

Uma proposta de arquitetura de serviço de chat em Rede Social Educacional

Trabalho de Graduação

Guilherme Oliveira Cavalcanti

AGENDA

- Contexto
- Objetivos
- Arquitetura Atual
- Problemas
- Decisões de projeto
- Arquitetura proposta
- Resultados

CONTEXTO

2009 @ CIn

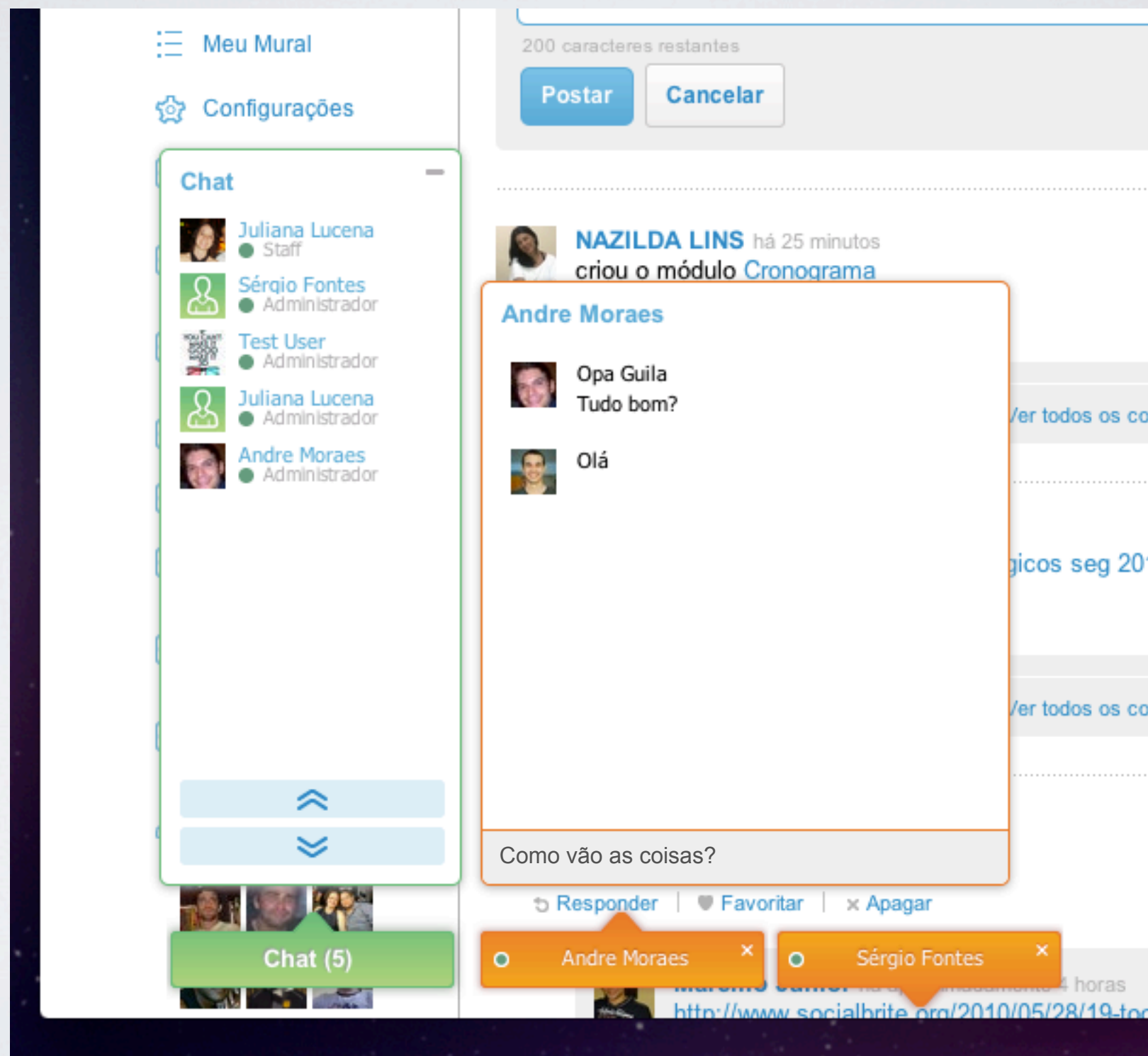


