

INSTITUTO SUPERIOR DE ENGENHARIA DE
LISBOA

SISTEMAS DISTRIBUÍDOS

Relatório de Trabalho

Brokering System (Série 1)

Autoria:

33724 David RAPOSO

33404 Ricardo MATA

32632 Pedro PEDROSO

Para:

Engº Luís ASSUNÇÃO

15 de Maio de 2014

Conteúdo

1	Introdução	1
2	Arquitetura	1
3	Implementação	1

1 Introdução

O primeiro trabalho desta cadeira pede que desenvolvamos um sistema de brokering entre clientes e processadores, sem que estes se conheçam. Serve o presente relatório para explicar a nossa implementação e para discutir as nossas soluções.

2 Arquitetura

A solução tem 3 intervenientes distintos. Um cliente que pretende que sejam realizadas tarefas remotamente, um (ou mais) serviço de execução de trabalhos que irá executar trabalhos submetidos pelo cliente, e um intermediário que irá delegar trabalhos aos serviços de execução. A partir deste ponto passaremos a referir-mo-nos ao intermediário como *broker* e aos serviços de execução como *workers* ou *worker*.

O cliente pretende que alguém lhe faça um trabalho. Para isso envia um pedido ao *broker* a dizer qual o trabalho que pretende que seja executado. O *broker* decide então qual o *worker* mais adequado para executar essa tarefa, e envia-lhe o pedido do cliente. Assim que seja possível o worker inicia a execução do trabalho, e assim que o trabalho estiver concluído, notifica o cliente e o broker de que o trabalho foi concluído.

3 Implementação

Configurámos cada uma das partes para se ligarem a uma porta TCP como um *Well Known Type*. Isto permite que cada uma das partes consiga comunicar entre si. Esta configuração define os *end-points* onde cada parte irá aceder para poder comunicar com outras partes, e o tipo de canal que é usado para a comunicação.

O cliente regista o *Well Known Type* do *broker* à interface partilhada. Isto permite criar uma interface de comunicação onde cada acesso feito pelo cliente vai ser tratado não localmente, mas pelo *broker*. Um pedido de trabalho é então enviada ao *broker* através dessa interface. Para o pedido de trabalho é criado uma instância do tipo *Job*. Este tipo tem a seguinte informação:

1. O nome do trabalho que se quer executar.
2. A localização dos ficheiros de input e output.
3. O identificador do trabalho no *broker*, para pedidos de estado.
4. Uma interface para que o *worker* possa avisar diretamente o cliente da conclusão do trabalho.

A existência da interface de comunicação para o worker torna este tipo num *proxy*. Isto requer um cuidado extra na configuração dos canais de comunicação (o canal tem de ser configurado como "full" para que a comunicação seja delegada pelo broker). É também de notar que esta solução foi implementada tendo em conta que os workers vão estar na mesma máquina que o broker e o cliente (apesar de se simular um ambiente remoto), de forma a que uma solução completamente remota iria necessitar de outra abordagem na passagem de parâmetros/receção de resultados entre o cliente e os *workers*.

Quando o objeto chega ao broker, é colocado num mapa de jobs. Este mapa contém informação sobre os *jobs* submetidos para que seja possível resubmetê-los caso haja necessidade de tal.