MAC 5765 - Sistemas de Middleware Avançados - Segundo Semestre de 2002

Trabalho Final: Melhorias no Gerenciador de Transações do JBoss

Data de Entrega: 29 de novembro de 2002

Neste trabalho você familiarizará com o lado interno de um gerenciador de transações simples[1]. Seu objetivo será melhorar o mini-gerenciador de transações distribuído com o JBoss.

1 Gerenciamento de Transações no JBoss

Atualmente existem no JBoss duas alternativas para gerenciamento de transações:

- Usar um MBean que integra ao JBoss o Tyrex (`http://tyrex.exolab.org`), um gerenciador de transações *open-source* desenvolvido pelo grupo Exolab. O MBean que faz essa integração é o

  `org.jboss.tm.plugins.tyrex.TransactionManagerService`,

  cujos fontes estão no subdiretório `varia`. Por empregar um gerenciador de transações completo, esta é a alternativa mais "pesada".

- Usar o MBean que ativa o mini-gerenciador de transações do JBoss. Este MBean é o

  `org.jboss.tm.TransactionManagerService`,

  cujos fontes estavam no subdiretório `server` das versões de JBoss anteriores à 3.2 e, a partir da versão 3.2, estão no subdiretório `transaction`. Esta alternativa emprega um gerenciador de transações que não tem *logging* (e portanto não faz tratamento de recuperação de quedas) nem propaga o contexto transacional para outros servidores JBoss (ou seja, não trata transações distribuídas envolvendo vários servidores JBoss). Por esse motivo ela é bem mais "leve" que a anterior.

A segunda alternativa é muito mais veloz, mas tem duas desvantagens:

- Ela não é adequada para aplicações tipo "missão crítica", que necessitam de garantia absoluta de que mesmo em caso de queda do sistema, este não ficará num estado inconsistente, com alguma transação executada "pela metade". Não se deve usar um gerenciador de transações sem *logging* para uma aplicação que manipula contas bancárias, por exemplo.

- Ele não funciona para o caso de transações que começam num servidor JBoss e chamam (direta ou indiretamente) *beans* que requerem contexto transacional e são implementados por outros servidores JBoss. Isto não significa que transações distribuídas não funcionam de jeito nenhum. O mini-gerenciador coordena o *two-phase commit* de transações envolvendo múltiplos gerenciadores de recursos XA (bancos de dados ou filas de mensagens), rodando em várias máquinas, desde que todos os *beans* correspondentes a esses recursos estejam no mesmo servidor JBoss.

2 O Que Você Deve Fazer

Sua tarefa é melhorar o mini-gerenciador de transações do JBoss, eliminando alguma das duas restrições acima. Em outras palavras, voce adicionará a esse gerenciador de transações uma das seguintes características, à sua escolha:

- *logging* e tratamento de recuperação de quedas,

- transações distribuídas envolvendo mais de um servidor JBoss.

[1] A expressão "gerenciador de transações relativamente simples" seria mais adequada aqui, pois nenhum sistema desses (por mais simplificado que seja) é *realmente* simples.

(Se você tiver gás, implemente ambas coisas!)

Eis duas das mensagens relacionadas com o assunto deste trabalho que apareceram recentemente na lista jboss-development:

- `http://www.mail-archive.com/jboss-development@lists.sourceforge.net/msg29526.html`
- `http://www.mail-archive.com/jboss-development@lists.sourceforge.net/msg30358.html`

Você deverá apresentar também uma monografia descrevendo o funcionamento e a arquitetura originais do mini-gerenciador de transações, a(s) melhoria(s) implementadas, as principais decisões de projeto tomadas, o emprego de padronizações (como XA e/ou JCA, se for o caso), e as dificuldades encontradas. Sua monografia deve descrever claramente quaisquer mudanças que você introduziu na arquitetura do mini-gerenciador de transações, bem como mudanças que você considera que deveriam ser feitas mas não teve a oportunidade de fazer.

O trabalho de implementação deve ser feito preferencialmente em equipes de duas pessoas. O ideal é o esquema de "programação pareada" (*pair programming*) de XP. Como alguns podem ter restrições de horário que os impeçam de trabalhar assim, aceitarei também implementações individuais. As monografias são estritamente individuais. Fique à vontade para trocar idéias com seu colega de equipe (ou com outras pessoas) sobre o conteúdo da monografia, mas a responsabilidade pela redação é só sua.

Bom trabalho!