

**ESCUELA TÉCNICA SUPERIOR  
DE INGENIEROS DE TELECOMUNICACIÓN**

**Universidad de Málaga**

**INGENIERÍA DE TELECOMUNICACIÓN**

**RESUMEN DEL PROYECTO  
FIN DE CARRERA**

**Sistema cliente-servidor  
para datos de tráfico aéreo**

## **Contenidos**

|   |           |
|---|-----------|
| <b>CONTENIDOS</b> .....                       | <b>1</b>  |
| <b>INTRODUCCIÓN</b> .....                     | <b>2</b>  |
| 1. PROCESO DE REALIZACIÓN.....                | 3         |
| 2. CONCEPTOS PREVIOS.....                     | 3         |
| <b>ARQUITECTURA DEL SISTEMA</b> .....         | <b>4</b>  |
| 1. SERVIDOR.....                              | 4         |
| 1.1. <i>Adquisición de datos</i> .....        | 5         |
| 1.2. <i>Comunicaciones</i> .....              | 6         |
| 1.3. <i>Gestión y control</i> .....           | 7         |
| 2. CLIENTES DEL SISTEMA.....                  | 8         |
| 2.1. <i>Cliente Copia</i> .....               | 9         |
| 2.2. <i>Cliente Tabla de estimación</i> ..... | 9         |
| 2.3. <i>Cliente Consola de radar</i> .....    | 10        |
| 3. EMPAQUETADO Y DISTRIBUCIÓN.....            | 11        |
| <b>BANCO DE PRUEBAS DEL SISTEMA</b> .....     | <b>11</b> |
| <b>LÍNEAS ACTUALES DE DESARROLLO</b> .....    | <b>13</b> |
| <b>CONCLUSIONES</b> .....                     | <b>13</b> |

## **Introducción**

Este documento resume el proyecto titulado “Sistema cliente-servidor para datos de tráfico aéreo”, realizado por Juan José Escobar Mira y dirigido por Gabriel Valencia Miranda, profesor del Departamento de Tecnología Electrónica de la Universidad de Málaga.

El proyecto fue desarrollado bajo el contrato de colaboración de la universidad con Aeropuertos Españoles y Navegación Aérea (AENA) en el Aeropuerto de Málaga. De este contrato el principal responsable y coordinador es Alfredo García Lopera, también profesor del Departamento de Tecnología Electrónica.

El proyecto surgió por la fuerte necesidad que tienen los distintos departamentos del aeropuerto de disponer de la información de tráfico aéreo. Ésta sólo está disponible a través de las consolas de radar, cuyo elevado coste de instalación y mantenimiento hacen que su uso quede restringido al control aéreo. Ello obliga a emplear la identificación visual para el resto de tareas de operación y gestión, lo que plantea serios inconvenientes: además de ser poco eficiente y fiable, la necesidad de confirmación visual reduce el tiempo de reacción disponible y ello obliga a realizar la planificación y asignación de recursos a corto plazo.

El objetivo del proyecto fue pues realizar un sistema que permitiese el acceso a la información relativa al estado del tráfico aéreo desde las diversas áreas de un aeropuerto con responsabilidad en tareas diarias de planificación y gestión en tiempo real. Para estas actividades es necesario conocer en todo momento la situación de las aeronaves, de manera que se puedan gestionar los recursos del aeropuerto de la forma más eficiente y permita proporcionar los servicios de forma adecuada. Algunas de éstas tareas son la monitorización y estimación de los tiempos de llegada en el Bloque Aeronáutico, la monitorización general del tráfico y su supervisión el departamento de Operaciones y en el de Dirección y control de tráfico.

La información necesaria para realizar de forma eficiente estas tareas es la localización e identificación de las aeronaves, la estimación de los tiempos de llegada, la representación gráfica y el almacenamiento de los datos.

El principal obstáculo para acceder a dicha información es que sólo está disponible a través de dos tipos de línea específicas: SACTA y DDE. El empleo de estas líneas fuera de su conexión a equipos como las consolas de radar de control de tráfico aéreo no está homologado y carece de la certificación y los permisos necesarios. Esto, junto al elevado coste de las líneas en cuanto a instalación y mantenimiento, impide su uso en todas las tareas de operación y gestión distribuidas por las distintas zonas del aeropuerto.

Por tanto se planteó también como objetivo reducir al máximo estos costes de instalación y mantenimiento en cada uno de los puntos en los que se requiriese la instalación del sistema.

En las especificaciones figuraban además, dada la finalidad del sistema, requisitos de seguridad, fiabilidad y mantenibilidad.

## 1. Proceso de realización

Para realizar del proyecto se siguieron las etapas básicas de la metodología típica en el desarrollo de sistemas software: definición, análisis, diseño, implementación y verificación.

Mediante la aplicación correcta de dicha metodología se establecen los cimientos básicos para conseguir la fiabilidad y la mantenibilidad y permite representar de forma amplia y completa el diseño e implementación y por tanto es un buen modelo a la hora de exponer la estructura del sistema descrito y los pasos realizados.

No es el objetivo de este documento presentar aquí todos estos procesos, que aparecen descritos con detalle en la memoria del proyecto, sino exponer la arquitectura y la funcionalidad del sistema realizado.

## 2. Conceptos previos

Existen dos fuentes de información de tráfico aéreo que proporcionan los datos correspondientes a los vuelos y su localización: los radares de vigilancia primario y secundario y los Sistemas de Automatización y Control de Tráfico Aéreo (SACTA). A cada una de estas fuentes se accede mediante una línea de comunicación. Como ya se ha mencionado anteriormente, éstas son las líneas DDE y SACTA. Los datos proporcionados por éstas son los únicos de los que puede disponer el sistema.

