

VISORMAP CS: INTEGRACIÓN DE HERRAMIENTAS DE TELEDETECCIÓN / GIS EN UNA VERSIÓN CLIENTE/SERVIDOR

J. Peralta (*), J. García-Consuegra (*, **) y J. E. Córcoles (*).

jperalta@idr-ab.uclm.es

(*) Sección de Teledetección y SIG, IDR. Campus Universitario s/n,02071 Albacete.

(**) Departamento de Informática, EPSA. Campus Universitario s/n,02071 Albacete.

RESUMEN

A continuación se presenta una arquitectura Cliente/Servidor de una herramienta en la que se integran un módulo de teledetección (VisorMap RS) y otro de GIS. El objetivo de la misma es obtener un sistema lo suficientemente flexible para adaptarse a las necesidades de diferentes usuarios y eficiente en la gestión de recursos. Se detallarán algunos aspectos del diseño e implementación así como las posibilidades que nos ofrece. Se ha pretendido facilitar el acceso de los usuarios a la información, aislándolo de todos los detalles de la implementación como ha sido posible.

ABSTRACT

A Client/Server architecture of an application that integrates a remote sensing (VisorMap RS) and GIS modules is outlined. The aim is to get a system as flexible as possible in order to satisfy different user requirements and efficient in resources management. Some design and implementation issues are described as well as possibilities that this system offers to users. User's access to the information has been simplified as much as possible, and the minimum implementation details have remained visible from user point of view.

Palabras clave: Cliente/Servidor, GIS, teledetección, Internet, LAN.

INTRODUCCIÓN

La Sección de Teledetección y SIG de la Universidad de Castilla-La Mancha, de la cual son miembros los autores de este trabajo, está desarrollando actualmente dos aplicaciones en las áreas de Teledetección (VisorMap RS) y SIG (García-Consuegra 1998). La primera de ellas tiene, entre otras, la funcionalidad de registrar imágenes de un modo automático, corrección de errores, segmentación de imágenes, clasificación, etc. Mientras, el segundo módulo nos permite realizar tareas básicas de los SIG relacionadas con la visualización (sobre diferentes tipos de información), bases de datos e interfaz gráfico de usuario. Esta información se almacena en diferentes capas. La peculiaridad de estos dos módulos es que pueden llegar a integrarse dentro de un entorno de trabajo en el que la salida del primero se convierte en la entrada de información para el segundo.

A partir de la versión local, se ha desarrollado otra versión distribuida que nos permite trabajar de acuerdo al modelo Cliente/Servidor (Jerke 1997) a través de redes locales (LAN) o de área extensa (Internet) indistintamente.

Para conseguir la versión Cliente/Servidor de estas aplicaciones se ha realizado un estudio de la naturaleza de estos sistemas de información así como de sus principales características. Así, por

ejemplo, a la hora de realizar clasificaciones (y en general el módulo de teledetección completo) se requerirán operaciones de un alto costo computacional, mientras que el módulo GIS se encontrará más condicionado por la estructuración y almacenamiento de la información así como por su acceso.

Para reflejar estas dos aproximaciones (de un lado las necesidades de capacidad de procesamiento y de otro el almacenamiento de la información) la herramienta ofrece una serie de servicios a los clientes, entre los cuales se encuentran el almacenamiento y gestión de la información, así como la ejecución de procesos de elevada carga. La consecución de estos dos aspectos puede llegar a ser contradictoria, ya que una mayor eficiencia a la hora de obtener resultados inmediatos puede suponer una peor gestión del almacenamiento de la información, por lo que en todo momento se ha intentado mantener un equilibrio entre ambos extremos.

Desde el punto de vista de la gestión de la información, una arquitectura Cliente/Servidor nos ofrece una serie de ventajas o posibilidades que se ha intentado que repercutan en una mayor funcionalidad desde el punto de vista del cliente o usuario final. Así, por ejemplo, la duplicidad de información facilita la gestión de peticiones de modo paralelo, de forma que pueden existir diferentes

servidores con la misma información con el objetivo de proporcionar seguridad frente a caídas (en caso de que un servidor quede inoperativo, las peticiones referentes a la información que ofrecía pueden ser redirigidas a las réplicas existentes), o simplemente un mejor rendimiento (desde el momento que permite que dos operaciones se ejecuten de forma simultánea).