- Plataforma de educação a distância construída numa estrutura de rede social

2010 @ Porto Digital



CHAT

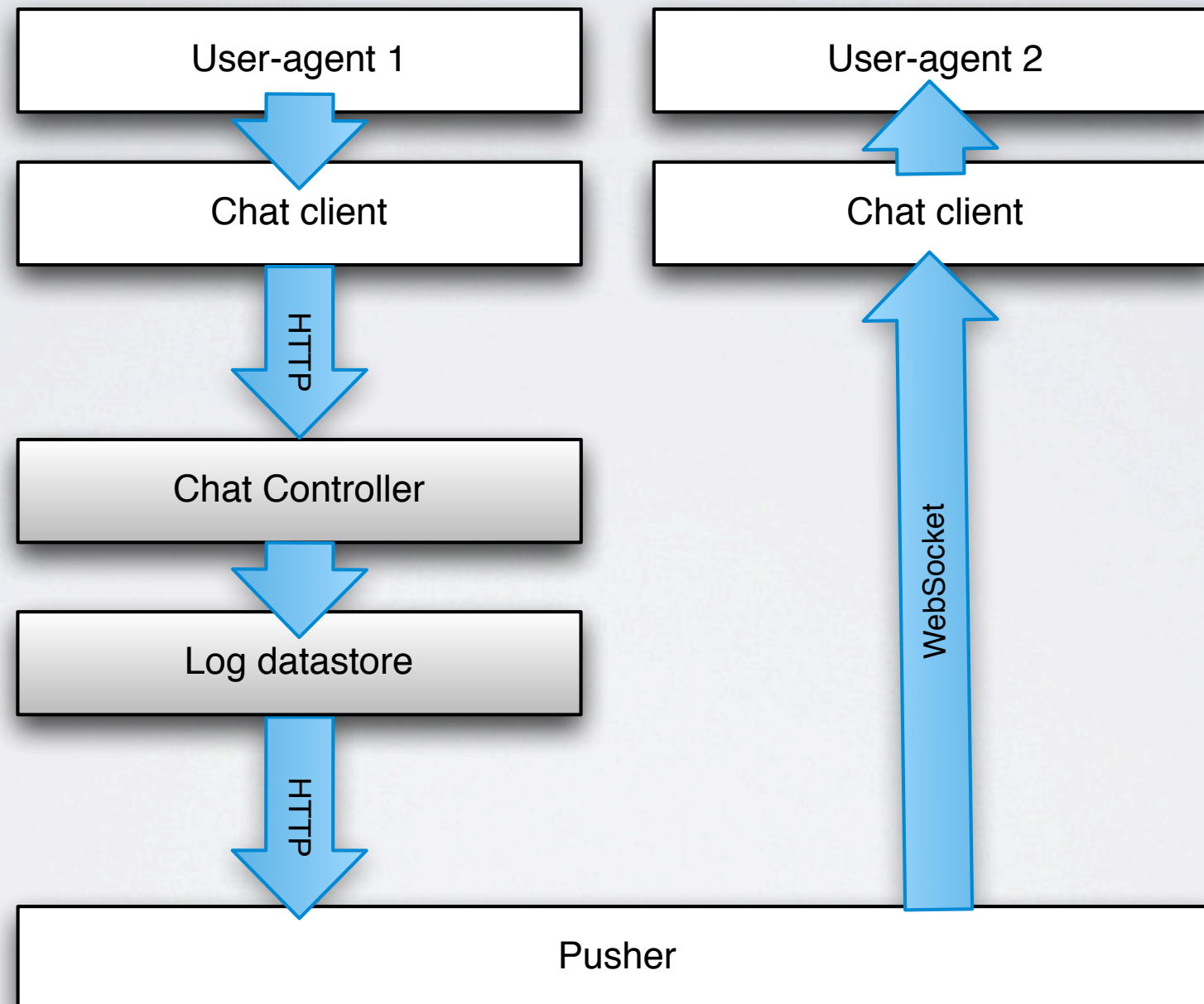


OBJETIVOS

- Detectar gargalos de performance e propor arquitetura alternativa
- Reduzir custos
- Permitir cenário de uso off-line