La línea DDE proporciona los datos recogidos por los radares primario y secundario referentes a los blancos detectados, información referente a la posición de la aeronave: distancia, acimut y altura, así como el código del transpondedor, que es un código identificador de la aeronave.

Es una línea serie, síncrona, con modulación FSK a 2400 bps. Los mensajes tienen un formato fijo: la estructura consiste en una trama de 128 bits de longitud con inversión lógica de los bits. Cada trama contiene dos unidades de mensaje.

Los datos SACTA llegan vía módem. Es una línea serie asíncrona que proporciona los datos de decorrelación de vuelos. La decorrelación relaciona los códigos del transpondedor de la aeronave con información de gestión: su código de vuelo y los aeropuertos de origen y destino.

Un código de vuelo se expresa como el código OACI de la compañía y el número del vuelo. Los aeropuertos de origen y destino se expresan mediante sus códigos IATA<sup>1</sup>.

La trama SACTA puede ser de varios tipos, con mensajes de información y longitudes distintas. Un blanco detectado por el radar sólo está identificado por el código transceptor, como se ha mencionado al hablar de la trama DDE. Asociando los datos SACTA al blanco correspondiente (igual código de transpondedor) se dispone de un identificador con mayor información, el código de vuelo, que es mucho más comprensible para un operador humano.

La descripción completa de los formatos de trama DDE y SACTA puede hallarse en la memoria del proyecto.

---

<sup>1</sup> OACI (International Civil Aviation Organization.) e IATA (International Air Transport Association) son dos organizaciones reguladoras en el ámbito aeronáutico.

## **Arquitectura del sistema**

A través del estudio de las especificaciones y del análisis se decidió que la mejor arquitectura posible era la cliente-servidor, ya que de esta forma se concentraba en un solo punto, el servidor, el acceso a las líneas de datos. Con ello se puede acceder a la información sin necesidad de instalar nuevas líneas en cada punto de acceso.

Además, para evitar la instalación del software y los problemas de mantenimiento con cada actualización, se eligió que los clientes del sistema fuesen implementados como Applets de Java.

Los motivos de la elección de este lenguaje de programación son varios: Java es simple, orientado a objetos, distribuido, interpretado, robusto, seguro, neutral a la arquitectura (portable), multitarea y dinámico. Todas estas cualidades muy favorables para un proyecto cliente-servidor como éste. Usarlo permite disponer de una buena base para conseguir estabilidad, fiabilidad, mantenibilidad y seguridad.

Un Applet de Java es una aplicación que se ejecuta como parte de otra; normalmente esta aplicación es un navegador web, y el Applet se carga como un elemento más de la página.

Las ventajas de realizar los clientes como Applets son varias; un Applet tiene toda la potencia de una aplicación Java, pero como se carga junto con la página web en la que está embebido, no hay necesidad de instalación previa ni de configuración, lo que reduce costes y trabajo si el número de usuarios es numeroso o dinámico.

Los Applets son portables, independientes del entorno y del navegador. Si bien Java es seguro, el entorno de un Applet está aislado, con lo cual la ejecución del código desconocido de un Applet nunca puede dañar al sistema en que se ejecuta: salvo que se habiliten permisos expresos, no puede acceder al espacio de ficheros del ordenador en que se ejecuta, ni establecer conexiones salvo con el servidor desde el cual se carga.

Los clientes implementados son aplicaciones Java ejecutables de forma independiente o como Applets, lo que favorece la flexibilidad del sistema. El servidor es una aplicación Java independiente que puede ser ejecutado con o sin interfaz gráfico.

### **1. Servidor**

Se ocupa de realizar la recolección de los datos de las líneas, la integración, interpretación y análisis de los mismos y ofrece un interfaz de acceso a la información para los clientes o para otras aplicaciones externas al sistema. El servidor mantiene en todo momento la configuración y el control de los distintos módulos y la monitorización del GUI<sup>2</sup>. El servidor está formado por tres subsistemas que realizan las funciones anteriores:

- Adquisición de datos
- Comunicaciones
- Gestión y control

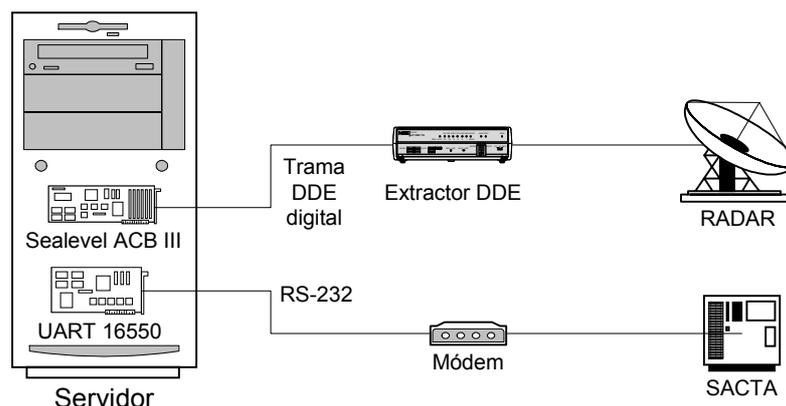
---

<sup>2</sup> GUI (*Graphical User Interface*): interfaz gráfico de usuario, que es el conjunto de elementos gráficos y dispositivos de entrada y salida (monitor, teclado, ratón, etc.) que permiten al operador interactuar con el sistema.

## 1.1. Adquisición de datos

La primera tarea del servidor es la adquisición de los datos. El esquema del hardware empleado es el que aparece en la Fig. 1.

