

Anexo I

Arquitectura EDECAN: Componentes y ficheros de configuración.

Componentes básicos de la arquitectura

La interfaz proporciona `gestor.exe`, `super_server.exe` y `service.exe`, 3 binarios compilados para Windows y Linux, además de un controlador del sistema. El resto de módulos deben ser creados en función del sistema que se quiera montar.

Servidor de servicios

El servidor de servicios alberga uno o varios módulos de la arquitectura, llamados de aquí en adelante servicios, y se corresponde con una máquina, identificada por tanto a través de su dirección IP o su nombre y un puerto a través del cual acceder. Tendrá lanzado un proceso “superserver” que espera la llegada de conexiones entrantes y lanza un ejecutable “service.exe” tras la recepción de cada petición de conexión, transfiriendo a dicho ejecutable la estructura de datos del *Socket* que acaba de abrir y que le conecta con el *hub* central o gestor de comunicaciones.

Ejecutable del servicio, service.exe

Es el encargado de la gestión de las comunicaciones entre el servicio y el gestor de comunicaciones. Una vez que es lanzado por el *superserver*, recibe de éste la estructura del *socket* y se configura como un servicio genérico, sin funcionalidad concreta. Tras la llegada del paquete de conexión, carga la librería dinámica apropiada para convertirse en un módulo específico, definida por el conjunto de funciones que contenga dicha librería. Internamente, está compuesto por dos hilos que gestionarán el envío y la recepción de paquetes, hacia y desde el gestor de comunicaciones. El hilo de envío está constantemente monitorizando la existencia de paquetes por enviar procedentes de las diversas funciones que componen el servicio. El hilo de recepción, procesará los paquetes que el servicio recibe desde el gestor de comunicaciones procedentes de otros servicios, y ejecutará las funciones apropiadas para la recepción de cada uno de los paquetes.

Gestor de comunicaciones

Es el nodo central de la red y, tras la puesta en marcha del sistema, se convierte en un *router* encargado de distribuir los paquetes que recibe de cada uno de los diferentes servicios que componen el sistema, a los que se conecta como cliente de la comunicación. Además, se conectará con un módulo controlador, a través del cual se pueden realizar funciones de control y comando del sistema. Principalmente, tiene 3 funciones que se desglosan a continuación:

- Reencaminar todos los paquetes que le llegan de un servidor hacia el destino. El re-encaminamiento lo realiza por defecto a partir de una tabla de rutas estática, definida en “gestor.xml”, aunque también existe la posibilidad de

encaminar los paquetes de forma dinámica (a partir de la información del propio paquete).

- Actúa como un servidor, esperando conexiones de algún controlador para montar un sistema, o modificar uno ya montado.
- A petición de algún controlador conectado a él, el gestor puede realizar modificaciones sobre los servicios propios del sistema, como conectarlos, desconectarlos, darlos de alta o de baja.

Controlador

Existe una aplicación gráfica (también está disponible su versión a través de línea de comandos) a partir de la cual llevar a cabo el envío de órdenes al gestor de comunicaciones. A través de ella también es posible monitorizar el funcionamiento del sistema a través del estado de los diferentes módulos o la velocidad de transmisión de cada uno de ellos. Permite asimismo la visualización y eventual modificación de la tabla de rutas del sistema.

Ficheros de configuración

Fichero de servicios disponibles (“servicios.xml”)

Es el fichero que contiene los distintos tipos de servicios en los que puede convertirse ‘service.exe’ (listado de los servicios que componen un sistema). Para cada uno de ellos, define su nombre y el fichero de configuración asociado a él. Este fichero se consulta por ‘service.exe’ cuando el gestor de comunicaciones envía el paquete de conexión en el que indica el servicio con el que desea establecer una sesión.

Un ejemplo de fichero de configuración es el siguiente:

```
<?xml version="1.0" ?>
<servicios>
  <servicio_circuito_virtual>config_files/services/servicio_circuito_virtual.xml</servicio_circuito_virtual>
  <ejemplo_uso>config_files/services/ejemplo_uso.xml</ejemplo_uso>
  <audio>config_files/services/audio.xml</audio>
  <Param_extractor>config_files/services/Param_extractor.xml</Param_extractor>
  <VAD>config_files/services/VAD.xml</VAD>
  <Segmentador>config_files/services/Segmentador.xml</Segmentador>
  <sdteam_reco>config_files/services/sdteam_reco.xml</sdteam_reco>
</servicios>
```

Estos son los servicios disponibles para nuestro sistema “Módulo análisis voz”. Hay que tener en cuenta, que junto con la declaración en este fichero del servicio, deberá existir una librería dinámica asociada, convenientemente compilada para el sistema operativo y la plataforma usada, así como un fichero de configuración de cada servicio.