ARQUITETURA ATUAL

TEMPO DE EXECUÇÃO



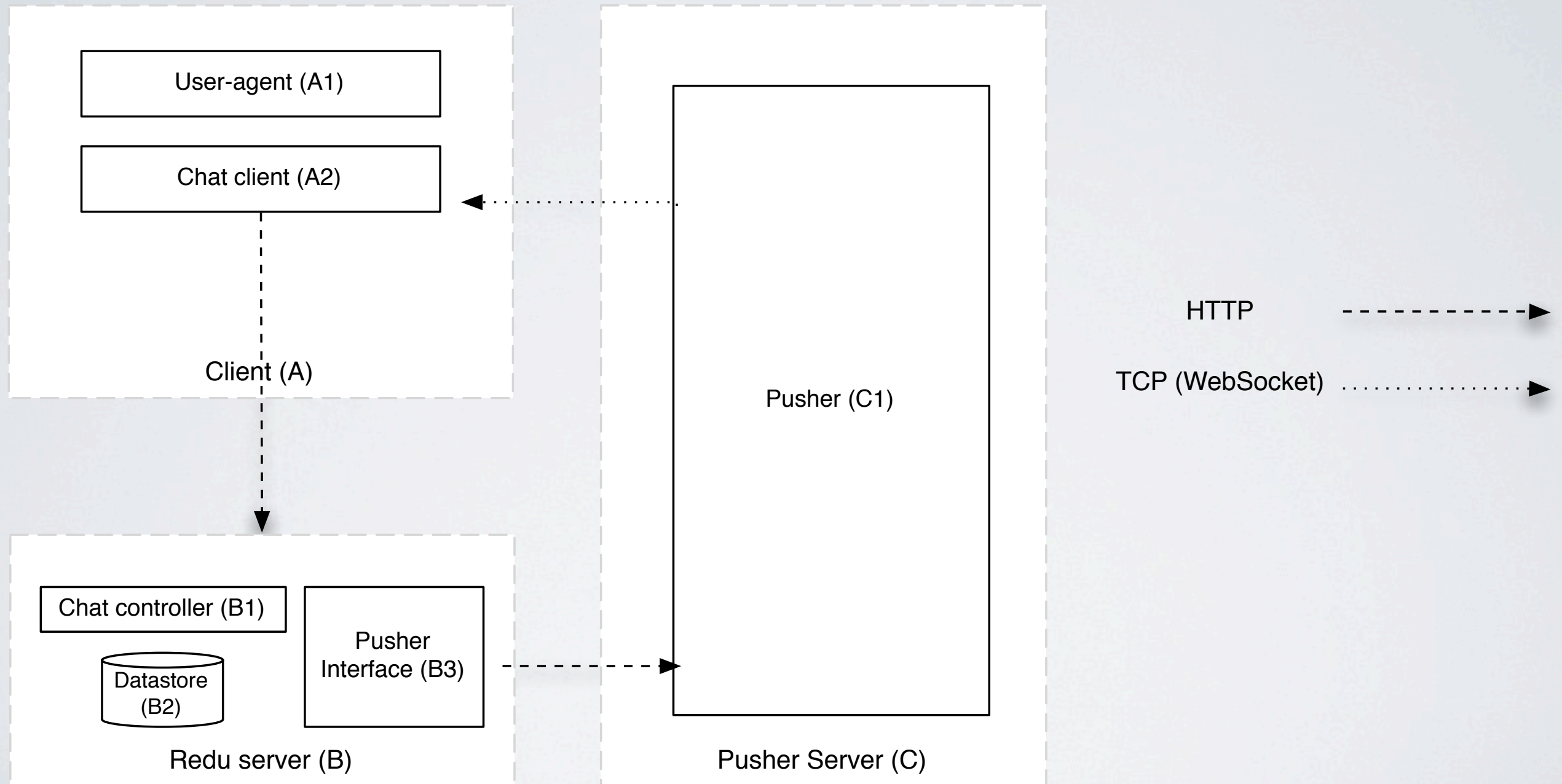
WEBSOCKET?

defines a **full-duplex** single socket connection over which **messages** can be sent between client and server