La adquisición de datos de la línea SACTA es sencilla, a través del puerto serie mediante un módem que demodula la señal de la línea SACTA y proporciona la trama lista para su proceso. El caso de la línea DDE es más complejo, pues por el protocolo y el formato que le son propios, no existe un dispositivo con el que se puedan obtener directamente las tramas como en el caso de la línea SACTA.



**Fig. 1 Esquema hardware del servidor**

Para realizar la adquisición de datos DDE se utilizó un módem DDE y una tarjeta de adquisición Sealevel ACB III. Fue necesario implementar un driver para la tarjeta y se eligió como plataforma de destino el sistema operativo Linux, entre otros motivos, por la amplia documentación existente. El driver fue realizado en lenguaje C, pues en Linux los drivers forman parte del kernel del SO. La implementación se realizó para la versión 2.2.14 del kernel, pero es compatible hasta la 2.4.x.

El acceso a los dispositivos en Linux es sencillo, pues se realiza a través de los ficheros de dispositivo: el puerto serie COM2 corresponde al fichero `/dev/Stty01` y a la tarjeta de adquisición se le asignó el fichero de dispositivo `/dev/acbiii`. De esta forma las operaciones de lectura y escritura de un dispositivo se realizan de la misma forma que con un fichero.

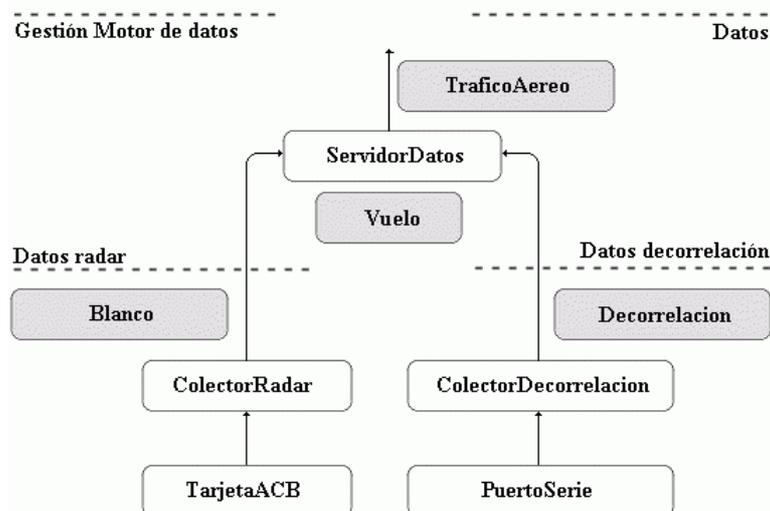
Las tramas SACTA son leídas del puerto serie por el ColectorDecorrelación (ver Fig. 2) y las tramas DDE son extraídas de la tarjeta de adquisición por el ColectorRadar. Estas dos tareas del servidor realizan la decodificación de los datos, el control de errores, cálculo de la tasa de error por trama (FER<sup>3</sup>), etc.

De cada trama DDE se obtiene un Blanco, que representa a un blanco de radar. De la misma forma, de una trama SACTA se obtiene un objeto Decorrelacion. La integración y combinación de Blancos y Decorrelaciones la realiza la tarea ServidorDatos, que proporciona un tipo de datos que combina ambos: el Vuelo. Los Vuelos proporcionan además nueva información procedente del análisis de los datos y de su evolución.

El tipo de datos final es el TraficoAereo, que reúne todos los vuelos presentes en el espacio aéreo en un momento dado. En TraficoAereo se lleva a cabo el análisis completo, el filtrado y la clasificación de vuelos.

<sup>3</sup> De las siglas en inglés *Frame Error Rate*.

La memoria del proyecto presenta con detalle la implementación en Java de las tareas adquisición de datos del servidor. El esquema aparece en la siguiente figura:



**Fig. 2 Adquisición de datos en el servidor**

□ Elementos □ Datos - - Interfaces

## 1.2. Comunicaciones

El módulo de comunicaciones del servidor realiza tres funciones:

- Control de acceso a los usuarios
- Protección de la información
- Distribución de los datos

El control de acceso se realiza mediante un protocolo de autenticación de los usuarios. El protocolo implementado está basado en el algoritmo A3 utilizado en telefonía móvil GSM.

Las comunicaciones se realizan de forma cifrada en todo momento para proteger el acceso a los datos. El protocolo comienza empleando una clave común que se sustituye por una de sesión cuando se inicia la autenticación. Una vez completada, se negocian algunos parámetros de sesión. A partir de este momento se puede acceder a los datos.

Se implementaron varios algoritmos de cifrado de clave pública: RC4/6, DES<sup>4</sup> y AES<sup>5</sup>. DES es el estándar actual más utilizado en las comunicaciones y transacciones comerciales y bancarias electrónicas, propuesto por NIST<sup>6</sup> en 1977. En 1997 eligió el algoritmo Rijndael como nuevo estándar para AES, más robusto y potente que ninguno anterior.

El servidor dispone de estos algoritmos criptográficos y puede seleccionar cuál de ellos emplea con los clientes.

También se emplean estos algoritmos para encriptar los ficheros que almacenan los datos de configuración, cuentas de usuario, etc.

<sup>4</sup> DES (Data Encryption Standar): estándar de encriptación de datos.

