

ESPECIFICAÇÃO DE TRABALHO PRÁTICO

Descrição Geral

Os alunos deverão desenvolver um editor de textos colaborativo, onde usuários conectados em rede poderão visualizar e editar um documento comum a todos os colaboradores. Um determinado usuário poderá criar um novo documento ou juntar-se à edição de um documento já criado por um outro usuário.

Objetivos

Proporcionar um contato prático do aluno com o desenvolvimento de sistemas distribuídos. Verificar a conveniência de uso do Serviço de Eventos de CORBA para esta aplicação em particular. Verificar o desempenho apresentado pelo Serviço de Eventos do OpenORB quando submetido a uma carga considerável (muitos colaboradores). Analisar esta aplicação sob os pontos de vista teóricos vistos em sala.

O Editor Colaborativo

Os alunos deverão desenvolver um ambiente colaborativo para edição de documentos. A figura 1 apresenta a interface sugerida para uso do ambiente. Cada usuário poderá criar um novo documento ou abrir um documento já existente.

Um documento está sempre dividido em seções (por exemplo: Introdução, Trabalhos Correlatos, A Arquitetura Proposta etc). Cada seção será editada por um e somente um usuário. As outras seções do documento serão apenas visualizadas (atualizadas periodicamente com as atualizações dos outros usuários do documento). Cada seção, daqui pra frente denominada *slot*, representará a unidade de bloqueio para edição por um usuário.

Ao iniciar a aplicação será solicitado um *nickname* que identificará o usuário unicamente na rede de autores colaborativos (figura 2). Ao criar um documento, o usuário que realizou a operação se torna o proprietário deste documento e deve informar o nome do documento e o número de *slots* que o documento possui, conforme ilustrado na figura 3. O usuário proprietário passará então a editar sempre o *slot* de número 0 (primeiro *slot*) e os outros *slots* deverão ser criados e estarão aguardando conexões de novos colaboradores.

Ao abrir um documento já existente, o usuário será direcionado para o primeiro *slot* livre do documento, informará o título do *slot* (Introdução, Trabalhos Correlatos etc) conforme ilustrado na figura 4 e passará a editá-lo. Caso não existam *slots* livres o usuário receberá a mensagem “Não existem *slots* livres. Abrindo documento somente para leitura!”. Uma vez editando o *slot*, este deverá ser periodicamente e automaticamente atualizado em todos os outros usuários colaboradores. Os *slots* dos outros autores serão também periodicamente e automaticamente atualizados neste cliente.

Ao clicar no botão Sair, o usuário abandonará o texto e deixará o *slot* livre para outro usuário colaborador. O texto do usuário será perdido no momento da saída. O sistema não prevê, neste momento, mecanismos para armazenamento persistente do documento. Supõe-se que o usuário proprietário do documento nunca irá deixar o sistema antes de todos os outros colaboradores.

Em cada *slot* deverá ser apresentado o título do *slot* (dado pelo usuário que o edita), o nome do usuário que edita o *slot* (ou 'Vazio' caso esteja vazio) e o conteúdo do *slot* (caso ocupado). Supõe-se que não existirão dois usuários com o mesmo *nickname* na rede de colaboradores.



Figura 1: interface principal do NetPad/FRB.



Figura 2: interface para login no sistema.



Figura 3: interface para criação de um documento.



Figura 4: interface para abertura de um documento.

Arquitetura Interna

A aplicação deverá fazer uso do Serviço de Eventos do CORBA como suporte básico para a comunicação muitos-para-muitos, assíncrona e desacoplada do ambiente colaborativo. Cada autor colaborativo será implementado como um objeto que, ao mesmo tempo, é um produtor e consumidor de eventos. O modelo de eventos a ser seguido é o modelo Push Canônico.

Cada cliente do ambiente colaborativo será formado por uma *thread* principal, responsável pelo gerenciamento da interface gráfica, por uma *thread* que representará o consumidor de eventos e por outra *thread* que representará o produtor de eventos, conforme ilustrado no figura 5.

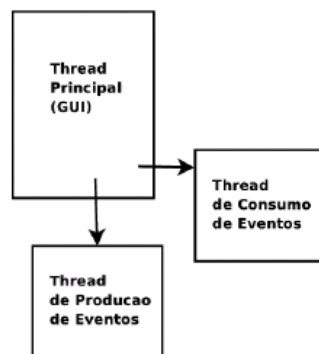


Figura 5: *threads* utilizadas pela aplicação.