Para la gestión y control de todo el sistema de información se ha desarrollado un módulo que permite abstraer de la arquitectura del sistema a los usuarios finales, de forma que el modo de trabajo de los mismos no variará sustancialmente respecto a las aplicaciones locales.

Todo esto se ha implementado mediante la tecnología ActiveX de Microsoft (basada en el protocolo DCOM). Este aspecto de la solución se comentará en el apartado dedicado a la implementación.

DESCRIPCIÓN DE LA ARQUITECTURA

En el sistema de información que se está describiendo se han integrado, al mismo tiempo que la gestión de la información propia de una herramienta de estas características (imágenes, bases de datos, etc.), los conceptos de usuario y grupo de la aplicación, a cada uno de los cuales se les asocian una serie de derechos o privilegios sobre los

diferentes recursos que forman el sistema. Esta clasificación en usuarios y grupos nos permite diferenciar diferentes perfiles de acceso (directores de proyecto, administradores del sistema, usuarios habituales, etc.) en función de los cuales se dispondrán de una serie de funcionalidades bien diferenciadas.

El sistema de información propuesto no se reduce a determinar la existencia de un servidor al cual se conectarán los clientes para pedir la información que en cada momento necesiten, sino que se trata de un esquema en el que intervienen diferentes módulos con funcionalidades bien definidas (Figura 1). A continuación se describirán brevemente los más importantes.

- Un Servidor de Tareas encargado de validar la identidad de los usuarios y mantener la información a partir de la cual gestionará los derechos de los usuarios o grupos sobre los objetos implicados en cada consulta o petición. Al mismo tiempo es el componente encargado de conocer en todo momento el estado del sistema y la ubicación real de la información. Por lo tanto se trata del módulo que mantiene el control general del sistema y lo mantiene en un estado consistente.
- Cierta número de Servidores Colaboradores encargados de atender realmente las peticiones que les asigna el Servidor de Tareas. Se

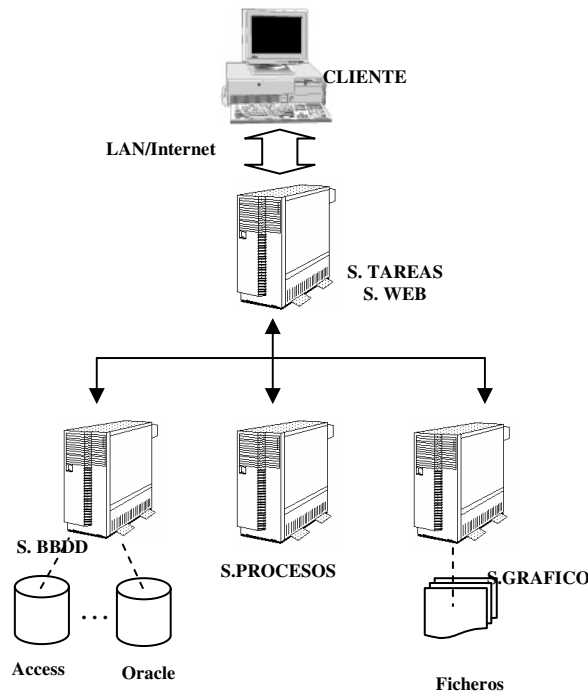


Figura 1. Arquitectura de la aplicación descrita.

distinguirán diferentes tipos dependiendo de la naturaleza de la información o tareas que deban soportar. Así, en principio se dispone de un Servidor Gráfico encargado de transferir la información gráfica (ya sea en formato vectorial, raster, icono, texto o poligonal) a los clientes. Este servidor debe proporcionar para cada petición, toda la información necesaria para una correcta utilización, es decir, estudia las dependencias existentes entre los diferentes ficheros y proporciona todos aquellos directamente relacionados. El Servidor de Bases de Datos se encarga de ofrecer los resultados a las consultas alfanuméricas que realizan los clientes, para lo cual nos ofrece una abstracción de las bases de datos que se encarga de gestionar (Jerke 1997), o lo que es lo mismo, nos facilita un acceso remoto a diferentes bases de datos (incluso a aquellas basadas en ficheros como pueden ser Access o dBase). Por último, el Servidor de Procesos se caracteriza por poseer una gran capacidad de procesamiento, con lo cual ofrece ejecución de procesos definidos mediante un lenguaje script propio.

