

# Projeto

## Sistemas Operativos

### Grupo 2

**Nomes:**

Sérgio Miguel Pereira Carvalho (A89313)

Carlos Miguel Correia de Oliveira (A89383)

José Miguel Faria Pedroso (A89395)

**Instituição:** Universidade do Minho

**Curso:** MIETI- Mestrado Integrado em Telecomunicações e Informática

**Unidade curricular:** Sistemas Operativos

Guimarães, 17 de janeiro de 2021

# Índice

1.Introdução.....	3
2. Descrição da solução e da comunicação entre processos.....	4
2.1 Servidor.....	4
2.2 Cliente .....	5
2.3 Considerações finais .....	5
3. Estrutura de processos.....	6
4. Limitações da solução desenvolvida .....	7
5. Conclusão .....	8

## 1. Introdução

No âmbito da Unidade Curricular de Sistemas Operativos, foi-nos pedido para realizar um projeto como forma de avaliação da componente prática da UC. Este consiste na implementação de um serviço capaz de transformar ficheiros de áudio por aplicação de uma sequência de filtros, no qual o serviço a implementar devera ser constituído por um servidor e um por cliente que deverão comunicar por pipes com nome. Tanto o cliente como o servidor apenas disponibilizarão uma interface de linha de comando.

O grupo considera que este projeto é algo que vai de acordo com a matéria que foi abordada ao longo do semestre pelos docentes, exigindo que o grupo tenha a capacidade de entender e interligar todos os conceitos abordados de forma a cumprir todos os objetivos pretendidos.

## 2. Descrição da solução e da comunicação entre processos

Na ótica de descrever da melhor forma aquilo que foi desenvolvido pelo grupo, iremos dividir este tópico em dois pontos, sendo estes clientes e servidor, pois acreditamos que a solução por nós pensada e implementada apesar de funcionar como um todo, as suas individualidades têm características distintas que merecem o devido destaque.

### 2.1 Servidor

O desenvolvimento desta área do projeto tem como ponto central a comunicação IPC onde foi decidido que no arranque do servidor, iriam ser criados dois pipes com nome que serão conhecidos por todos os clientes, um para os clientes poderem realizar os seus pedidos, outro para o servidor enviar o seu estado atual.

Ao receber um pedido de transformação de um ficheiro, o servidor terá duas tarefas, a primeira será verificar se o pedido foi enviado de forma correta, a segunda será verificar se todos os filtros solicitados se encontram disponíveis, sendo estas verificadas este cria um processo filho, pois será aqui que o pedido será tratado, isto é, por cada pedido recebido no servidor, é criado um processo descendente que irá efetivamente tratar deste pedido. Dentro deste ponto a ferramenta utilizada para fazer uma comunicação IPC entre este filho e os filtros pedidos pelo cliente, será um pipe anónimo que juntamente com um correto uso de redireccionamentos, fará que os filtros sejam sequenciais em termos de input e output, tendo em conta que no primeiro filtro o seu stdin será o ficheiro enviado pelo cliente e no último o seu stdout será o pipe com nome criado especificamente pelo cliente que realizou o pedido, sendo que este tem a tarefa de guardar o ficheiro.

Antes de falarmos daquilo que o servidor realiza quando recebe um pedido de estado, convém referir que todos os processos filhos que são criados para gerir os pedidos, têm o seu pid guardado no pai. Com esta informação sempre preservada, o servidor ao receber um pedido de estado irá enviar um sinal SIGUSR1 para todos os processos filhos que irão enviar para o cliente, via pipe com nome, o respetivo pedido que estão a processar, no caso do processo pai irá enviar o uso dos filtros no momento do pedido.

## 2.2 Cliente

Esta área do projeto foi tida como secundária numa primeira fase, isto é numa ótica de implementação, visto que o corpo central da aplicação seria o servidor, porém a sua solução foi desde logo desenhada e alinhada com aquilo que o servidor seria capaz de responder, pelo que iremos de seguida apresentar as suas ideias principais, bem como a comunicação IPC que estabelece com o servidor.

Como visto no tópico anterior, o cliente apenas tem dois modos de interação com o servidor, realizar um pedido ou obter a informação do estado do mesmo. Para podermos implementar estas funcionalidades recorreremos em ambos os casos a pipes com nome, sendo que na primeira função, o cliente ao realizar o pedido irá criar um pipe com nome único, isto porque este terá o seu pid como identificador. Após submeter o pedido, o processo cliente pai morre, ficando o seu filho com a função de receber do pipe criado anteriormente a resposta do servidor e posteriormente criar o ficheiro final, eliminando de seguida o pipe temporário.

