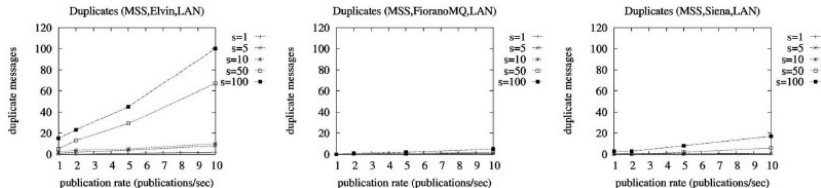


(b)

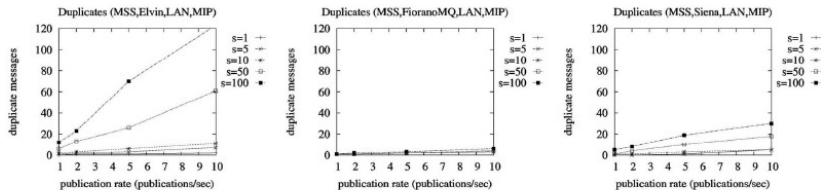


(c)

# Mensagens duplicadas com/sem opção ping

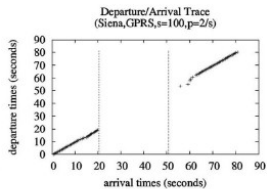
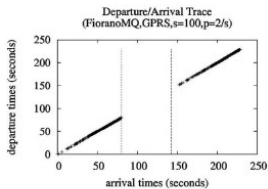
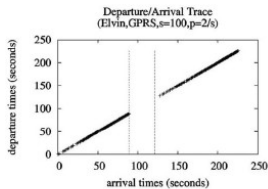


(a)

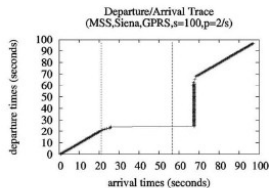
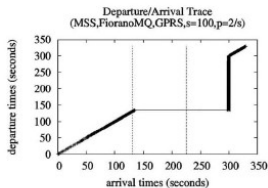
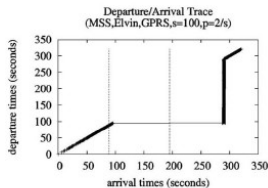


(b)

# Rede sem fios com/sem serviço de suporte à mobilidade

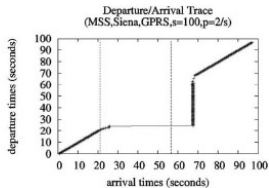
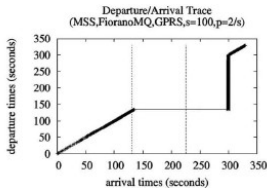
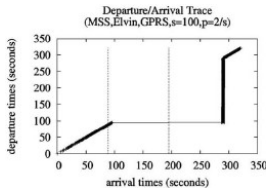
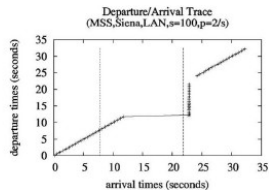
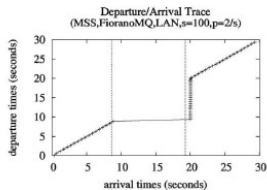
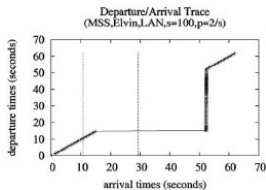


(a)



(b)

# Serviço de suporte à mobilidade: com fios vs sem fios



## Rede sem fios com/sem opção de sincronização *ping*

Summary of Delay, Losses, and Duplicates upon Reattach over GPRS for  $s = 10, 50, 100$  and  $p = 2/sec$

	$s$	delay		losses		duplicates	
		basic	ping	basic	ping	basic	ping
Elvin	10	13551	17151	0	0	18	22
	50	48608	52525	0	0	76	88
	100	95031	98798	0	0	148	159
FioranoMQ	10	13932	15747	0	0	11	18
	50	40544	42215	0	0	43	50
	100	74773	77304	0	0	90	90
Siena	10	8294	11751	0	0	7	12
	50	10984	23585	2	0	4	17
	100	10841	39608	10	0	4	30

## Conclusões

- O objectivo do serviço proposto é providenciar suporte a clientes móveis de um sistema de *Publish/Subscribe*:
  - permitindo o armazenamento de publicações
  - gerindo a mudança de subscrições de um ponto de acesso para outro
- O serviço fornece ainda duas opções de sincronização, de forma a reduzir a perda de publicações durante o processo de mudança de ponto de acesso
- A avaliação de desempenho realizada evidenciou:
  - a portabilidade do serviço, demonstrada pela implementação em vários sistemas *P/S*
  - a validade do serviço proposto para as implementações consideradas, sob diferentes configurações de rede e níveis de carga
  - a eficácia dos mecanismos de sincronização propostos

- **Design and Evaluation of a Support Service for Mobile, Wireless Publish/Subscribe Applications**

M.Caporuscio, A.Carzaniga, A.Wolf

*IEEE Transactions on Software Engineering, 29(12): pp 1059-1071, Dec. 2003*

- **The Many Faces of Publish-Subscribe**

P.Eugster, P.Felber, R.Guerraoui, A.Kermarrec

*ACM Computing Surveys, Vol. 35, No. 2, pp 114-131, June 2003*