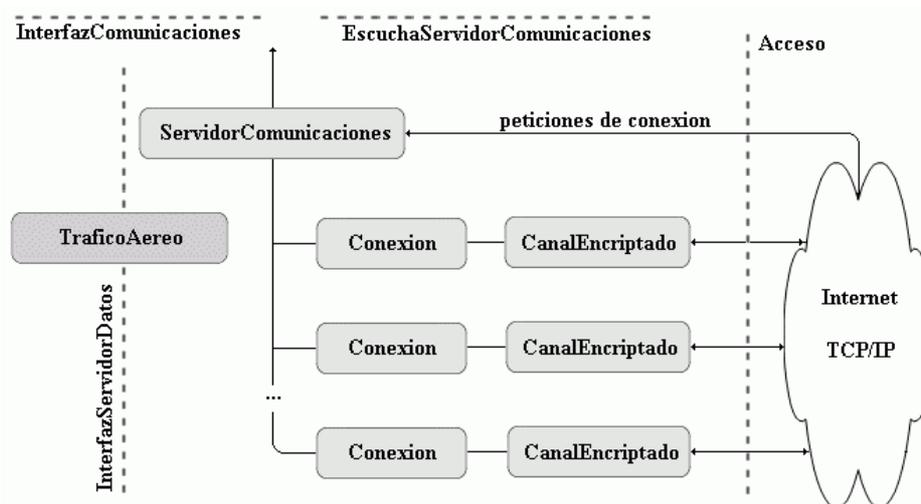
<sup>5</sup> AES (Advanced Encryption Estándar): estándar de encriptación avanzado.

<sup>6</sup> NIST (National Institute of Standards and Technology): instituto nacional estadounidense de estándares y tecnología.

La tarea principal de servicio de datos la realiza la clase `ServidorComunicaciones`. Esta permanece a la espera de peticiones de conexión que le llegan a través de una red TCP/IP genérica.

Para cada una de las peticiones crea un objeto de tipo `Conexion`, que se ocupa de atender la conexión desde la autenticación hasta el envío de la información de tráfico aéreo. Para ello hace uso de un `CanalEncriptado`, que es una conexión bidireccional cifrada.

En el diagrama siguiente se observa el esquema de las comunicaciones:



**Fig. 3 Comunicaciones en el servidor**

### 1.3. Gestión y control

Aunque están claramente separados, el módulo de gestión y control y el GUI están muy relacionados entre sí, puesto que el operador del servidor realiza tanto la gestión como el control del mismo a través del interfaz gráfico.

El módulo de gestión y control está implementado por la clase `Servidor` y el GUI por la clase `ServidorGUI`.

Desde el GUI del servidor puede accederse a las siguientes funciones:

- Arranque y parada de la adquisición de datos y de los servicios. Es necesario introducir previamente el nombre de usuario y la contraseña de una cuenta de administrador<sup>7</sup>.
- Cambio de parámetros: puerto TCP donde se presta el servicio, número máximo de usuarios permitido, frecuencia máxima con que se envían los datos a los clientes, etc.
- Gestión de cuentas de usuario: añadir, modificar y eliminar cuentas.
- Monitorizar el estado del servidor: conexiones abiertas, usuarios conectados, registro de eventos, FER de las líneas de datos, etc.

El GUI del servidor dispone además de un cliente embebido de representación gráfica de datos y estimación del tiempo de llegada (Cliente Consola de Radar, como se verá más adelante), de modo que se puede acceder a toda la funcionalidad de este cliente a través del propio servidor.

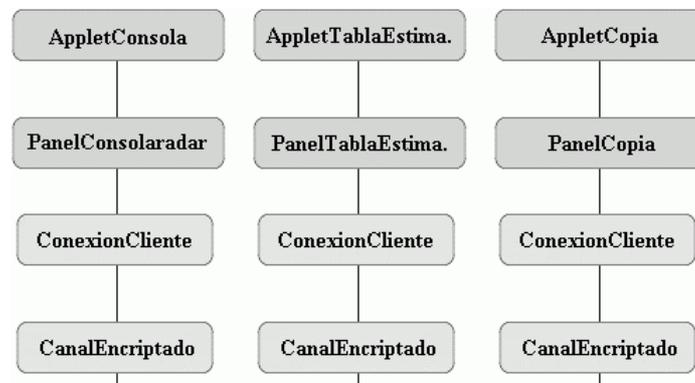
<sup>7</sup> Administrador: nombre genérico de los accesos que cuentan con privilegios especiales.

## 2. Clientes del sistema

Existen tres clientes básicos con los cuales se cubren las posibles necesidades de datos:

- Cliente Copia: realiza el almacenamiento en disco de los datos de tráfico aéreo para posibilitar un análisis o consulta posterior a los mismos.
- Cliente Tabla de estimación: presenta una tabla con los vuelos de llegada y los tiempos estimados de llegada, así como su estado: *Aproximación*, *Aterrizando*, *En pista* y *Operado*.
- Cliente Consola de radar: representa los datos gráficamente sobre un mapa, similar a una consola de radar.
- 

La estructura de todos los clientes es idéntica:



**Fig. 4 Esquema de los 3 clientes básicos**

La clase *ConexionCliente* es la homóloga a la clase *Conexion* del servidor, y se ocupa de realizar, a través del canal bidireccional cifrado *CanalEncriptado*, todas las funciones de conexión, autenticación y transferencia de información con el servidor. Cada cliente implementa el GUI mediante un Panel (*PanelConsolaRadar*, *PanelTablaEstimacion* y *PanelCopia*) y un Applet (*AppletConsolaRadar*, *AppletTablaEstimacion* y *AppletCopia*).

En todos los clientes se solicita el nombre de usuario y la contraseña para realizar la autenticación y permitir el acceso al servicio de datos.

Otro elemento común es el indicador de calidad en las líneas de datos, que se corresponde con los valores medios de FER a corto y largo plazo (filtros de media móvil).