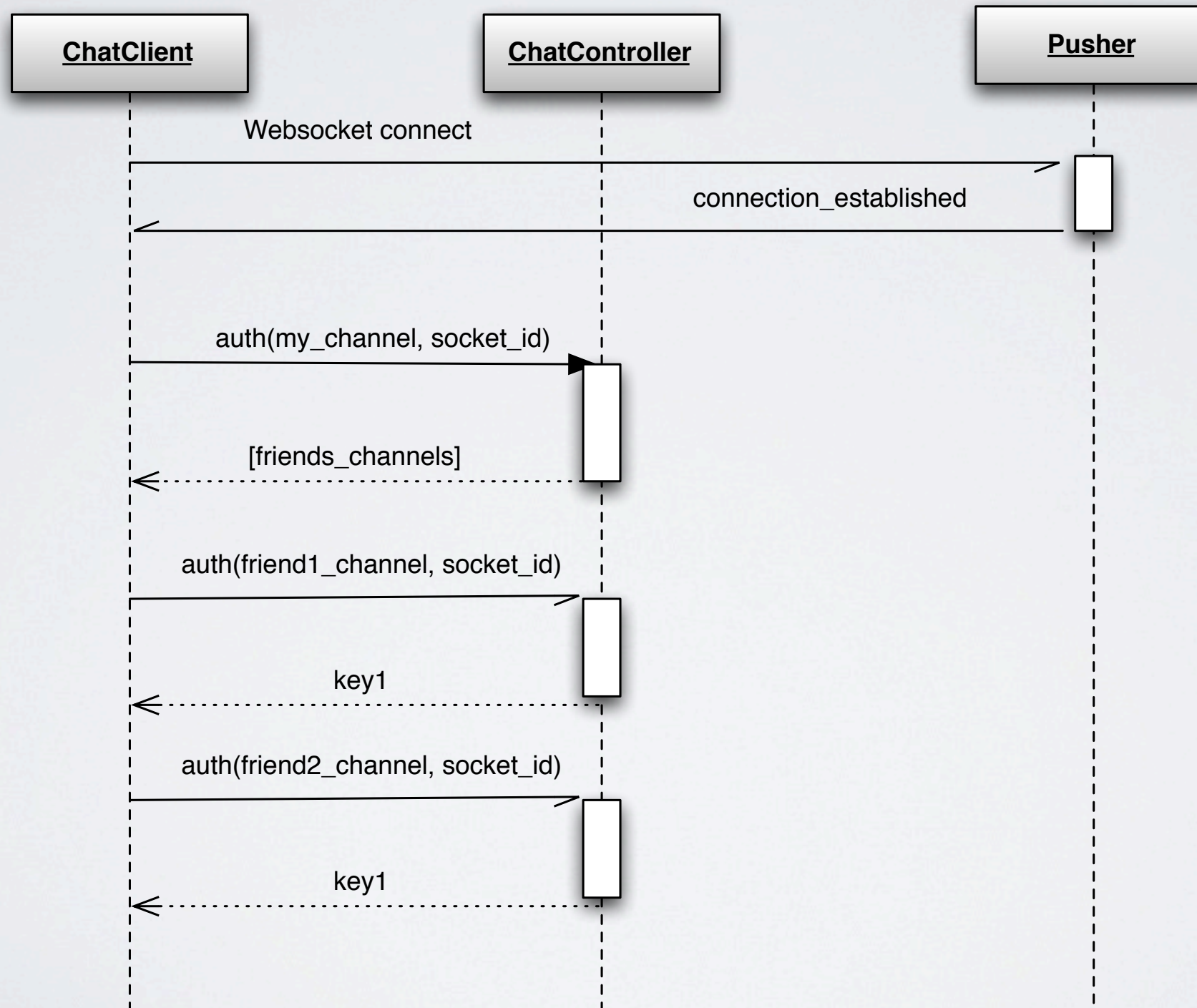
The WebSocket API, W3C

Tell, don't ask.

COMPONENTES



AUTENTICAÇÃO



PROBLEMAS

PERFORMANCE

- Muitas requisições HTTP para autenticação no Chat
- Limitação client-side

PERFORMANCE

Controlador#action	Hits	% do total
ChatController#auth	156145	63.7%
EnvironmentsController#preview	32221	13.2%
UsersController#home	5834	2.4%

Indicativo de quantidade conexões relativas ao chat

PERFORMANCE

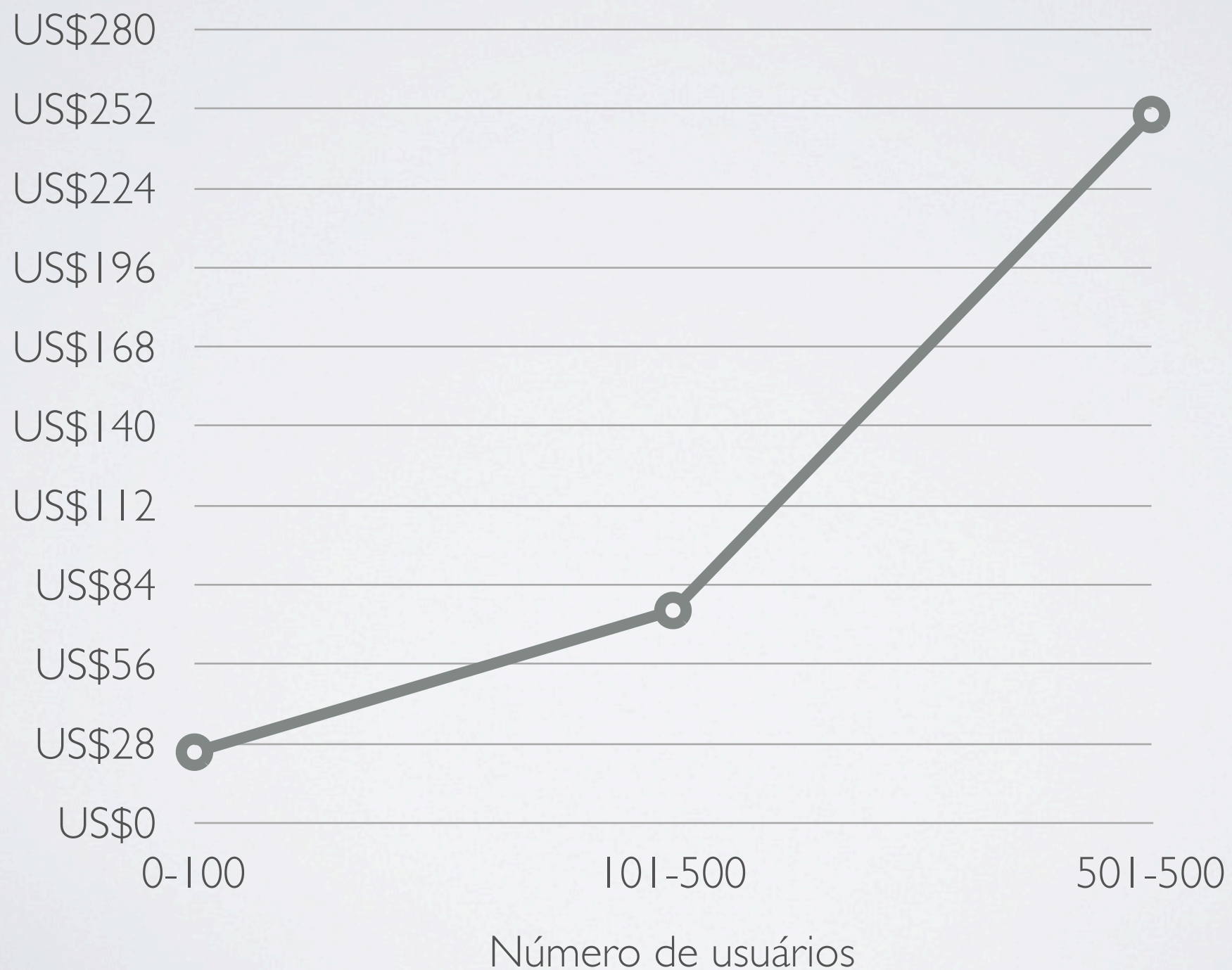
- Máximo de 6 conexões abertas simultaneamente
- Conexões seguintes entram numa fila