A *thread* que representa o consumidor de eventos receberá mensagens de controle e mensagens de atualização de *slots*, seguindo o protocolo descrito na próxima seção. Ao ser recebida uma mensagem de atualização de *slot*, o *slot* informado na mensagem deverá ser atualizado com o conteúdo presente na mensagem, representando uma edição de um autor remoto.

A *thread* que representa o produtor fará, periodicamente, o envio do conteúdo do *slot* sendo editado pelo usuário. Este conteúdo será repassado, pelo canal de eventos, para todos os outros editores colaboradores do documento.

A figura 6 apresenta o relacionamento da aplicação com os serviços de nomes e de eventos, bem como com a fábrica de canais de eventos, disponibilizada pelo OpenORB.

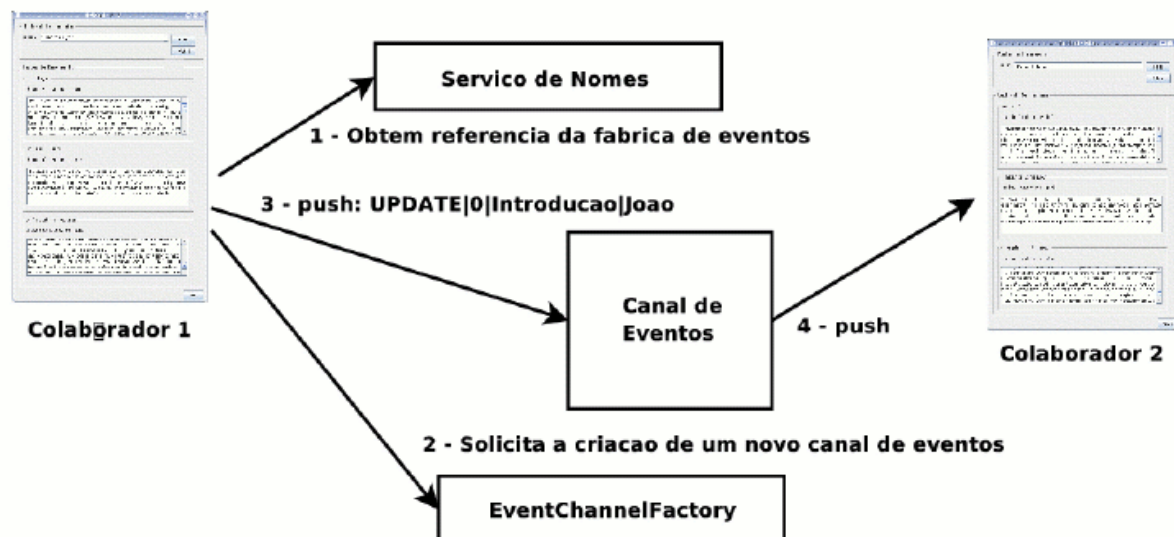


Figura 6: relacionamento da aplicação com os serviços do ambiente distribuído.

Como o serviço de eventos COS (padronizado pela OMG) não realiza filtragem de mensagens, para cada documento colaborativo criado deverá existir um canal de eventos que o representa e que esteja devidamente registrado no servidor de nomes. Este registro deverá sempre ser realizado no contexto raiz do servidor de nomes do OpenORB. Para simplificação do trabalho, não serão solicitadas rotinas de consultas ao servidor de nomes com o objetivo de apresentar quais documentos colaborativos estão abertos no momento. Entretanto, os alunos interessados nesta funcionalidade serão orientados pelo professor. Com base no exposto, supõe-se que cada usuário conhece os nomes dos documentos colaborativos sendo editados no momento e este nome será digitado na interface de abertura de documento (apresentada na figura 4).

Protocolo de Comunicação entre Autores

Nesta seção serão descritos os formatos das mensagens trocadas entre autores com o objetivo de abrir documentos, notificar a posse de um slot e atualizar documentos e notificar a saída de um *slot*.