## 2.1. Cliente Copia

Realiza el almacenamiento en ficheros de los datos. En la siguiente figura se presenta el aspecto del GUI correspondiente:

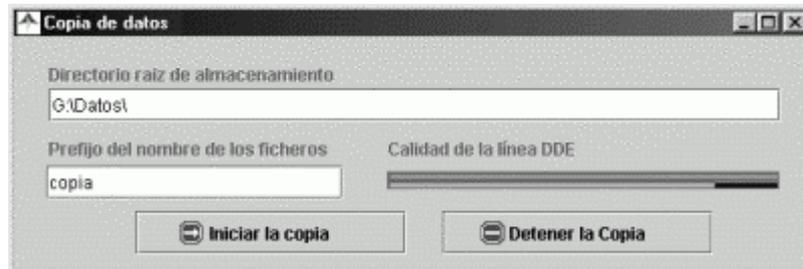


Fig. 5 Cliente Copia

En el cliente de copia es posible especificar el directorio de almacenamiento y el prefijo del nombre de los ficheros; cada día se abre un fichero distinto y el nombre del fichero se compone de dicho prefijo y de una cadena indicativa de la fecha correspondiente. Existen también controles para iniciar y detener la copia de datos.

Una de sus aplicaciones es la de sustituir las actuales cintas de registro en las que se almacenan actualmente dichos datos.

## 2.2. Cliente Tabla de estimación

Presenta una tabla con los vuelos de llegada y los tiempos estimados de llegada, así como su estado: *Aproximación*, *Aterrizando*, *En pista* y *Operado*.

Se puede establecer un filtrado por alturas estableciendo la altura mínima y máxima de las aeronaves mediante el doble control deslizante que se encuentra a la derecha en el GUI:



Fig. 6 Cliente Tabla de estimación

En el cálculo de la estimación del tiempo de llegada se tiene en cuenta el rumbo de aproximación, la distancia, la altura y la velocidad.

Como dependiendo del rumbo las maniobras de aproximación son distintas, existe un botón en el GUI que permite establecer si la aproximación se realiza de Noroeste a Sudeste, de Sudeste a Noroeste, o según la dirección que indique el servidor. La dirección puede variar por el viento, el tipo de aeronave, etc. y afecta al tiempo de llegada, lo cual ha de ser tenido en cuenta.

### 2.3. Cliente Consola de radar

Intenta imitar las consolas de radar representando los vuelos gráficamente sobre un mapa del entorno del aeropuerto:

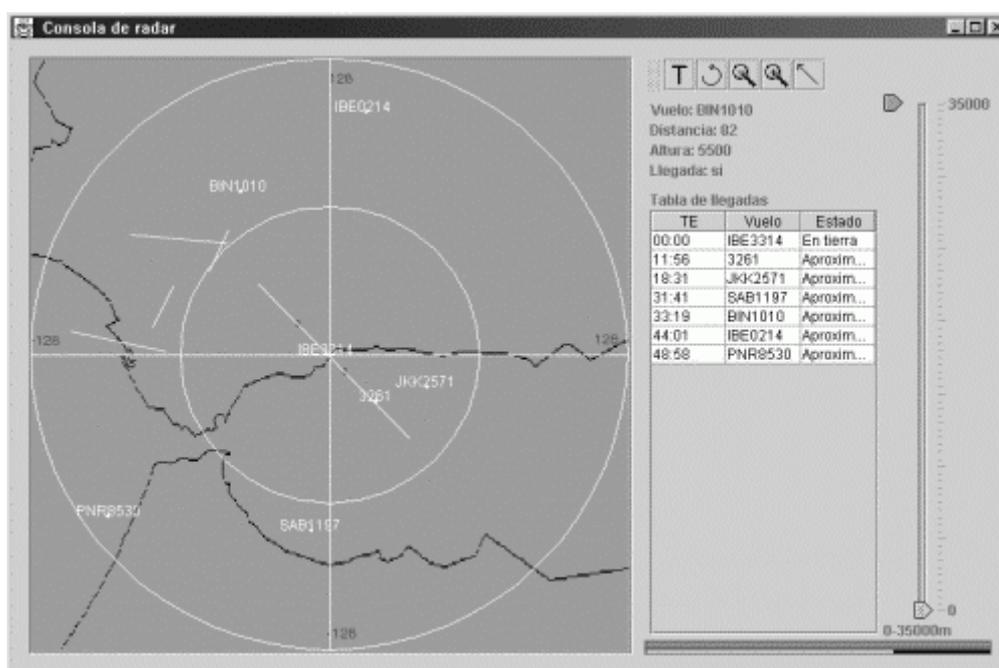


Fig. 7 Cliente Consola de radar

Integra como un componente más un panel de Tabla de estimación, de manera que dispone de todas sus funciones: filtrado por alturas, configuración de la dirección de aproximación y estimación de los tiempos de llegada.

Combina además funciones de filtrado por origen/destino del vuelo. Permite seleccionar con el ratón una aeronave y mostrar su información. El mapa dispone de varios niveles de resolución (*zoom*) en el rango de 16 a 128 millas náuticas y permite seleccionar el tipo de etiqueta de la aeronave (código transceptor o código de vuelo). Otra función es la rotación de las etiquetas para evitar el solapamiento en el caso de aeronaves cuya proyección en el mapa está próxima.