CUSTO

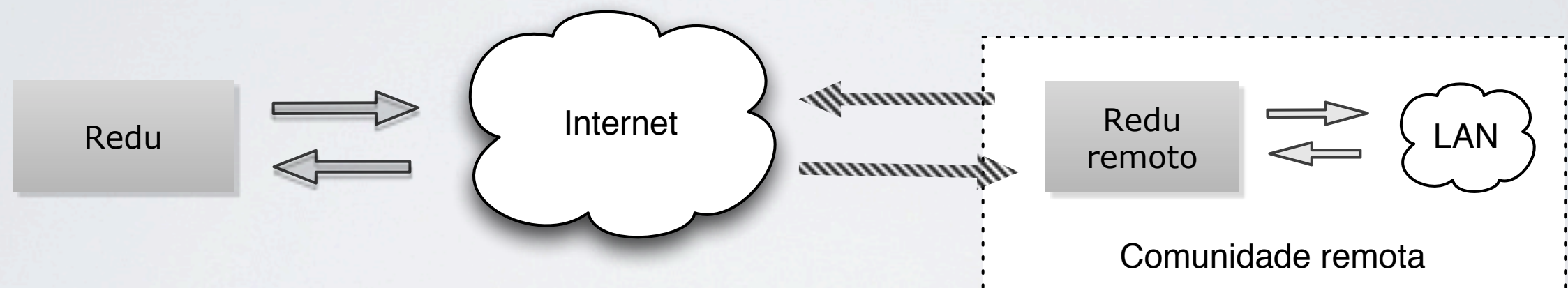
- Uso de webservice de terceiros (Pusher)
- Cobrança por conexões simultaneas e mensagens

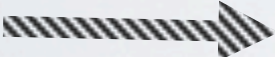
CUSTO


Custo por usuários simultâneos



REDU OFF-LINE



 Interação periódica

 Interação permanente

METODOLOGIA

A simple document describing key architecture decisions can go a long way in demystifying past and future system architectures.

TYREE, J.; AKERMAN, A. Architecture decisions: Demystifying architecture.

CRITÉRIOS

Pode ser desenvolvida em até 60 dias?

Reduz de maneira efetiva os custos?

Tecnologias conhecidas pela equipe?

Usa tecnologias/serviços testados em produção?

Melhora a experiência do usuário no uso do Chat?

Facilita a manutenção?

Facilita a criação de estratégia de escalabilidade?

Permite uso off-line?

Critérios levados em conta na escolha da melhor abordagem

DECISÕES DE ARQUITETURA

PROBLEMA: DESEMPENHO

Como reduzir N° de requisições de autenticação?	
Decisão	Realizar autenticação em um número pequeno e constante de requisições
Limitações	Não reduz custos; Não permite uso off-line;
Argumento	Melhor custo benefício do ponto de vista da experiência do usuário