Fichero de configuración del servicio (p.e. “sdteam_reco.xml”)

El fichero de configuración de un servicio contiene una serie de comandos, asociados cada uno de ellos con una función. Además, también contiene el nombre de la librería dinámica asociada a ese tipo de servicio. Este es el fichero de configuración del reconecedor:

```
<?xml version="1.0" ?>
<servicio nombre="sdteam_reco">
  <dll>sdteam_reco_service.dll</dll>
  <so>libsdtteam_reco_service.so</so>
  <comando nombre="conexion">
    <funcion>entrada</funcion>
  </comando>
  <comando nombre="desconexion">
    <funcion>salida</funcion>
  </comando>
  <comando nombre="desconexion_caidagestor">
    <funcion>salida_caidagestor</funcion>
  </comando>
  <comando nombre="audio_in_bin">
    <funcion>add_samples_reco_s</funcion>
  </comando>
  <comando nombre="mfcc_in_bin">
    <funcion>add_frame_reco_s</funcion>
  </comando>
  <comando nombre="get_results">
    <funcion>get_results_reco_s</funcion>
  </comando>
  <comando nombre="get_results_time">
    <funcion>get_info_results_reco_s</funcion>
  </comando>
  <comando nombre="reinitiate_reco">
    <funcion>reinitiate</funcion>
  </comando>
</servicio>
```

Cuando llega un comando al servicio, este ejecutará la función que le corresponde, a partir de este fichero de configuración (p.e.: si llega un comando “desconexión”, el servicio ejecutará la función “salida”).

Fichero de configuración del sistema (p.e. “gestor_demo_stma_completo.xml”)

El fichero de configuración de un sistema contiene una serie de servicios que serán los componentes del sistema, así como la tabla de rutas asociada a dichos servicios. Además, para cada servicio se indica el nombre del mismo, el puerto a través del cual se accede, el nombre de la máquina en la cual reside y otros campos que aportan información sólo al usuario humano del sistema como la ubicación geográfica de la máquina. Además para determinados servicios es preciso indicar una configuración

previa para que sea posible su uso. Existen tres modos de indicar la configuración de un servicio:

- A través de un fichero residente en la misma máquina en la que reside el servicio, siendo dicha ruta relativa al binario ejecutable del servicio. Ejemplo de un servicio cuya configuración se realiza a través de la indicación de un fichero local, la veremos en el param_extractor y el sdteam_reconocedor.
- A través de un fichero ubicado en otra máquina distinta de aquella en la que reside el servicio, siendo dicha ruta una URL válida. En ella deben especificarse tanto el nombre de la máquina como la ruta hasta el fichero.
- A través de la indicación embebida de la información necesaria para la configuración del servicio. Así, será en el propio campo configuración donde se insertará todo lo necesario para configurar el servicio. Como por ejemplo la indicación <delay_ratio> del servicio audio.

Como se ha comentado previamente, junto con la definición de los servicios, el archivo que configura el sistema debe contener la tabla de rutas a través de la cual se lleva a cabo el encaminamiento por defecto de los paquetes. Gracias a ella, el gestor de comunicaciones enviará a los módulos en ella indicados los paquetes procedentes de cada servicio.

Podemos ver todo esto en gestor_demo_stma_completo.xml:

```
<?xml version="1.0" ?>
<gestor>
  <servicios>
    <servicio nombre="servicio_circuito_virtual">
      <puerto>6611</puerto>
      <direccion>localhost</direccion>
      <localizacion>CPS, GTC2PC8</localizacion>
      <configuracion>NONE</configuracion>
      <tipo_servicio>servicio_circuito_virtual</tipo_servicio>
    </servicio>
    <servicio nombre="param_extractor">
      <puerto>6611</puerto>
      <direccion>localhost</direccion>
      <localizacion>CPS, GTC2PC8</localizacion>
      <configuracion>
        <service_contains_file>config_files\services\etsi_T3_16k.cfg</service_
        _contains_file>
      </configuracion>
      <tipo_servicio>Param_extractor</tipo_servicio>
    </servicio>
    <servicio nombre="ejemplo_uso">
      <puerto>6611</puerto>
      <direccion>localhost</direccion>
      <localizacion>CPS, GTC2PC8</localizacion>
```

```

    <configuracion>NONE</configuracion>
    <tipo_servicio>ejemplo_uso</tipo_servicio>
</servicio>
<servicio nombre="audio">
    <puerto>6611</puerto>
    <direccion>localhost</direccion>
    <localizacion>CPS, GTC2PC8</localizacion>
    <configuracion>
        <delay_ratio>1</delay_ratio>
    </configuracion>
    <tipo_servicio>audio</tipo_servicio>
</servicio>
<servicio nombre="vad">
    <puerto>6611</puerto>
    <direccion>localhost</direccion>
    <localizacion>CPS, GTC2PC8</localizacion>
    <configuracion>NONE</configuracion>
    <tipo_servicio>VAD</tipo_servicio>
</servicio>
<servicio nombre="segmentador">
    <puerto>6611</puerto>
    <direccion>localhost</direccion>
    <localizacion>CPS, GTC2PC8</localizacion>
    <configuracion>NONE</configuracion>
    <tipo_servicio>Segmentador</tipo_servicio>
</servicio>
<servicio nombre="sdteam_reconocedor">
    <puerto>6611</puerto>
    <direccion>localhost</direccion>
    <localizacion>CPS, GTC2PC8</localizacion>
    <configuracion>
        <destino_micro>microfono</destino_micro>
        <filereco>config_files/services/sdteam-
reco/radio/reco_k_tree_trifon_16G_radio.cfg</filereco>
    </configuracion>
    <tipo_servicio>sdteam_reco</tipo_servicio>
</servicio>
</servicios>
<tabla_routing>
<ejemplo_uso>
    <audio>1</audio>
    <param_extractor>1</param_extractor>
    <ejemplo_uso>0</ejemplo_uso>
    <servicio_circuito_virtual>0</servicio_circuito_virtual>
    <vad>0</vad>
    <segmentador>0</segmentador>