### 3. Empaquetado y distribución

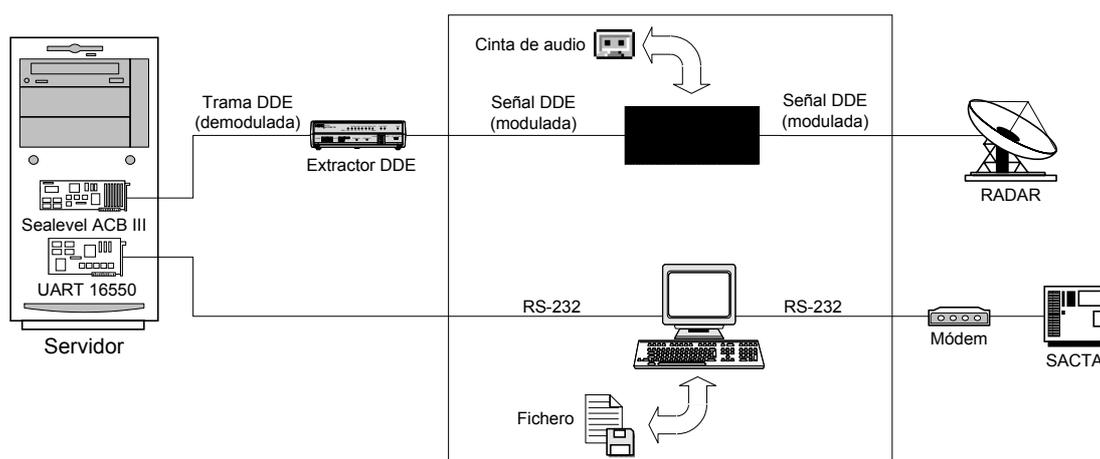
Todas las clases generadas, los archivos de imágenes, mapas y ficheros de configuraciones estáticas se han comprimido en un solo fichero Jar firmado digitalmente. El formato Jar realmente es un fichero con formato de compresión Zip que, sin embargo, es ejecutable por Java. La ventaja es que todo lo necesario para ejecutar las aplicaciones se encuentra en un solo fichero. Además, al estar comprimido acelera la descarga cuando se ejecutan los clientes como Applets en una página web.

Mediante la verificación de la firma digital del Jar firmado se consigue verificar la identidad del origen del software (AENA) y el estado inalterado del software desde que fue firmado (no ha sufrido ninguna manipulación).

### Banco de pruebas del sistema

La verificación del sistema requería realizar una serie exhaustiva de pruebas con el fin de comprobar el comportamiento frente a situaciones controlables.

Ante la imposibilidad de trabajar con conexión directa a las líneas de datos durante la fase de verificación se ideó un sistema de prueba que emulaba las líneas de datos:



**Fig. 8 Banco de pruebas**

Como la señal DDE tiene una modulación de baja frecuencia, se grabaron los datos en cinta usando un grabador/reproductor de cassette con control de velocidad, para permitir corregir las derivas en frecuencia provocadas por derivas en la velocidad de la cinta. Para las pruebas. La señal del reproductor se inyectaba al módem DDE.

Los datos SACTA, una vez demodulada la señal, se guardan en ficheros y se transmiten por el puerto serie utilizando otro ordenador.

De esta forma se consiguieron emular las dos líneas de datos y se pudieron realizar las pruebas necesarias sin interferir en las líneas.

Mediante este sistema se comprobó el correcto funcionamiento del sistema en cuanto a adquisición de datos se refiere, pero para realizar pruebas de tareas más complejas, como el análisis de la información, fue necesario generar datos de prueba. Aprovechando que en Linux

los dispositivos se acceden mediante ficheros, se sustituyeron éstos por unos medida de cada prueba.

Se escribieron programas simuladores que generaban datos de tráfico con formato de tramas DDE y SACTA y que volcaban en ficheros, y éstos fueron usados en lugar de los de dispositivo.

Mediante estos programas se generaron trayectorias precisas de los blancos para comprobar que el análisis de los vuelos era correcto en cada caso posible.

Se realizaron asimismo pruebas de carga y sobrecarga del servidor para comprobar los límites de servicio manteniendo unos ciertos parámetros de calidad mínima y los resultados fueron más que satisfactorios: el servidor es capaz de prestar servicio a casi 400 clientes, siendo la especificación inicial de menos de 50. Para interpretar esta cifra hay que tener en cuenta que las pruebas se realizaron en un ordenador de prestaciones más bien modestas teniendo en cuenta la capacidad de los servidores actuales.

## **Líneas actuales de desarrollo**

En un entorno tan complejo como es la gestión de un aeropuerto resulta imprescindible el mantenimiento y la actualización del sistema para adaptarse a la continua evolución de las necesidades.

Además de esta labor de mantenimiento es necesario ampliar el sistema para añadir nuevas funciones. Ya durante el proceso de desarrollo de este proyecto surgieron algunas ideas de posibles ampliaciones y aplicaciones en las que el acceso a la información de tráfico aéreo sería una herramienta sumamente útil.

Actualmente el proyecto sigue vivo con varias vertientes de desarrollo:

- Adaptación del formato de intercambio de datos de radar ASTERIX<sup>8</sup>, que será el nuevo estándar en el ámbito Europeo.
- Mejora de la estimación de tiempos de llegada y del control de la posición de las aeronaves mediante la inclusión de las sendas de planeo y las maniobras estándar de aproximación, aterrizaje, etc.
- Nuevas funcionalidades: empleo de mapas de representación de coordenadas vectoriales.
- Nuevos tipos de clientes: actualización y sincronización automática de bases de datos.
- Creación de un aula de entrenamiento virtual para controladores aéreos: mediante el uso de las posibilidades de representación gráfica del cliente Consola de radar y de los datos almacenados por el cliente Copia.
- Adaptación de aplicaciones para hacer uso de las posibilidades de automatización que ofrece el interfaz de acceso a los datos. Son múltiples las aplicaciones actualmente en uso en el aeropuerto, tanto para la gestión como para la operación, que se pueden beneficiar de la conectividad del sistema. Por ejemplo, se está adaptando la megafonía para los señaleros .