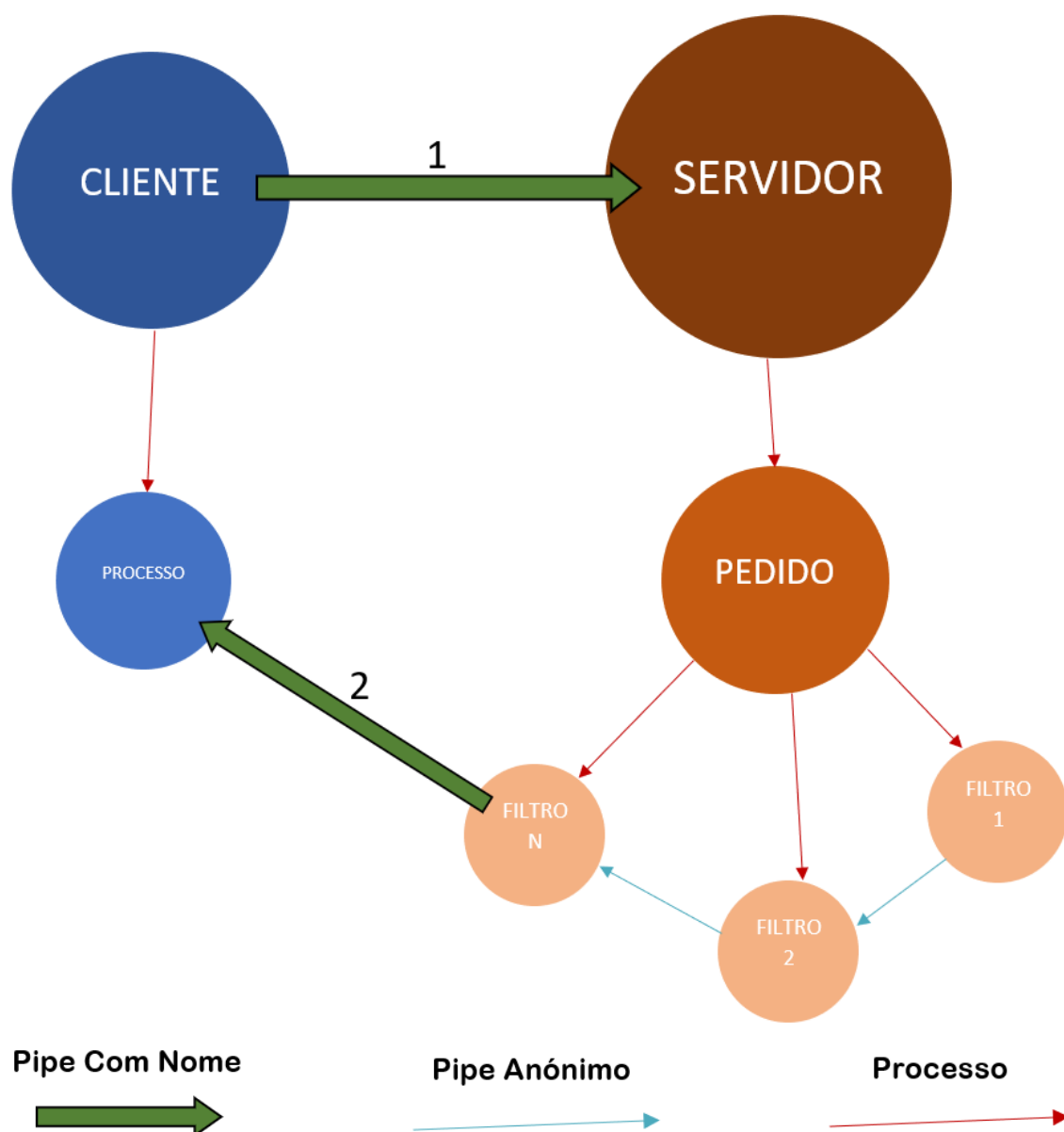
Em relação à segunda funcionalidade, o cliente envia para o servidor um pedido de estado, pelo que irá receber esta resposta através do pipe com nome criado, especificamente, para este feito pelo servidor. Após imprimir para o cliente este mesmo estado, fecha o pipe com nome e termina a sua execução.