```

```

    <sdteam_reconocedor>0</sdteam_reconocedor>
</ejemplo_uso>
<servicio_circuito_virtual>
  <param_extractor>0</param_extractor>
  <servicio_circuito_virtual>0</servicio_circuito_virtual>
  <vad>0</vad>
  <ejemplo_uso>0</ejemplo_uso>
  <audio>0</audio>
  <segmentador>0</segmentador>
  <sdteam_reconocedor>0</sdteam_reconocedor>
</servicio_circuito_virtual>
<vad>
  <param_extractor>0</param_extractor>
  <servicio_circuito_virtual>0</servicio_circuito_virtual>
  <vad>0</vad>
  <ejemplo_uso>1</ejemplo_uso>
  <audio>0</audio>
  <segmentador>1</segmentador>
  <sdteam_reconocedor>0</sdteam_reconocedor>
</vad>
<audio>
  <param_extractor>1</param_extractor>
  <ejemplo_uso>1</ejemplo_uso>
  <audio>0</audio>
  <servicio_circuito_virtual>0</servicio_circuito_virtual>
  <vad>0</vad>
  <segmentador>0</segmentador>
  <sdteam_reconocedor>0</sdteam_reconocedor>
</audio>
<param_extractor>
  <param_extractor>0</param_extractor>
  <servicio_circuito_virtual>0</servicio_circuito_virtual>
  <vad>0</vad>
  <ejemplo_uso>0</ejemplo_uso>
  <audio>0</audio>
  <segmentador>0</segmentador>
  <sdteam_reconocedor>0</sdteam_reconocedor>
</param_extractor>
<segmentador>
  <servicio_circuito_virtual>0</servicio_circuito_virtual>
  <ejemplo_uso>0</ejemplo_uso>
  <audio>0</audio>
  <param_extractor>0</param_extractor>
  <vad>0</vad>
  <segmentador>0</segmentador>

```

```

    <sdteam_reconocedor>0</sdteam_reconocedor>
</segmentador>
<sdteam_reconocedor>
    <servicio_circuito_virtual>0</servicio_circuito_virtual>
    <ejemplo_uso>1</ejemplo_uso>
    <audio>0</audio>
    <param_extractor>0</param_extractor>
    <vad>0</vad>
    <segmentador>0</segmentador>
    <sdteam_reconocedor>0</sdteam_reconocedor>
</sdteam_reconocedor>
</tabla_routing>
</gestor>

```

El *delay* del audio es necesario para evitar en el sistema completo comportamientos no deseados, como muestras que siguen llegando tras cerrar el archivo y que por tanto no son tenidas en cuenta en el análisis. Por eso es mejor fijar la velocidad de envío, bastando con que el retardo sea 1 (valor por defecto en caso de no indicarse configuración alguna), es decir, el tiempo de transmisión es el mismo que el de reproducción. Con *delays* menores que 1 sigue funcionando igual pero no merece la pena disminuirlo porque el tiempo total de procesado lo da el eslabón más lento de la cadena. Aunque los dos primeros módulos (VAD y segmentador) son capaces de procesar a la velocidad de envío, el análisis del reconocedor tarda más que su duración.

Hasta aquí hemos visto los archivos de configuración relativos al sistema principal, el de procesado del audio, pero bajo la arquitectura EDECAN teníamos montado también el servicio web. Este es su gestor:

```

<?xml version="1.0"?>
<gestor>
  <servicios>
    <servicio nombre="puente_texto">
      <puerto>6611</puerto>
      <direccion>voz01.cps.unizar.es</direccion>
      <localizacion>Cluster</localizacion>
      <configuracion>
        <port>6000</port>
      </configuracion>
      <tipo_servicio>puente_texto</tipo_servicio>
    </servicio>
    <servicio nombre="ejemplo_texto">
      <puerto>6612</puerto>
      <direccion>voz01.cps.unizar.es</direccion>
      <localizacion>Cluster</localizacion>
      <configuracion>NONE</configuracion>
    </servicio>
  </servicios>
</gestor>

```

```
    <tipo_servicio>ejemplo_texto</tipo_servicio>
  </servicio>
</servicios>
<tabla_routing>
  <ejemplo_texto>
    <ejemplo_texto> 0 </ejemplo_texto>
    <puente_texto> 1 </puente_texto>
  </ejemplo_texto>
  <puente_texto>
    <ejemplo_texto> 1 </ejemplo_texto>
    <puente_texto> 0 </puente_texto>
  </puente_texto>
</tabla_routing>
</gestor>
```