## **Conclusiones**

El objetivo del presente proyecto era el desarrollo de un sistema que permitiese proporcionar a un grupo amplio de usuarios los datos de tráfico aéreo de tres formas distintas: representación gráfica sobre un mapa, estimación de tiempos de llegada y almacenamiento de los datos para su uso posterior. Las principales restricciones impuestas eran la portabilidad, la flexibilidad y el bajo coste.

Se ha hecho uso de una técnica habitual en el proceso de desarrollo de productos software: especificación, análisis, diseño, implementación y verificación.

Se ha desarrollado un sistema con arquitectura cliente-servidor con cuatro aplicaciones: el servidor y tres aplicaciones cliente. Adicionalmente se ha creado un paquete Java que se puede usar como librería para futuras ampliaciones.

El servidor del sistema corre bajo el sistema operativo Linux, para el que se ha desarrollado un driver para la tarjeta de adquisición de datos. Puede utilizarse con o sin interfaz gráfico y su funcionalidad cumple de sobra las especificaciones.

Cada uno de los clientes cubre una de las necesidades de datos. Los clientes son Applets de Java, que pueden ser ejecutados como parte de una página web o como aplicaciones

---

<sup>8</sup>All purpose STructured Eurocontrol Radar Information eXchange.

independientes. También puede incluirse un cliente como un control más dentro de otra aplicación Java.

Las aplicaciones se han empaquetado en formato *JAR* para hacer más eficiente su distribución y permitir la firma digital del *software*.

Como documentación existe, además de la memoria del proyecto, un Manual de usuario y un Manual de referencia.

El diseño y realización de todo el sistema se ha llevado a cabo manteniendo en mente cinco objetivos: seguridad, flexibilidad, portabilidad, fiabilidad y sencillez.

Para cubrir el objetivo de la seguridad se ha hecho uso de técnicas criptográficas y de las características de los Applets de Java, que proporcionan un entorno de ejecución aislado.

La flexibilidad se ha logrado mediante un diseño cuidado de los interfaces tanto internos como externos de las aplicaciones, y gracias a la filosofía de la programación orientada a objetos. Esto permite maximizar la durabilidad del software desarrollado y facilitar su mantenimiento y ampliación.

Otro aspecto que confiere flexibilidad al sistema desarrollado es la existencia de un interfaz de acceso a los datos para otras aplicaciones externas al sistema. Esto aumenta las posibilidades de uso y aplicación del sistema. Las aplicaciones externas avalan su eficacia y refuerzan el apoyo y el crecimiento del mismo.

La realización del proyecto supuso una integración multidisciplinar: adquisición de datos, drivers, formatos de intercambio de información de radar, tecnologías de Internet, TCP/IP, Applets de Java, técnicas criptográficas, protocolos de autenticación y firma digital, desarrollo de protocolos propios internos al sistema, etc.

Se ha creado un sistema formado por un conjunto de aplicaciones, que es completo y puede funcionar de forma independiente, original y efectivo para desarrollar o apoyar multitud de tareas en el aeropuerto.

El proyecto desarrollado se encuentra instalado en el Bloque Aeronáutico del Aeropuerto de Málaga, demostrando su gran utilidad y aplicabilidad real en la gestión de recursos aeroportuarios y la automatización de los mismos y avalando su idoneidad para ser instalado en otros aeropuertos.

#### **4. Resumen de los datos**

---

##### **Datos del autor**

Nombre y apellidos            Juan José Escobar Mira

##### **Datos del proyecto de fin de carrera**

Título                            Sistema cliente-servidor para datos de tráfico aéreo

Tutor                            Gabriel Valencia Miranda  
                                      Alfredo García Lopera (Coordinador)

Departamento                Tecnología Electrónica  
Universidad                    Universidad de Málaga

Fecha de lectura              10 de Julio de 2001  
Calificación                    Matrícula de Honor (unanimidad)  
Tribunal                        Dr. Alfonso Ariza Quintana  
                                      Gabriel Valencia Miranda  
                                      Alfredo García Lopera

Málaga, 23 de Enero de 2002

Ilmo. Decano del Colegio Oficial de Ingenieros de Telecomunicación

Le ruego acepte la documentación del proyecto titulado "**Sistema cliente-servidor para datos de tráfico aéreo**" que acompaña esta carta a fin de presentarlo a la convocatoria de los Premios **COIT/AEIT**

El proyecto que presento surgió de un contrato de colaboración existente entre Aeropuertos Españoles y Navegación Aérea (AENA) y el Departamento de Tecnología Electrónica de la Universidad de Málaga. Además del tutor del proyecto, Gabriel Valencia Miranda, también ha colaborado estrechamente en su realización Alfredo García Lopera, responsable principal y coordinador de los proyectos del contrato con AENA, siendo a efectos prácticos cotutor del mismo. De ambos adjunto una carta de presentación.

El proyecto surgió por la fuerte necesidad de disponer de los datos de radar en los diversos departamentos del aeropuerto y su objetivo era conseguir un sistema de bajo coste que permitiese adquirir y distribuir esta información por la propia red de AENA.

Se desarrolló un sistema con arquitectura cliente-servidor con cuatro aplicaciones: el servidor y tres tipos de clientes. El servidor realiza las tareas de adquisición, integración, combinación e interpretación de los datos y ofrece un interfaz de acceso a los mismos para los clientes. Las aplicaciones cliente cubren tres necesidades diferentes de la información: representación gráfica, estimación de los tiempos de llegada de aeronaves y copia de datos para su uso posterior. Los clientes se implementaron como Applets Java, para que puedan ser ejecutados como parte de una página web, pero también pueden ser ejecutados como aplicaciones independientes. Adicionalmente se creó un paquete Java (librería) para ser utilizado en la creación de nuevas aplicaciones.