- Un Servidor Web que permite realizar accesos a la información del sistema sin necesidad de instalar el módulo cliente de la aplicación. Obviamente la funcionalidad ofrecida por este método de acceso se verá reducida en cierta manera, pero a cambio conseguimos independencia de la plataforma sobre la que se ejecuta y un interfaz más sencillo para usuarios no especializados. Esta posibilidad va dirigida a aquellos usuarios que no accederán habitualmente al sistema o que no disponen de la aplicación cliente y sin embargo necesitan realizar algún tipo de consulta.

RECURSOS NECESARIOS

Uno de los objetivos principales de diseño que se han intentado reflejar a la hora de decidir el diseño que mejor se ajustaba a las necesidades existentes ha sido el mantener un coste económico mínimo para el usuario. Esto quiere decir que se ha pretendido evitar soluciones comerciales que puedan suponer un gran desembolso económico a la hora de disponer de licencias de uso.

Uno de los elementos fundamentales de este tipo de sistemas de información es la existencia de un servidor de Bases de Datos, pero su disponibilidad está fuertemente condicionada por las posibilidades económicas. Este problema se ha solucionado mediante la implementación de la herramienta a través de interfaces de acceso a datos que nos permiten una independencia total de la solución

comercial elegida. Así aquellos que dispongan de Bases de Datos como Oracle o Informix podrán aprovechar sus prestaciones, mientras que aquellos que posean la información en sistemas de ficheros (como pueden ser Access o dBase) no perderán en ningún momento la funcionalidad básica, que en este caso incluye la consulta remota de información, de la cual se encargarán los módulos Servidores de Bases de Datos descritos anteriormente.

Por otra parte, las disponibilidades de hardware condicionarán el rendimiento del sistema, pero en ningún caso supondrán una limitación a la hora de implantar este sistema de información. De hecho los requerimientos mínimos se reducen a un ordenador personal, que actuará como servidor, conectado a una red que proporcione acceso a los clientes. En caso de que se desee una mejor distribución de la carga, se pueden manejar diferentes configuraciones (llevar los Servidores Gráficos a otras máquinas, aislar al Servidor de Procesos en una única máquina, etc.) y en función de ellas y de los recursos disponibles, obtendremos un rendimiento diferente. La configuración óptima dependerá de la topología de la red así como de las características de las máquinas disponibles.

ALGUNOS DETALLES DE IMPLEMENTACIÓN

La solución propuesta se ha desarrollado para plataformas Windows NT y 95/98. Se basa en la utilización de la tecnología ActiveX de Microsoft, basada en el estándar Component Object Model (Appleman 1997), que nos permite comunicar entre sí módulos situados en máquinas diferentes (incluso a través de diferentes redes). Sin embargo todos los procesos y protocolos de comunicación entre los diferentes componentes (y potencialmente diferentes máquinas) se realizan de un modo transparente al usuario, de forma que éste no debe ser consciente en ningún momento de aspectos de la implementación que puedan surgir debido a la naturaleza distribuida de la herramienta.

El protocolo DCOM en el que se basa la tecnología ActiveX se encuentra disponible de forma nativa en Windows NT, mientras que en Windows 95/98 se debe instalar a partir de paquetes gratuitos que ofrece Microsoft.

En una sesión de un usuario cualquiera se suceden los siguientes pasos:

- El Cliente establece la conexión con el Servidor de Tareas, el cual validará la identidad del usuario y activará un servicio que registrará todas las acciones que realice durante esa sesión.