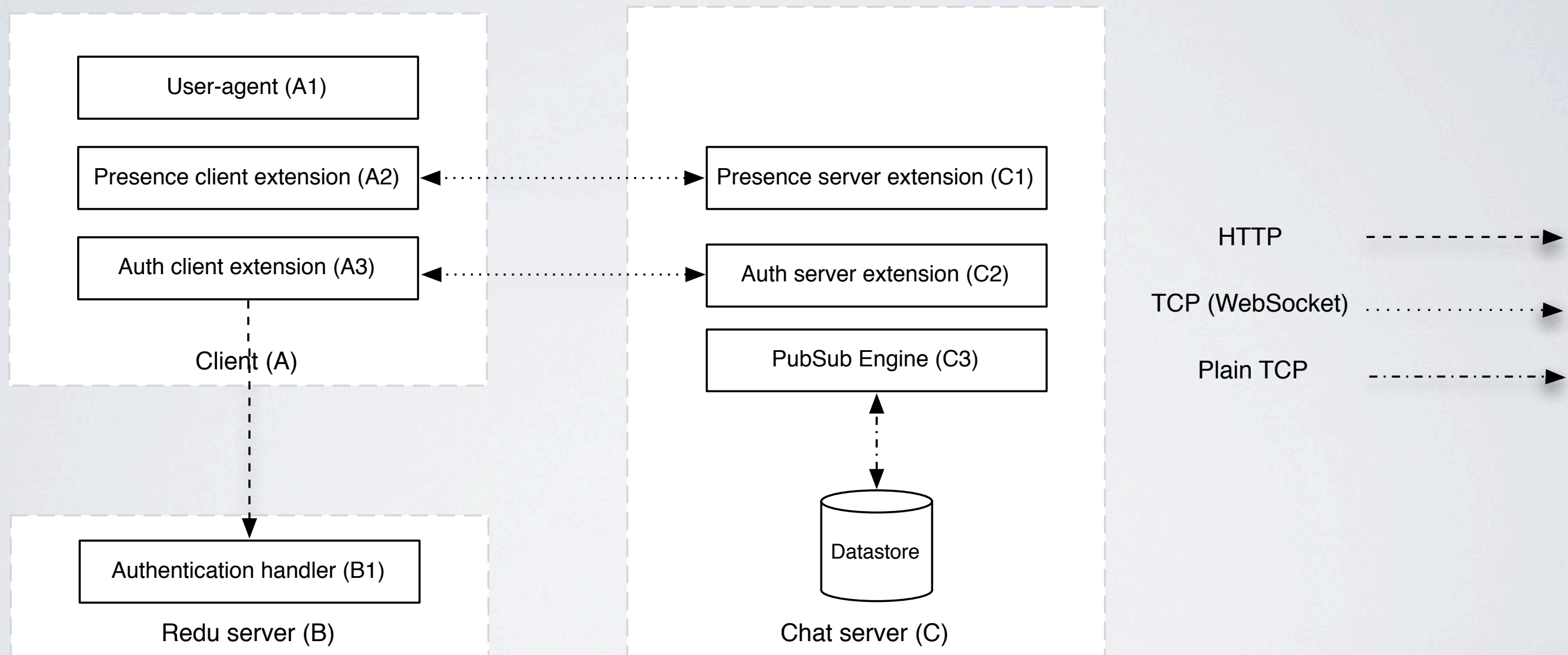
PROBLEMA: CUSTO

Como reduzir custos por usuário gerado pelo Chat?

Decisão	Desenvolver solução própria para notificações servidor-cliente
Limitações	N/A
Argumento	Custo de webservice próprio inferior ao Pusher; Permite uso off-line

