

Mansol-dl

Este proyecto ha tenido como objetivo el diseño de un sistema cliente-servidor para la extracción de ficheros de audio a partir de la URL de clips de Youtube. Este fork ha sido nombrado **mansol-dl** y ha sido realizado por:

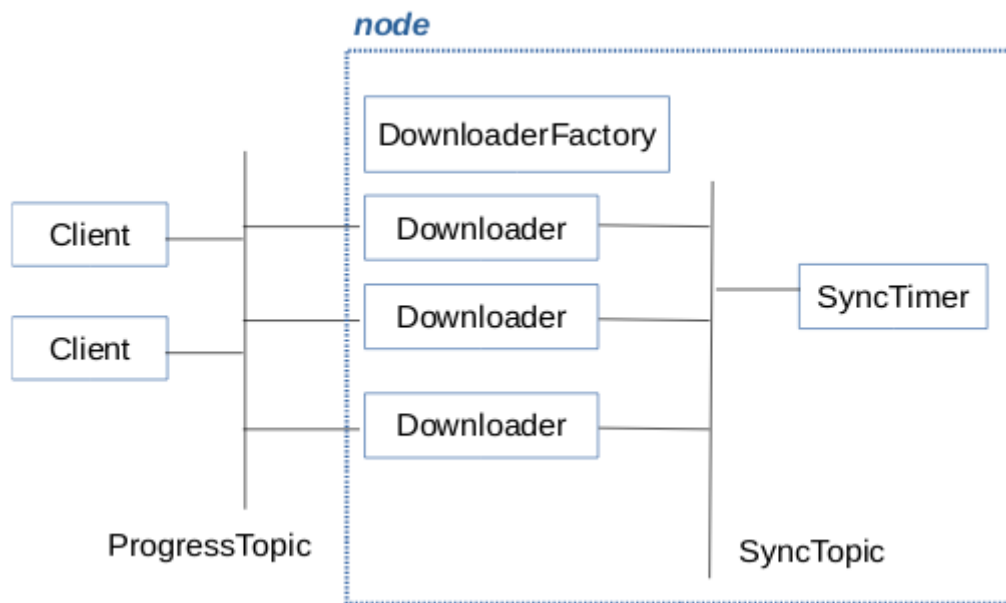
- Sergio Herrera Piqueras (sherrerap)
- Pablo Rodríguez Solera (SOLERA)

Antes de nada, cabe destacar que el proyecto ha sido realizado con el uso de repositorios git, y se han hecho públicos para el uso libre del mismo. Puedes verlo [aquí](#).

Vamos a desglosar los apartados del proyecto:

Arquitectura del proyecto

El proyecto se compone de 3 tipos de componentes: servidores, clientes y canales de eventos, y se ha seguido el esquema de la arquitectura proporcionada:



Con la excepción de la parte de la derecha del esquema: el canal de eventos SyncTopic para la sincronización de servidores no ha sido implementado.

Servidores de descarga

Primero, se ha implementado la interfaz DownloadScheduler:

- La recepción de peticiones de descarga de los clientes se ha resuelto mediante la implementación del método *addDownloadTask(string url)*. A su vez, la información sobre la descarga será comunicada al cliente mediante un canal de eventos, mientras que el servidor también mostrará la información sobre las solicitudes recibidas.
- El almacenamiento de archivos obtenidos tras el proceso de extracción del audio a un directorio local del servidor se ha realizado mediante la creación de una carpeta Descargas en el directorio donde se ejecuta el mismo.

- Se ha implementado el método *Transfer* get(string song)* para la transferencia de archivos desde el directorio local del servidor (*/Descargas*) hasta el directorio local del cliente (*el directorio raíz donde se ejecuta el cliente*).
- La recepción de peticiones de listado de canciones se procesa en el servidor con la implementación de *getSongsList()* y la lista es devuelta al cliente.

Cientes

Los clientes realizan las siguientes tareas gracias a la implementación de un menú para la mejor usabilidad de las opciones del sistema:

- Solicitación de descarga de una URL correspondiente a un clip de Youtube (*opción 1*).
- Recepción de las notificaciones relativas al proceso de la descarga y muestra al usuario, implementado mediante *notify(ClipData clipData)*.
- Obtención de los ficheros de audio (*opción 3*).
- Obtención de la lista completa de ficheros descargados (*opción 2*).

Canales de eventos

La comunicación entre clientes y servidores se realiza a través de un canal de eventos *ProgressTopic*. Este canal realiza 4 llamadas de tipo notify, que son recibidas y mostradas por el cliente:

- Pending.
- InProgress.
- Done.
- Error¹.

Factoría de servidores de descarga

El despliegue de los servidores de descarga se realiza de manera dinámica mediante el uso de una factoría *SchedulerFactory*, con un método *DownloadScheduler* make(string name)*. Nuestro sistema está diseñado de tal forma que cada vez que el cliente realice una solicitud - ya sea de descarga con URL, de descarga del servidor, etc- la factoría le proporcione un nuevo objeto *DownloaderScheduler*².

Gestión del servicio mediante IceGrid

Tenemos todo el sistema gestionado mediante IceGrid. Para arrancar el sistema, haremos lo siguiente:

- `make start`
- `make copy-binaries`
- `make run-node`
- Arrancar el servidor mediante la interfaz IceGrid.
- Ejecutar `make run-iceclient 0 ./Client.py --Ice.Config=client.config Downloader1`

1 Esta funcionalidad no es completamente funcional: cuando se inserta una url inválida, se comprueba la excepción lanzada y se debería notificar el error, pero esto no ocurre. Es un bug conocido.

2 Los objetos se eliminan una vez terminada su tarea.

Gestión del servicio sin IceGrid

También tenemos la opción de desplegar el sistema sin el uso de IceGrid. Para arrancar el sistema, haremos lo siguiente:

- `make start`
- `make run-server`
- **Ejecutar** `./Client.py "Downloader1 -t -e 1.1:tcp -h 192.168.1.36 -p 4061 -t 60000" -Ice.Config=client.config3`

³ La ip dependerá de la máquina donde se ejecute el servidor principal. Por ello, deberá modificarse para utilizar la ip correspondiente.