- Una vez que la conexión se ha autorizado, el cliente requerirá la realización de una cierta tarea (consulta a Base de Datos, obtención de una capa gráfica, etc.), la cual se añadirá a una cola de peticiones. En ese momento el control se devuelve inmediatamente al cliente, de forma que puede continuar realizando tareas para las cuales no necesite la información que acaba de solicitar.
- Dependiendo de la naturaleza de la petición, el Servidor de Tareas solicitará al Servidor Colaborador que disponga de la información solicitada la ejecución de la misma.
- El Servidor Colaborador enviará la información resultado directamente al Cliente, evitando de ese modo el posible cuello de botella que podría suponer la transferencia de todas las peticiones a través de la misma máquina. Cada una de las peticiones se atiende en una hebra de ejecución distinta, permitiendo de ese modo la ejecución concurrente de diferentes peticiones.
- Cuando el Servidor Colaborador concluya el trabajo, lo notificará al Servidor de Tareas y al Cliente, actualizándose en ese momento el estado del sistema (liberación de recursos).

El concepto de abstracción del usuario final de los detalles de implementación se ha extendido al interfaz de acceso a la información, definiéndose para ello dos estructuras bien diferenciadas:

- El catálogo, cuya finalidad es mantener toda la información de control referente a la información gráfica o capas de un mapa (formatos raster, polígono, vector, icono y texto). Para acceder a estos datos (a través de las estructuras de control subyacentes) se utilizará un interfaz que se ha diseñado con el objetivo de abstraer los conceptos de fichero y directorio, de forma que un objeto del catálogo puede estar formado por varios ficheros almacenados en una ruta de una máquina que el usuario desconoce, y que en ningún momento necesitará conocer.
- El repositorio, que es una estructura análoga al catálogo pero dirigida a gestionar y mantener las consultas y tablas de información alfanumérica de las diferentes Bases de Datos que pueden existir en el sistema. Al igual que el catálogo, existe un interfaz gráfico que evita al usuario conocer la localización exacta de la información así como el modo de acceso a ella. De hecho, las consultas se realizan mediante el lenguaje SQL (Structured Query Language), pero su conocimiento es innecesario, ya que la aplicación dispone de un generador de SQL

que utiliza, a parte de la información que solicita al usuario, la información de control existente en el repositorio.

CONCLUSIONES

Debido a la naturaleza cambiante de los sistemas de información geográfica (utilicen o no técnicas de teledetección en alguna de las fases del tratamiento de la información), la utilización de una versión distribuida en la cual diferentes usuarios puedan compartir la misma información en las herramientas de este tipo resulta ampliamente ventajosa frente a otros acercamientos (aplicaciones monopuesto, sistemas centralizados, etc.).

Al mismo tiempo, la reducción del coste económico del hardware de propósito general (como pueden ser ordenadores personales, redes locales, etc.) y la mejora en cuanto a las prestaciones y calidad de las comunicaciones, suponen un óptimo aprovechamiento de las ventajas que nos ofrecen este tipo de sistemas de información, minimizando los posibles inconvenientes (como puede ser la reducción de prestaciones).

Actualmente los primeros prototipos de esta herramienta se encuentran en funcionamiento, obteniendo unos resultados francamente esperanzadores, ya que la simulación de un entorno de trabajo con 3600 usuarios concurrentes realizando peticiones de información de carácter gráfico demostró la robustez del mismo, con el único inconveniente del descenso de prestaciones debido al cuello de botella que puede suponer la red de comunicaciones.

BIBLIOGRAFÍA

- Appleman, D. 1997. Developing ActiveX components with Visual Basic 5.0: A guide to the perplexed. Emeryville: Ziff-Davis.
- García-Consuegra, J.D., Gallud, J.A. y Abad, F., 1998. IDR Visor Map: a simple and friendly geographical system. *GIS technologies and their environmental Applications*. Computational Mechanics Publications.
- Jerke, et alii. 1997. Visual Basic 5 Client/Server How-To. Waite Group Press.
- Laurini, R. 1994. Sharing geographic information in distributed databases. *URISA, Urban and Regional Information Association*.
- Microsoft, 1998. ActiveX Core Technologies Moving to Leading UNIX platforms. <http://www.microsoft.com/presspass/press/1997/Mar97/Unixpr.htm>