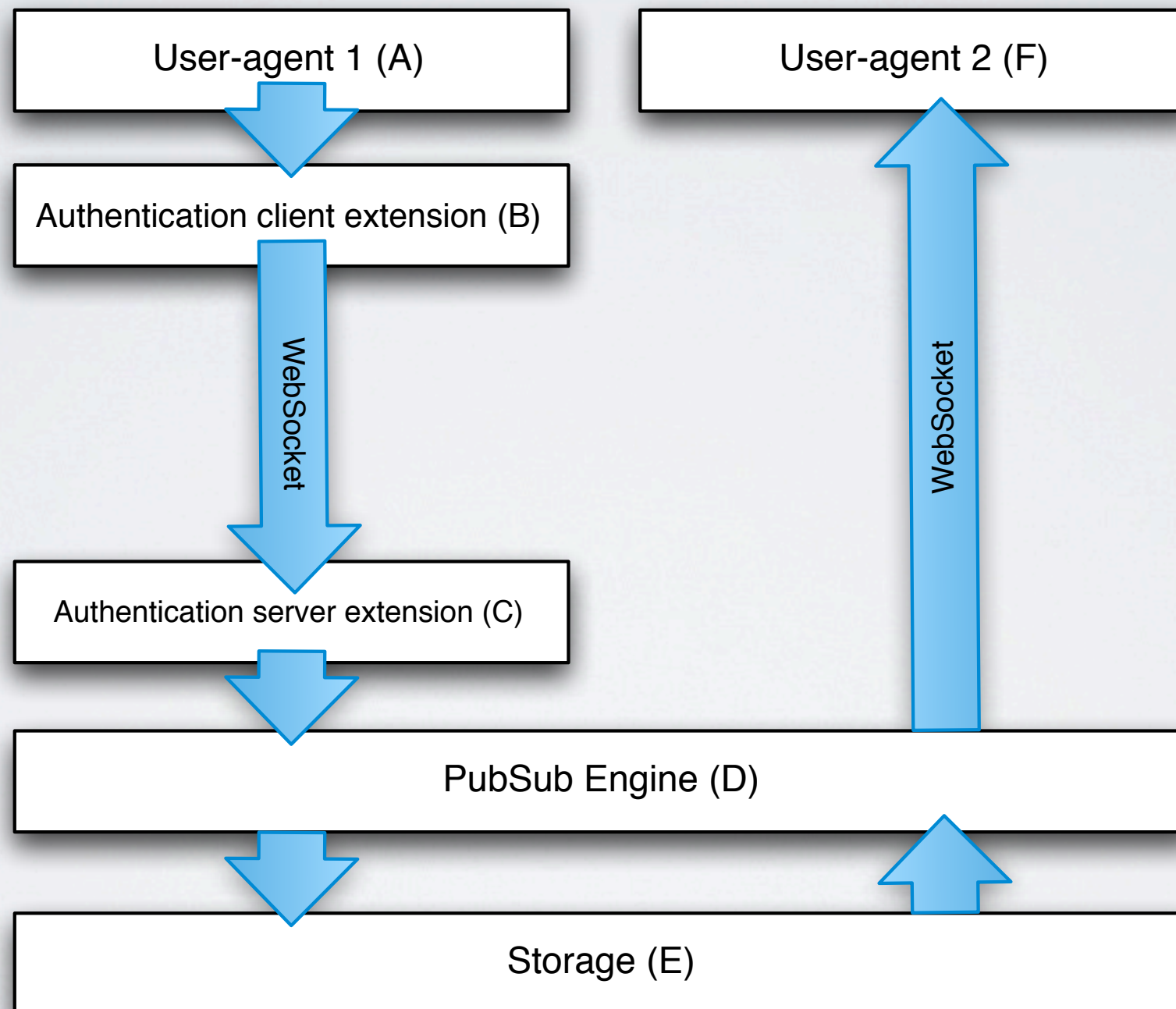
ARQUITETURA PROPOSTA

COMPONENTES



Componentes da arquitetura proposta

TEMPO DE EXECUCÃO



Componentes e protocolos em tempo de execução

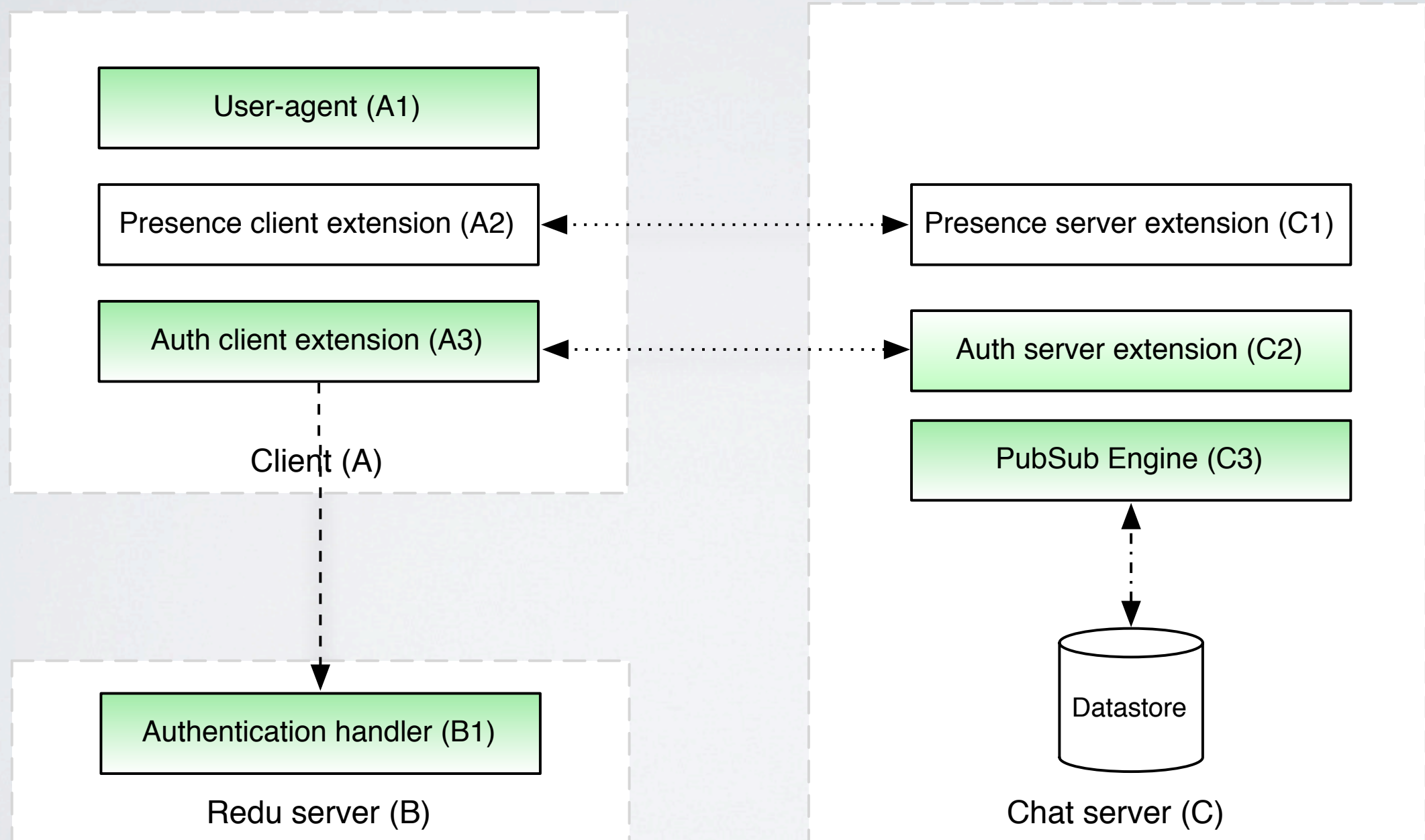
RESULTADOS

DESENVOLVIMENTO

Abordagem	Resultado
Como reduzir N° de requisições de autenticação?	Implementado
Como reduzir custos por usuário gerado pelo Chat?	Implementado parcialmente

