

Relatório PPC – Atividade 4

Renaming files and folders



Ciências
ULisboa

Henrique Barata fc54387

Para a realização deste projeto foi utilizado o Actor Model. Este permitiu a paralelização do programa evitando bloqueios quando as diferentes threads comunicavam entre si e lidar com erros facilmente.

Neste caso foram criados vários Actors, RenameActors, um para cada folder ou file, que tinham o papel de alterar o seu nome. Assim os Actors seguiam uma organização em forma de tree, igual a dos ficheiros. Foi criado um outro Actor, o Costumer, que realizava o envio de mensagens para o RenameActor root, para começar o processo de renomeação dos ficheiros ou para acabar o programa quando este recebe resposta dos seus pedidos.

Neste programa não há limite de depth, ou seja, pode haver várias camadas de folders. Isto garante-se, pois, cada RenameActor verifica se a sua folder é ou não um diretório vazio, e cria descendentes, novos RenameActors para cada uma das folders dentro do seu diretório. Foi escolhida esta arquitetura para poder funcionar para qualquer número de folders mesmo tendo diferentes números de descendentes, evitando assim a criação de Actors desnecessários.

Para garantir a serialização das operações, cada Actor pai esperava receber resposta de todos os filhos para dar continuidade. Assim cada Ator apenas mudava o nome da sua pasta quando todos os filhos acabassem. Este continuaria a receber pedidos e a guardar para realizar mais tarde.

Para paralelizar o programa foi criado um Actor para cada folder, assim várias threads podem correr ao mesmo tempo. Neste caso não acho que criar uma thread para cada folder seja a opção mais rentável, devido a complexidade da tarefa de cada uma.

Como cada folder era tratada apenas por um Actor e este espera que todos os filhos acabem, também não iria haver problemas de concorrência, onde achei que não seria necessário implementar qualquer synchronization operations. Mas depois de testar, e ao reparar que o numero de alterações não era sempre o mesmo, deparei-me com um problema, um ficheiro que eu fosse alterar o nome, não alteraria se este nome já existisse, por exemplo o file_1 que ao trocar _1 por _13 (como no exemplo do professor, mas nestes ficheiros o 0 não estava, por isso optei por colocar o _ para substituir apenas o primeiro e não todos os 1s), não era alterado pois já existia um file_13. Para resolver isso apenas coloquei um temporizador, não infinito, pois pode haver ficheiros que não serão alterados, e se existisse um ficheiro com aquele nome, o Actor iria esperar e tentar novamente, várias vezes.

Resolvi a questão de recursividade da estrutura dos ficheiros com a função listFiles(), que retornava o conjunto de files no diretório. Foi feito em todos os subdiretórios e files a partir da root.

No total houve 122 alterações de _1 para _13.