El interfaz de acceso que ofrece el servidor permite que cualquier aplicación pueda utilizar los datos de tráfico aéreo. Actualmente son varias ya las aplicaciones de operación y gestión que se pretenden modificar para conseguir mayor automatización de los procesos.

La realización del proyecto supuso una integración multidisciplinar:

- Adquisición de datos: driver para Linux de una tarjeta de comunicaciones.
- Formatos de intercambio de información de radar: DDE y SACTA.
- Tecnologías de Internet: TCP/IP, Applets de Java, etc.
- Seguridad: Técnicas criptográficas (cifrado), protocolos de autenticación y firma digital.
- Desarrollo de protocolos propios internos al sistema.

Por otra parte el proyecto no ha dejado de desarrollarse, sino que continúa en tres vertientes:

- Adaptación del protocolo ASTERIX, nuevo estándar de intercambio de datos de radar.
- Modificación de aplicaciones ya existentes (externas al sistema) para aprovechar las posibilidades de automatización.
- Ampliación del sistema con nuevas funcionalidades: mapas de coordenadas vectoriales, nuevos tipos de clientes...

El sistema fruto de este proyecto se encuentra instalado y funcionando en el Departamento de Operaciones de AENA del aeropuerto de Málaga. Otros aeropuertos se muestran interesados por el mismo, aunque no hay aún pronunciamiento oficial, pues el se encuentra en periodo de evaluación hasta conseguir el certificado de software homologado por AENA.

Creo realmente que la calidad técnica del proyecto, las áreas que abarca, así como su gran utilidad en la gestión de recursos aeroportuarios y la automatización de los mismos, lo cualifican para presentarlo como candidato a los premios del Colegio.

Sin más que decir quedo a su disposición para cualquier aclaración que necesite.

Saludos cordiales,

Juan José Escobar Mira  
Ingeniero de Telecomunicación



**UNIVERSIDAD DE MÁLAGA**  
**Departamento de Tecnología Electrónica**

E.T.S. INGENIEROS DE TELECOMUNICACIÓN

Campus de Teatinos - 29071 Málaga  
Telf.: 95 213 13 52 - Fax: 95 213 14 47  
E-mail: dte@dte.uma.es

Málaga, 15 de Enero de 2002

Ilmo Decano del Colegio Oficial de Ingenieros de Telecomunicación  
Madrid

Le ruego acepte en la actual convocatoria para el Mejor Proyecto Fin de Carrera el realizado por D. Juan José Escobar Mira titulado **“Sistema Cliente-Servidor para Datos de Tráfico Aéreo”**

Este Proyecto ha sido realizado dentro de un acuerdo de colaboración con AENA del cual soy responsable como Investigador Principal. En él, su autor ha logrado desarrollar una aplicación para Aviación Civil que se ha demostrado de gran utilidad y amplio uso. Fundamentalmente integra los datos de identificación comercial de vuelo junto a los procedentes de los Radares Primario y Secundario de un Centro de Control en un Servidor . Este servidor los dispone en la Red Local del Aeropuerto como una página WEB mediante una aplicación Java sobre LINUX. De este modo se tiene una representación de vuelos en 2D y sus datos identificativos. Se realizan cálculos realistas de tiempos de llegada de cada vuelo los cuales se exportan a las aplicaciones de Gestión y Operaciones del Aeropuerto, Estas pueden proceder en automático sin necesidad de operadores externos que introduzcan datos de llegada mediante su visualización humana.

Juan José Escobar ha desarrollado su trabajo con una gran inventiva y eficiencia. Ha logrado desarrollar una aplicación de las Telecomunicaciones en Aviación Civil, integrando protocolos propios junto a una aplicación Internet. El proyecto está instalado en el Aeropuerto de Málaga funcionando a plena satisfacción y demostrando su idoneidad para ser instalado en otros Aeropuertos.

Quedo a tu disposición para cuantas aclaraciones necesites  
Saludos cordiales, Alfredo

Fdo: Alfredo García Lopera  
Profesor de Tecnología Electrónica



UNIVERSIDAD DE MÁLAGA  
Departamento de Tecnología Electrónica

Málaga, 18 de Enero de 2002

El Proyecto Fin de Carrera que se presenta, titulado **Sistema cliente-servidor para datos de tráfico aéreo**, realizado por D. **Juan José Escobar Mira** ha sido desarrollado, durante el curso 2000/2001, en el Departamento de Tecnología Electrónica como consecuencia de un contrato de colaboración existente entre la Universidad de Málaga y Aeropuertos Españoles y Navegación Aérea (AENA).

El objeto del proyecto es la representación gráfica en tiempo real del tráfico aéreo del Aeropuerto de Málaga junto con la monitorización y estimación de los tiempos de llegada de los vuelos. Una información que anteriormente era exclusiva de las consolas de radar y que, gracias a la aplicación desarrollada en este Proyecto Fin de Carrera, ahora es posible disponer de ella en otros Departamentos del Aeropuerto de Málaga que la consideran muy importante para realizar actividades de gestión, supervisión y planificación del tráfico aéreo.

La aplicación desarrollada obtiene los datos de los blancos directamente del radar secundario del Aeropuerto de Málaga y los datos de la decorrelación que asocian el transceptor del vuelo con el código de vuelo via modem. La información proporcionada por la aplicación se distribuye a través de la red interna de AENA mediante un sistema cliente-servidor.

Gabriel Valencia Miranda  
Profesor Dpto. Tecnología Electrónica