INDICADORES

Controlador	Hits	% do total
ChatController#multiauth	16989	19.2%
UsersController#index	9452	10.7%
BaseController#site_index	6806	7.7%



TRABALHOS FUTUROS

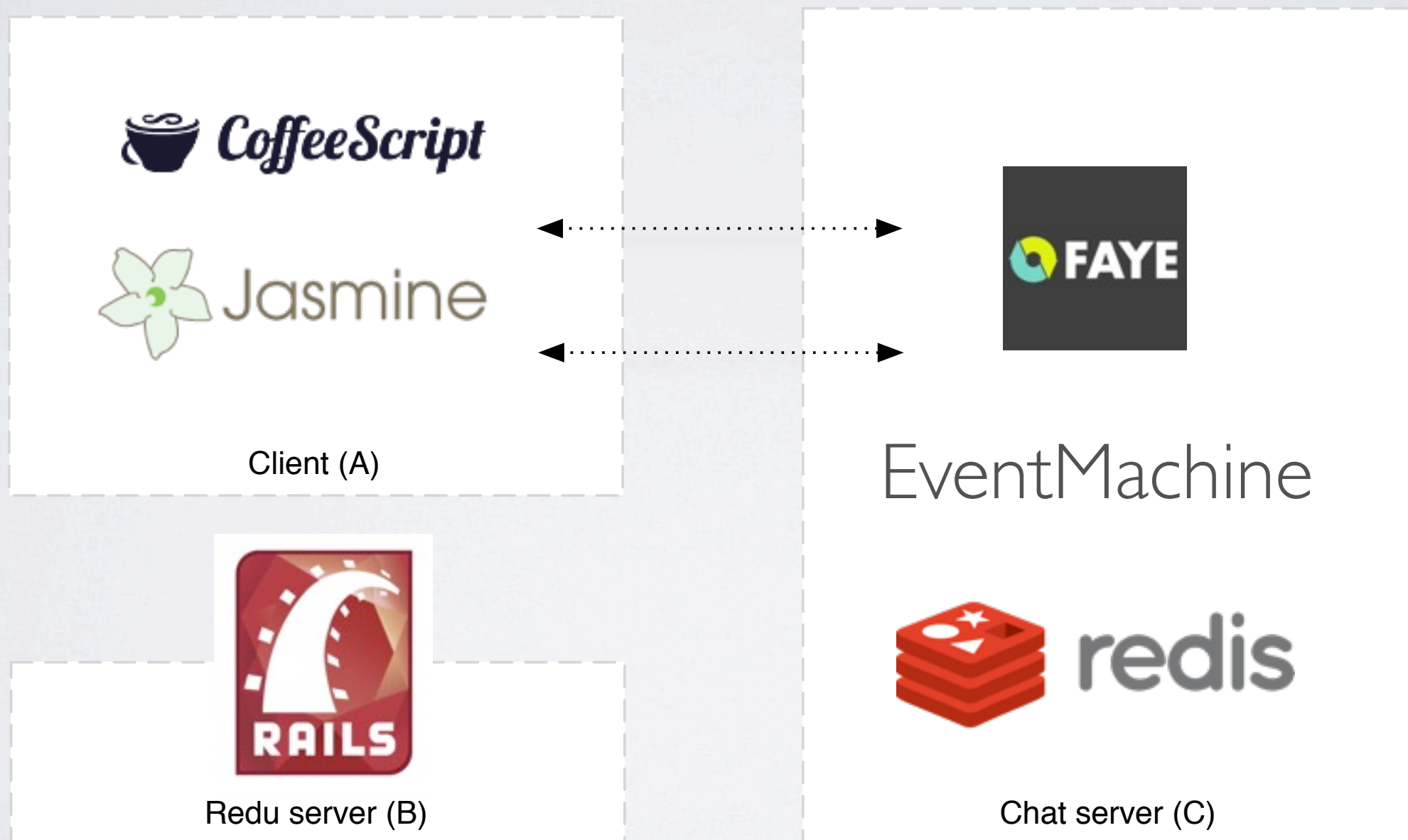
- Informações de presença
- Generalização para outras aplicações real-time
- Chat embarcado

AGRADECIMENTOS



SE DER TEMPO...

TECNOLOGIAS



EVENTMACHINE

- Escrito em Ruby e C++
- I/O não bloqueante
- Baseado no *reactor pattern*
- Usa o epoll, kqueue ou select()
- Orientado a eventos

FUNDAMENTAÇÃO

- Distância transacional
 - Estrutura
 - Diálogo
- Autonomia
- Evasão