As possíveis mensagens a serem trocadas são:

- **QUERY|<nickname>**: mensagem enviada quando um autor abre um documento colaborativo. Esta mensagem chegará a todos os autores do documento, porém será respondida apenas pelo autor proprietário do documento. A resposta a esta mensagem QUERY é a mensagem ANSWER, descrita a seguir.
- **ANSWER|<nickname>|<nro_total_slots>|<slot>**: mensagem enviada pelo proprietário de um documento, como resposta a uma mensagem de QUERY. Esta mensagem chegará a todos os autores do documento, porém será tratada somente pelo autor cujo *nickname* seja igual ao *nickname* presente na mensagem. Esta mensagem apresentará o número do *slot* que o autor que enviou a mensagem de QUERY (e que está tratando esta mensagem de ANSWER) irá editar, ou -1 caso não existam *slots* disponíveis, estando o documento liberado somente para leitura. Existindo um *slot* disponível ou não, o sistema deverá obter o número total de *slots*, criar as seções na interface gráfica e aguardar as mensagens de UPDATE (descritas abaixo) para atualizar o conteúdo de cada *slot* correspondente à uma edição remota.
- **UPDATE|<no_slot>|<nickname_slot>|<titulo_slot>|<texto_slot>**: mensagem enviada por cada autor e que representa uma atualização no texto de cada um. Esta mensagem será enviada por todos os autores periodicamente e todos os autores que receberem a mensagem (com exceção do próprio autor que enviou) deverão tratar esta mensagem. O tratamento consiste em obter os dados de *nickname*, título da seção e texto da seção e refletir isto na interface correspondente ao *slot* em questão.
- **LEAVE|<nickname>|<no_slot>**: mensagem enviada quando o autor clica no botão Sair. Esta mensagem será recebida por todos os autores, porém deverá ser tratada somente pelo autor proprietário do documento. O autor proprietário deverá liberar o *slot* recebido na mensagem de modo a atribuí-lo na chegada da próxima mensagem de QUERY.

Dados a serem mantidos no objeto que representa um autor: um *flag* indicando se ele é um autor proprietário, uma lista de *slots* com um indicativo de ocupação, *nickname* e número do *slot* em uso (-1 caso seja somente um cliente de leitura).

Produtos a Serem Entregues

A entrega do trabalho será composta pelos seguintes itens:

- **Código-fonte do sistema:** código-fonte de todo o sistema. Atentar para comentários. É fortemente sugerido o uso do javadoc para geração automática da documentação.
- **Apresentação do sistema funcionando no laboratório:** nesta fase da avaliação TODOS os componentes da equipe devem estar presentes. Deverá ficar claro qual a participação de cada componente na composição do trabalho. Os componentes poderão ser arguidos.
- **Entrega do questionário IMPRESSO:** o questionário anexado a esta especificação deverá ser entregue de forma impressa no momento da apresentação. Não serão aceitos questionários em disquete ou por e-mail, de modo a facilitar o gerenciamento e a correção.

Critérios para Correção:

- O principal critério a ser avaliado na correção é o grau de atendimento a esta especificação. Para cada item não implementado haverá uma penalização correspondente.
- Trabalhos em atraso sofrerão uma penalidade de 1 ponto por dia útil, até um limite de 4 dias úteis.
- Critérios como criatividade e desempenho da solução serão também avaliados.
- A pontuação será realizada da seguinte forma:
 - Apresentação do sistema e arguição: 7 pontos
 - Questionário: 3 pontos.
- Membros de uma mesma equipe poderão obter notas diferentes, como resultado do processo de arguição.

QUESTIONÁRIO

O objetivo deste questionário é relacionar assuntos teóricos vistos em sala com aspectos vislumbrados neste trabalho prático. As questões deverão ser respondidas apresentando uma análise crítica das funcionalidades implementadas em relação a aspectos tais como: modelos de sistemas distribuídos (síncrono/assíncrono), ordenação de mensagens, serviços de comunicação em grupo e questões de confiabilidade e disponibilidade do serviço.

Questão 1) A aplicação implementada requer algum tipo de ordenação de mensagens ? Justifique sua resposta apresentando situações onde a ordenação é requerida.

Questão 2) Quais as implicações do uso deste sistema em ambientes síncronos e assíncronos ? Que tipo de consequência o sincronismo ou o assincronismo traria para a aplicação ?

Questão 3) A aplicação requer uma plataforma para comunicação em grupo ? Qual seria o papel do Serviço de Membership nesta aplicação ? Qual seria a consequência de uma entrega não-atômica de uma nova visão de grupo ?