## 2.3 Considerações finais

O objetivo do grupo foi desde o início tentar que a solução desenvolvida conseguisse reunir o maior número de objetivos propostos pelo que foram com sucesso implementadas as seguintes funcionalidades (as mais importantes):

- O servidor inicia o tratamento dos pedidos de forma sequencial e suporta o processamento concorrente dos mesmos.
- O servidor termina de forma graciosa o seu funcionamento, isto é, acaba os pedidos que tem em espera e só no fim encerra o processo.
- Evita a criação de ficheiros temporários.
- O processamento de um pedido não é iniciado pelo servidor se não existirem instâncias disponíveis de todos os filtros necessários.

### 3. Estrutura de processos



A figura acima representa o nosso problema de uma forma geral e generalizada, apresentando os pipes com nome, pipes anónimos e processos utilizados. Tal como mencionado no ponto 2.1 e 2.2 o servidor recebe informação do cliente através do pipe com nome “1”, o qual é conhecido por todos os clientes existentes. O servidor cria um pedido, este trata-se de um descendente do servidor, neste existirá uma comunicação entre filhos através de pipes anónimos e terminando no pipe com nome “2”, o qual irá transmitir toda a sua informação para o processo concorrente criado pelo cliente, este pipe é apenas conhecido por este cliente e pelo servidor.

## 4. Limitações da solução desenvolvida

Neste tópico serão abordadas algumas limitações que foram encontradas ao longo da implementação e que não conseguiram ser resolvidas a tempo da entrega deste trabalho, porém o grupo deu sempre o seu melhor no sentido de as minimizar e muitas vezes as resolver. Sendo que é sentido pelos vários elementos que constituem este grupo, que este projeto foi desenvolvido em corrida contra o tempo, devido principalmente à grande densidade de momentos de avaliação, não querendo dizer que houve um desinteresse por este momento avaliativo, mas sim que foram feitos todos os esforços para que o trabalho fosse o melhor possível, ficando a sensação que com um pouco mais de tempo o trabalho podia ter atingido um nível bastante elevado.

No seguimento do parágrafo anterior, as limitações encontradas na solução e implementação do grupo são:

- O cliente não conseguir passar como input para o servidor, um ficheiro que contenha no seu nome espaços em branco.
- Na eventualidade do servidor não ter todos os filtros disponíveis, o pedido é descartado em vez de ficar por exemplo em fila de espera até haver disponibilidade no servidor.
- Não foi utilizado o ficheiro Makefile para compilar e efetuar testes ao sistema desenvolvido.

Para concluir este capítulo, o grupo tem a referir que apesar destas limitações, outras funcionalidades foram implementadas que oferecem grandes recursos ao servidor naquilo que diz respeito a concorrência e desempenho, pelo que no final sentimos que foi um trabalho bastante positivo.

## 5. Conclusão

O presente trabalho iniciou-se com o planeamento do projeto, bem como a correta interpretação de todos os aspetos que eram pedidos, tanto os obrigatórios como os facultativos. Tivemos sempre em conta alguns pontos que foram abordados ao longo das aulas teóricas e práticas, aliando a isto, algumas pesquisas que foram realizadas com o objetivo de resolver alguns problemas que fomos obtendo ao longo do projeto.

Ao longo do processo, tivemos algumas dificuldades que, como referido em cima, foram prontamente esclarecidas e explicadas com o auxílio de uma pequena pesquisa, permitindo-nos assim ultrapassá-las e concluir as tarefas impostas.

Em suma, este foi um projeto ambicioso que acompanhou sempre a nossa evolução ao longo da UC, fazendo-nos desenvolver capacidades de superação e trabalho de equipa, bem como os nossos conhecimentos desenvolvidos ao longo do semestre, onde foi posto à prova o nosso espírito de entreajuda, e onde tivemos de aplicar todos os conhecimentos adquiridos. Embora não sendo esta a altura ideal do semestre pois estamos com bastantes momentos de avaliação, destacamos todos os esforços realizados pelos docentes que deram bastante ajuda na elaboração deste projeto e que ainda atenderam ao nosso pedido de adiar a entrega do projeto. Como tal, sentimos que demos o nosso melhor nas circunstâncias já evidenciadas e, por isso, achamos que foi um trabalho positivo.