

347

IMPLEMENTAÇÃO DE THREADS REMOTAS EM JAVA. Otávio B. Gaspareto, Aury F. Filho, Cristiano A. da Costa (Centro de Ciências Exatas e Tecnológicas – UNISINOS)

Java é uma linguagem atual com suporte a programação distribuída. Entretanto, a API nativa de Java para programação distribuída, denominada RMI (*Remote Method Invocation*), não suporta métodos remotos assíncronos. O suporte a threads facilita a programação distribuída, pois é possível aliar execução paralela e concorrente. Existem vários projetos que lidam com programação distribuída em Java e possuem suporte a execução de métodos remotos assíncronos. Entre eles, podemos citar JavaParty (que possui *threads* remotas também), ProActive PDC (que suporta a execução assíncrona através de *wait-by-necessity*, utilizando para isso objetos futuros) e HORB (que suporta execução assíncrona de métodos através da colocação da chave *_Async* no nome do método). O projeto JavaH possui um protótipo atualmente implementado que suporta distribuição, migração e a execução de métodos remotos síncronos. Para promover concorrência no ambiente distribuído, é apresentada uma alternativa para a implementação de *threads* remotas. Para tanto, é necessário contornar várias restrições da linguagem Java. Por exemplo, uma *thread* não é serializável, o que faz com que uma *thread* remota não possa ser migrada, pois o estado de um objeto com métodos em execução é inconsistente. Para contornar tal problema, introduzimos uma *interface*, denominada ResidentH, na API do JavaH, para que, após implementar essa *interface*, um objeto não possa ser migrado. Caso o programador deseje uma *thread* remota, ele deve estender uma classe da API do JavaH, denominada ThreadH. A classe ThreadH implementa quase a mesma funcionalidade da classe Thread da API do Java, com algumas modificações. Porém, a classe ThreadH serve também como um marcador para o pré-compilador, que gera, a partir da classe que estendeu ThreadH, as classes que irão ser usadas como um *proxy* para acessar a *thread* que está executando remotamente. Futuramente, espera-se prover migração para *threads* remotas, utilizando alternativas que não modifiquem a Máquina Virtual Java (JVM), pois essa é uma restrição do projeto JavaH. (Fapergs/UNISINOS)